

FreeBSD kernel CAM code Reference Manual

Generated by Doxygen 1.4.7

Sat Feb 24 10:09:34 2007

Contents

1	FreeBSD kernel CAM code Main Page	1
2	FreeBSD kernel CAM code Directory Hierarchy	3
2.1	FreeBSD kernel CAM code Directories	3
3	FreeBSD kernel CAM code Data Structure Index	5
3.1	FreeBSD kernel CAM code Data Structures	5
4	FreeBSD kernel CAM code File Index	11
4.1	FreeBSD kernel CAM code File List	11
5	FreeBSD kernel CAM code Directory Documentation	13
5.1	/usr/src/sys/cam/ Directory Reference	13
5.2	/usr/src/sys/cam/scsi/ Directory Reference	15
5.3	/usr/src/ Directory Reference	16
5.4	/usr/src/sys/ Directory Reference	17
5.5	/usr/ Directory Reference	18
6	FreeBSD kernel CAM code Data Structure Documentation	19
6.1	asc_key Struct Reference	19
6.2	asc_table_entry Struct Reference	20
6.3	async_node Struct Reference	21
6.4	bus_match_pattern Struct Reference	22
6.5	bus_match_result Struct Reference	23
6.6	cam_ccbq Struct Reference	24
6.7	cam_devq Struct Reference	26
6.8	cam_eb Struct Reference	28
6.9	cam_ed Struct Reference	29
6.10	cam_et Struct Reference	30
6.11	cam_path Struct Reference	31

6.12	cam_periph Struct Reference	33
6.13	cam_periph_map_info Struct Reference	37
6.14	cam_pinfo Struct Reference	38
6.15	cam_sim Struct Reference	39
6.16	cam_status_entry Struct Reference	42
6.17	camq Struct Reference	43
6.18	camq_entry Union Reference	45
6.19	ccb Union Reference	46
6.20	ccb_abort Struct Reference	52
6.21	ccb_accept_tio Struct Reference	53
6.22	ccb_calc_geometry Struct Reference	55
6.23	ccb_debug Struct Reference	57
6.24	ccb_dev_match Struct Reference	58
6.25	ccb_dev_position Struct Reference	60
6.26	ccb_dm_cookie Struct Reference	61
6.27	ccb_en_lun Struct Reference	63
6.28	ccb_eng_exec Struct Reference	65
6.29	ccb_eng_inq Struct Reference	68
6.30	ccb_getdev Struct Reference	70
6.31	ccb_getdevlist Struct Reference	72
6.32	ccb_getdevstats Struct Reference	74
6.33	ccb_hdr Struct Reference	76
6.34	ccb_immed_notify Struct Reference	80
6.35	ccb_notify_ack Struct Reference	82
6.36	ccb_pathinq Struct Reference	83
6.37	ccb_pathinq_settings_fc Struct Reference	88
6.38	ccb_pathinq_settings_sas Struct Reference	89
6.39	ccb_pathinq_settings_spi Struct Reference	90
6.40	ccb_pathstats Struct Reference	91
6.41	ccb_priv_area Union Reference	92
6.42	ccb_priv_entry Union Reference	93
6.43	ccb_relsim Struct Reference	94
6.44	ccb_rescan Struct Reference	96
6.45	ccb_resetbus Struct Reference	97
6.46	ccb_resetdev Struct Reference	98
6.47	ccb_scsiio Struct Reference	99

6.48	ccb_setasync Struct Reference	103
6.49	ccb_setdev Struct Reference	105
6.50	ccb_spriv_area Union Reference	106
6.51	ccb_termio Struct Reference	107
6.52	ccb_trans_settings Struct Reference	108
6.53	ccb_trans_settings_fc Struct Reference	111
6.54	ccb_trans_settings_sas Struct Reference	112
6.55	ccb_trans_settings_scsi Struct Reference	113
6.56	ccb_trans_settings_spi Struct Reference	114
6.57	cd_audio_page Struct Reference	116
6.58	cd_audio_page::port_control Struct Reference	118
6.59	cd_mode_data Struct Reference	119
6.60	cd_mode_data_10 Struct Reference	120
6.61	cd_mode_data_6_10 Union Reference	121
6.62	cd_mode_params Struct Reference	122
6.63	cd_page_sizes Struct Reference	123
6.64	cd_pages Union Reference	124
6.65	cd_params Struct Reference	125
6.66	cd_quirk_entry Struct Reference	126
6.67	cd_softc Struct Reference	127
6.68	cd_toc_single Struct Reference	129
6.69	cd_tocdata Struct Reference	130
6.70	cdb_t Union Reference	131
6.71	cdchanger Struct Reference	132
6.72	ch_softc Struct Reference	134
6.73	da_quirk_entry Struct Reference	137
6.74	da_softc Struct Reference	138
6.75	dev_match_pattern Struct Reference	139
6.76	dev_match_result Struct Reference	140
6.77	device_match_pattern Struct Reference	141
6.78	device_match_result Struct Reference	143
6.79	disk_pages Union Reference	145
6.80	disk_pages::flexible_disk_page Struct Reference	146
6.81	disk_pages::format_device_page Struct Reference	150
6.82	disk_pages::rigid_geometry_page Struct Reference	153
6.83	disk_params Struct Reference	156

6.84 encobj Struct Reference	157
6.85 encvec Struct Reference	158
6.86 format_capacity_descriptor Struct Reference	159
6.87 format_capacity_list_header Struct Reference	160
6.88 format_defect_list_header Struct Reference	161
6.89 format_ipat_descriptor Struct Reference	162
6.90 ioc_enable_lun Struct Reference	163
6.91 lun_info Struct Reference	164
6.92 lun_info::scsi_low_inq_data Struct Reference	168
6.93 lun_info::scsi_low_mode_sense_data Struct Reference	170
6.94 match_pattern Union Reference	172
6.95 match_result Union Reference	173
6.96 op_table_entry Struct Reference	174
6.97 page_device_capabilities Struct Reference	175
6.98 page_element_address_assignment Struct Reference	177
6.99 page_transport_geometry_parameters Struct Reference	179
6.100pass_softc Struct Reference	180
6.101periph_driver Struct Reference	182
6.102periph_match_pattern Struct Reference	183
6.103periph_match_result Struct Reference	184
6.104probe_softc Struct Reference	185
6.105pt_softc Struct Reference	186
6.106read_dvd_struct_data_list Struct Reference	187
6.107read_dvd_struct_list_entry Struct Reference	188
6.108read_dvd_struct_write_prot Struct Reference	189
6.109read_element_status_descriptor Struct Reference	190
6.110read_element_status_header Struct Reference	192
6.111read_element_status_page_header Struct Reference	193
6.112sa_comp_t Union Reference	194
6.113sa_devs Struct Reference	196
6.114sa_devs::sa_mode_devs Struct Reference	197
6.115sa_quirk_entry Struct Reference	198
6.116sa_softc Struct Reference	199
6.117sc_p Struct Reference	205
6.118scfg Struct Reference	207
6.119scsi_changedef Struct Reference	209

6.120	scsi_control_page Struct Reference	211
6.121	scsi_da_rw_recovery_page Struct Reference	213
6.122	scsi_data_compression_page Struct Reference	215
6.123	scsi_defect_desc_block Struct Reference	217
6.124	scsi_defect_desc_bytes_from_index Struct Reference	218
6.125	scsi_defect_desc_phys_sector Struct Reference	219
6.126	scsi_dev_conf_page Struct Reference	220
6.127	scsi_erase Struct Reference	222
6.128	scsi_exchange_medium Struct Reference	223
6.129	scsi_format_unit Struct Reference	225
6.130	scsi_generic Struct Reference	226
6.131	scsi_initialize_element_status Struct Reference	227
6.132	scsi_inquiry Struct Reference	228
6.133	scsi_inquiry_data Struct Reference	230
6.134	scsi_inquiry_pattern Struct Reference	234
6.135	scsi_load_unload Struct Reference	235
6.136	scsi_log_header Struct Reference	236
6.137	scsi_log_param_header Struct Reference	237
6.138	scsi_log_select Struct Reference	238
6.139	scsi_log_sense Struct Reference	239
6.140	scsi_low_error_code Struct Reference	241
6.141	scsi_low_funcs Struct Reference	242
6.142	scsi_low_msg_log Struct Reference	244
6.143	scsi_low_msgin_data Struct Reference	245
6.144	scsi_low_msgout_data Struct Reference	246
6.145	scsi_low_osdep_funcs Struct Reference	247
6.146	scsi_low_osdep_interface Struct Reference	248
6.147	scsi_low_osdep_lun_interface Struct Reference	250
6.148	scsi_low_osdep_targ_interface Struct Reference	251
6.149	scsi_low_softc Struct Reference	252
6.150	scsi_low_statics Struct Reference	257
6.151	scsi_mode_blk_desc Struct Reference	258
6.152	scsi_mode_block_descr Struct Reference	259
6.153	scsi_mode_hdr_10 Struct Reference	260
6.154	scsi_mode_hdr_6 Struct Reference	261
6.155	scsi_mode_header_10 Struct Reference	262

6.156	scsi_mode_header_6 Struct Reference	263
6.157	scsi_mode_page_header Struct Reference	264
6.158	scsi_mode_select_10 Struct Reference	265
6.159	scsi_mode_select_6 Struct Reference	266
6.160	scsi_mode_sense_10 Struct Reference	267
6.161	scsi_mode_sense_6 Struct Reference	268
6.162	scsi_mode_sense_data Struct Reference	270
6.163	scsi_move_medium Struct Reference	272
6.164	scsi_op_quirk_entry Struct Reference	274
6.165	scsi_pause Struct Reference	275
6.166	scsi_play_10 Struct Reference	276
6.167	scsi_play_12 Struct Reference	277
6.168	scsi_play_msf Struct Reference	278
6.169	scsi_play_rel_12 Struct Reference	280
6.170	scsi_play_track Struct Reference	281
6.171	scsi_position_to_element Struct Reference	283
6.172	scsi_prevent Struct Reference	285
6.173	scsi_read_block_limits Struct Reference	286
6.174	scsi_read_block_limits_data Struct Reference	287
6.175	scsi_read_buffer Struct Reference	288
6.176	scsi_read_capacity Struct Reference	289
6.177	scsi_read_capacity_16 Struct Reference	290
6.178	scsi_read_capacity_data Struct Reference	291
6.179	scsi_read_capacity_data_long Struct Reference	292
6.180	scsi_read_cd_cap_data Struct Reference	293
6.181	scsi_read_cd_capacity Struct Reference	295
6.182	scsi_read_defect_data_10 Struct Reference	297
6.183	scsi_read_defect_data_12 Struct Reference	298
6.184	scsi_read_defect_data_hdr_10 Struct Reference	299
6.185	scsi_read_defect_data_hdr_12 Struct Reference	300
6.186	scsi_read_dvd_struct_data_bca Struct Reference	301
6.187	scsi_read_dvd_struct_data_copy_manage Struct Reference	302
6.188	scsi_read_dvd_struct_data_copyright Struct Reference	303
6.189	scsi_read_dvd_struct_data_dcb Struct Reference	304
6.190	scsi_read_dvd_struct_data_dds Struct Reference	305
6.191	scsi_read_dvd_struct_data_disc_id Struct Reference	306

6.192scsi_read_dvd_struct_data_disc_key Struct Reference	308
6.193scsi_read_dvd_struct_data_disc_key_blk Struct Reference	309
6.194scsi_read_dvd_struct_data_generic_dcb Struct Reference	310
6.195scsi_read_dvd_struct_data_header Struct Reference	311
6.196scsi_read_dvd_struct_data_layer_desc Struct Reference	312
6.197scsi_read_dvd_struct_data_lead_in Struct Reference	314
6.198scsi_read_dvd_struct_data_manufacturer Struct Reference	317
6.199scsi_read_dvd_struct_data_medium_status Struct Reference	318
6.200scsi_read_dvd_struct_data_physical Struct Reference	319
6.201scsi_read_dvd_struct_data_prot_discid Struct Reference	320
6.202scsi_read_dvd_struct_data_rmd Struct Reference	321
6.203scsi_read_dvd_struct_data_rmd_borderout Struct Reference	322
6.204scsi_read_dvd_struct_data_spare_area Struct Reference	323
6.205scsi_read_dvd_structure Struct Reference	324
6.206scsi_read_element_status Struct Reference	326
6.207scsi_read_format_capacities Struct Reference	328
6.208scsi_read_header Struct Reference	329
6.209scsi_read_subchannel Struct Reference	330
6.210scsi_read_toc Struct Reference	332
6.211scsi_reassign_blocks Struct Reference	333
6.212scsi_reassign_blocks_data Struct Reference	334
6.213scsi_release Struct Reference	335
6.214scsi_report_key Struct Reference	336
6.215scsi_report_key_data_agid Struct Reference	338
6.216scsi_report_key_data_asf Struct Reference	339
6.217scsi_report_key_data_challenge Struct Reference	340
6.218scsi_report_key_data_header Struct Reference	341
6.219scsi_report_key_data_key1_key2 Struct Reference	342
6.220scsi_report_key_data_rpc Struct Reference	343
6.221scsi_report_key_data_title Struct Reference	345
6.222scsi_report_luns Struct Reference	346
6.223scsi_report_luns_data Struct Reference	348
6.224scsi_request_sense Struct Reference	349
6.225scsi_request_volume_element_address Struct Reference	350
6.226scsi_reserve Struct Reference	352
6.227scsi_reserve_release_unit Struct Reference	353

6.228	scsi_rewind Struct Reference	354
6.229	scsi_rezero_unit Struct Reference	355
6.230	scsi_rw_10 Struct Reference	356
6.231	scsi_rw_12 Struct Reference	358
6.232	scsi_rw_16 Struct Reference	360
6.233	scsi_rw_6 Struct Reference	362
6.234	scsi_sa_rw Struct Reference	363
6.235	scsi_send_diag Struct Reference	364
6.236	scsi_send_key Struct Reference	365
6.237	scsi_send_key_data_rpc Struct Reference	366
6.238	scsi_send_receive Struct Reference	367
6.239	scsi_send_volume_tag Struct Reference	368
6.240	scsi_send_volume_tag_parameters Struct Reference	370
6.241	scsi_sense Struct Reference	371
6.242	scsi_sense_data Struct Reference	372
6.243	scsi_sense_quirk_entry Struct Reference	374
6.244	scsi_set_speed Struct Reference	376
6.245	scsi_space Struct Reference	377
6.246	scsi_start_stop_unit Struct Reference	378
6.247	scsi_static_inquiry_pattern Struct Reference	379
6.248	scsi_status_iu_header Struct Reference	380
6.249	scsi_sync_cache Struct Reference	381
6.250	scsi_tape_locate Struct Reference	382
6.251	scsi_tape_position_data Struct Reference	384
6.252	scsi_tape_read_position Struct Reference	386
6.253	scsi_test_unit_ready Struct Reference	387
6.254	scsi_verify Struct Reference	388
6.255	scsi_vpd_unit_serial_number Struct Reference	389
6.256	scsi_write_and_verify Struct Reference	390
6.257	scsi_write_buffer Struct Reference	391
6.258	scsi_write_filemarks Struct Reference	392
6.259	sense_key_table_entry Struct Reference	393
6.260	sense_t Union Reference	394
6.261	ses_hlptxt Union Reference	395
6.262	ses_object Struct Reference	396
6.263	ses_objstat Struct Reference	397

6.264	ses_softc Struct Reference	398
6.265	SesCfgHdr Struct Reference	400
6.266	SesComStat Struct Reference	401
6.267	SesEncDesc Struct Reference	402
6.268	SesEncHdr Struct Reference	403
6.269	SesThdr Struct Reference	404
6.270	slccb Struct Reference	405
6.271	sscfg Struct Reference	409
6.272	targ_cmd_descr Struct Reference	410
6.273	targ_info Struct Reference	411
6.274	targ_info::synch Struct Reference	416
6.275	targ_softc Struct Reference	417
6.276	targbh_cmd_desc Struct Reference	419
6.277	targbh_softc Struct Reference	421
6.278	typidx Struct Reference	423
6.279	volume_tag Struct Reference	424
6.280	xpt_quirk_entry Struct Reference	425
6.281	xpt_scan_bus_info Struct Reference	426
6.282	xpt_softc Struct Reference	427
6.283	xpt_traverse_config Struct Reference	428
7	FreeBSD kernel CAM code File Documentation	429
7.1	notreviewed.dox File Reference	429
7.2	/usr/src/sys/cam/cam.c File Reference	430
7.3	/usr/src/sys/cam/cam.h File Reference	434
7.4	/usr/src/sys/cam/cam_ccb.h File Reference	443
7.5	/usr/src/sys/cam/cam_debug.h File Reference	463
7.6	/usr/src/sys/cam/cam_periph.c File Reference	465
7.7	/usr/src/sys/cam/cam_periph.h File Reference	480
7.8	/usr/src/sys/cam/cam_queue.c File Reference	493
7.9	/usr/src/sys/cam/cam_queue.h File Reference	500
7.10	/usr/src/sys/cam/cam_sim.c File Reference	508
7.11	/usr/src/sys/cam/cam_sim.h File Reference	511
7.12	/usr/src/sys/cam/cam_xpt.c File Reference	515
7.13	/usr/src/sys/cam/cam_xpt.h File Reference	572
7.14	/usr/src/sys/cam/cam_xpt_periph.h File Reference	579
7.15	/usr/src/sys/cam/cam_xpt_sim.h File Reference	584

7.16	/usr/src/sys/cam/scsi/scsi_all.c File Reference	588
7.17	/usr/src/sys/cam/scsi/scsi_all.h File Reference	608
7.18	/usr/src/sys/cam/scsi/scsi_cd.c File Reference	658
7.19	/usr/src/sys/cam/scsi/scsi_cd.h File Reference	695
7.20	/usr/src/sys/cam/scsi/scsi_ch.c File Reference	718
7.21	/usr/src/sys/cam/scsi/scsi_ch.h File Reference	741
7.22	/usr/src/sys/cam/scsi/scsi_da.c File Reference	752
7.23	/usr/src/sys/cam/scsi/scsi_da.h File Reference	773
7.24	/usr/src/sys/cam/scsi/scsi_dvcfg.h File Reference	784
7.25	/usr/src/sys/cam/scsi/scsi_iu.h File Reference	786
7.26	/usr/src/sys/cam/scsi/scsi_low.c File Reference	789
7.27	/usr/src/sys/cam/scsi/scsi_low.h File Reference	830
7.28	/usr/src/sys/cam/scsi/scsi_low_pisa.c File Reference	867
7.29	/usr/src/sys/cam/scsi/scsi_low_pisa.h File Reference	869
7.30	/usr/src/sys/cam/scsi/scsi_message.h File Reference	870
7.31	/usr/src/sys/cam/scsi/scsi_pass.c File Reference	877
7.32	/usr/src/sys/cam/scsi/scsi_pass.h File Reference	888
7.33	/usr/src/sys/cam/scsi/scsi_pt.c File Reference	889
7.34	/usr/src/sys/cam/scsi/scsi_pt.h File Reference	901
7.35	/usr/src/sys/cam/scsi/scsi_sa.c File Reference	902
7.36	/usr/src/sys/cam/scsi/scsi_sa.h File Reference	935
7.37	/usr/src/sys/cam/scsi/scsi_ses.c File Reference	952
7.38	/usr/src/sys/cam/scsi/scsi_ses.h File Reference	985
7.39	/usr/src/sys/cam/scsi/scsi_targ_bh.c File Reference	993
7.40	/usr/src/sys/cam/scsi/scsi_target.c File Reference	1004
7.41	/usr/src/sys/cam/scsi/scsi_targetio.h File Reference	1019

Chapter 1

FreeBSD kernel CAM code Main Page

IMPORTANT: This API documentation may contain both functions which are public and functions that are for internal use only. Since we have not reviewed every part of the documentation yet, *some internal functions are not marked as such*. Until we finish reviewing the API documentation and add appropriate comments to functions which are only for internal use, you should take this into account. In case you want to use a function of this kernel subsystem in another kernel subsystem you should search for precedence of use outside this subsystem. If the function is not used outside this subsystem you should ask on the mailinglists about it, else you risk breaking something.

Chapter 2

FreeBSD kernel CAM code Directory Hierarchy

2.1 FreeBSD kernel CAM code Directories

This directory hierarchy is sorted roughly, but not completely, alphabetically:

usr	18
src	16
sys	17
cam	13
scsi	15

Chapter 3

FreeBSD kernel CAM code Data Structure Index

3.1 FreeBSD kernel CAM code Data Structures

Here are the data structures with brief descriptions:

asc_key	19
asc_table_entry	20
async_node	21
bus_match_pattern	22
bus_match_result	23
cam_ccbq	24
cam_devq	26
cam_eb	28
cam_ed	29
cam_et	30
cam_path	31
cam_periph	33
cam_periph_map_info	37
cam_pinfo	38
cam_sim	39
cam_status_entry	42
camq	43
camq_entry	45
ccb	46
ccb_abort	52
ccb_accept_tio	53
ccb_calc_geometry	55
ccb_debug	57
ccb_dev_match	58
ccb_dev_position	60
ccb_dm_cookie	61
ccb_en_lun	63
ccb_eng_exec	65
ccb_eng_inq	68
ccb_getdev	70
ccb_getdevlist	72

ccb_getdevstats	74
ccb_hdr	76
ccb_immed_notify	80
ccb_notify_ack	82
ccb_pathinq	83
ccb_pathinq_settings_fc	88
ccb_pathinq_settings_sas	89
ccb_pathinq_settings_spi	90
ccb_pathstats	91
ccb_priv_area	92
ccb_priv_entry	93
ccb_relsim	94
ccb_rescan	96
ccb_resetbus	97
ccb_resetdev	98
ccb_scsiio	99
ccb_setasync	103
ccb_setdev	105
ccb_priv_area	106
ccb_termio	107
ccb_trans_settings	108
ccb_trans_settings_fc	111
ccb_trans_settings_sas	112
ccb_trans_settings_scsi	113
ccb_trans_settings_spi	114
cd_audio_page	116
cd_audio_page::port_control	118
cd_mode_data	119
cd_mode_data_10	120
cd_mode_data_6_10	121
cd_mode_params	122
cd_page_sizes	123
cd_pages	124
cd_params	125
cd_quirk_entry	126
cd_softc	127
cd_toc_single	129
cd_tocdata	130
cdb_t	131
cdchanger	132
ch_softc	134
da_quirk_entry	137
da_softc	138
dev_match_pattern	139
dev_match_result	140
device_match_pattern	141
device_match_result	143
disk_pages	145
disk_pages::flexible_disk_page	146
disk_pages::format_device_page	150
disk_pages::rigid_geometry_page	153
disk_params	156
encobj	157
encvec	158

format_capacity_descriptor	159
format_capacity_list_header	160
format_defect_list_header	161
format_ipat_descriptor	162
ioc_enable_lun	163
lun_info	164
lun_info::scsi_low_inq_data	168
lun_info::scsi_low_mode_sense_data	170
match_pattern	172
match_result	173
op_table_entry	174
page_device_capabilities	175
page_element_address_assignment	177
page_transport_geometry_parameters	179
pass_softc	180
periph_driver	182
periph_match_pattern	183
periph_match_result	184
probe_softc	185
pt_softc	186
read_dvd_struct_data_list	187
read_dvd_struct_list_entry	188
read_dvd_struct_write_prot	189
read_element_status_descriptor	190
read_element_status_header	192
read_element_status_page_header	193
sa_comp_t	194
sa_devs	196
sa_devs::sa_mode_devs	197
sa_quirk_entry	198
sa_softc	199
sc_p	205
scfg	207
scsi_changedef	209
scsi_control_page	211
scsi_da_rw_recovery_page	213
scsi_data_compression_page	215
scsi_defect_desc_block	217
scsi_defect_desc_bytes_from_index	218
scsi_defect_desc_phys_sector	219
scsi_dev_conf_page	220
scsi_erase	222
scsi_exchange_medium	223
scsi_format_unit	225
scsi_generic	226
scsi_initialize_element_status	227
scsi_inquiry	228
scsi_inquiry_data	230
scsi_inquiry_pattern	234
scsi_load_unload	235
scsi_log_header	236
scsi_log_param_header	237
scsi_log_select	238
scsi_log_sense	239

scsi_low_error_code	241
scsi_low_funcs	242
scsi_low_msg_log	244
scsi_low_msgin_data	245
scsi_low_msgout_data	246
scsi_low_osdep_funcs	247
scsi_low_osdep_interface	248
scsi_low_osdep_lun_interface	250
scsi_low_osdep_targ_interface	251
scsi_low_softc	252
scsi_low_statics	257
scsi_mode_blk_desc	258
scsi_mode_block_descr	259
scsi_mode_hdr_10	260
scsi_mode_hdr_6	261
scsi_mode_header_10	262
scsi_mode_header_6	263
scsi_mode_page_header	264
scsi_mode_select_10	265
scsi_mode_select_6	266
scsi_mode_sense_10	267
scsi_mode_sense_6	268
scsi_mode_sense_data	270
scsi_move_medium	272
scsi_op_quirk_entry	274
scsi_pause	275
scsi_play_10	276
scsi_play_12	277
scsi_play_msf	278
scsi_play_rel_12	280
scsi_play_track	281
scsi_position_to_element	283
scsi_prevent	285
scsi_read_block_limits	286
scsi_read_block_limits_data	287
scsi_read_buffer	288
scsi_read_capacity	289
scsi_read_capacity_16	290
scsi_read_capacity_data	291
scsi_read_capacity_data_long	292
scsi_read_cd_cap_data	293
scsi_read_cd_capacity	295
scsi_read_defect_data_10	297
scsi_read_defect_data_12	298
scsi_read_defect_data_hdr_10	299
scsi_read_defect_data_hdr_12	300
scsi_read_dvd_struct_data_bca	301
scsi_read_dvd_struct_data_copy_manage	302
scsi_read_dvd_struct_data_copyright	303
scsi_read_dvd_struct_data_dcb	304
scsi_read_dvd_struct_data_dds	305
scsi_read_dvd_struct_data_disc_id	306
scsi_read_dvd_struct_data_disc_key	308
scsi_read_dvd_struct_data_disc_key_blk	309

scsi_read_dvd_struct_data_generic_dcb	310
scsi_read_dvd_struct_data_header	311
scsi_read_dvd_struct_data_layer_desc	312
scsi_read_dvd_struct_data_lead_in	314
scsi_read_dvd_struct_data_manufacturer	317
scsi_read_dvd_struct_data_medium_status	318
scsi_read_dvd_struct_data_physical	319
scsi_read_dvd_struct_data_prot_discid	320
scsi_read_dvd_struct_data_rmd	321
scsi_read_dvd_struct_data_rmd_borderout	322
scsi_read_dvd_struct_data_spare_area	323
scsi_read_dvd_structure	324
scsi_read_element_status	326
scsi_read_format_capacities	328
scsi_read_header	329
scsi_read_subchannel	330
scsi_read_toc	332
scsi_reassign_blocks	333
scsi_reassign_blocks_data	334
scsi_release	335
scsi_report_key	336
scsi_report_key_data_agid	338
scsi_report_key_data_asf	339
scsi_report_key_data_challenge	340
scsi_report_key_data_header	341
scsi_report_key_data_key1_key2	342
scsi_report_key_data_rpc	343
scsi_report_key_data_title	345
scsi_report_luns	346
scsi_report_luns_data	348
scsi_request_sense	349
scsi_request_volume_element_address	350
scsi_reserve	352
scsi_reserve_release_unit	353
scsi_rewind	354
scsi_rezero_unit	355
scsi_rw_10	356
scsi_rw_12	358
scsi_rw_16	360
scsi_rw_6	362
scsi_sa_rw	363
scsi_send_diag	364
scsi_send_key	365
scsi_send_key_data_rpc	366
scsi_send_receive	367
scsi_send_volume_tag	368
scsi_send_volume_tag_parameters	370
scsi_sense	371
scsi_sense_data	372
scsi_sense_quirk_entry	374
scsi_set_speed	376
scsi_space	377
scsi_start_stop_unit	378
scsi_static_inquiry_pattern	379

scsi_status_iu_header	380
scsi_sync_cache	381
scsi_tape_locate	382
scsi_tape_position_data	384
scsi_tape_read_position	386
scsi_test_unit_ready	387
scsi_verify	388
scsi_vpd_unit_serial_number	389
scsi_write_and_verify	390
scsi_write_buffer	391
scsi_write_filemarks	392
sense_key_table_entry	393
sense_t	394
ses_hlptxt	395
ses_object	396
ses_objstat	397
ses_softc	398
SesCfgHdr	400
SesComStat	401
SesEncDesc	402
SesEncHdr	403
SesThdr	404
slccb	405
sscfg	409
targ_cmd_descr	410
targ_info	411
targ_info::synch	416
targ_softc	417
targbh_cmd_desc	419
targbh_softc	421
typidx	423
volume_tag	424
xpt_quirk_entry	425
xpt_scan_bus_info	426
xpt_softc	427
xpt_traverse_config	428

Chapter 4

FreeBSD kernel CAM code File Index

4.1 FreeBSD kernel CAM code File List

Here is a list of all files with brief descriptions:

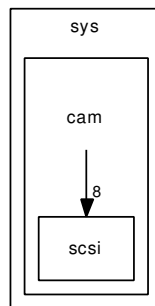
/usr/src/sys/cam/cam.c	430
/usr/src/sys/cam/cam.h	434
/usr/src/sys/cam/cam_ccb.h	443
/usr/src/sys/cam/cam_debug.h	463
/usr/src/sys/cam/cam_periph.c	465
/usr/src/sys/cam/cam_periph.h	480
/usr/src/sys/cam/cam_queue.c	493
/usr/src/sys/cam/cam_queue.h	500
/usr/src/sys/cam/cam_sim.c	508
/usr/src/sys/cam/cam_sim.h	511
/usr/src/sys/cam/cam_xpt.c	515
/usr/src/sys/cam/cam_xpt.h	572
/usr/src/sys/cam/cam_xpt_periph.h	579
/usr/src/sys/cam/cam_xpt_sim.h	584
/usr/src/sys/cam/scsi/scsi_all.c	588
/usr/src/sys/cam/scsi/scsi_all.h	608
/usr/src/sys/cam/scsi/scsi_cd.c	658
/usr/src/sys/cam/scsi/scsi_cd.h	695
/usr/src/sys/cam/scsi/scsi_ch.c	718
/usr/src/sys/cam/scsi/scsi_ch.h	741
/usr/src/sys/cam/scsi/scsi_da.c	752
/usr/src/sys/cam/scsi/scsi_da.h	773
/usr/src/sys/cam/scsi/scsi_dvcfg.h	784
/usr/src/sys/cam/scsi/scsi_iu.h	786
/usr/src/sys/cam/scsi/scsi_low.c	789
/usr/src/sys/cam/scsi/scsi_low.h	830
/usr/src/sys/cam/scsi/scsi_low_pisa.c	867
/usr/src/sys/cam/scsi/scsi_low_pisa.h	869
/usr/src/sys/cam/scsi/scsi_message.h	870
/usr/src/sys/cam/scsi/scsi_pass.c	877
/usr/src/sys/cam/scsi/scsi_pass.h	888
/usr/src/sys/cam/scsi/scsi_pt.c	889
/usr/src/sys/cam/scsi/scsi_pt.h	901

/usr/src/sys/cam/scsi/scsi_sa.c	902
/usr/src/sys/cam/scsi/scsi_sa.h	935
/usr/src/sys/cam/scsi/scsi_ses.c	952
/usr/src/sys/cam/scsi/scsi_ses.h	985
/usr/src/sys/cam/scsi/scsi_targ_bh.c	993
/usr/src/sys/cam/scsi/scsi_target.c	1004
/usr/src/sys/cam/scsi/scsi_targetio.h	1019

Chapter 5

FreeBSD kernel CAM code Directory Documentation

5.1 /usr/src/sys/cam/ Directory Reference



Directories

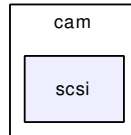
- directory [scsi](#)

Files

- file [cam.c](#)
- file [cam.h](#)
- file [cam_ccb.h](#)
- file [cam_debug.h](#)
- file [cam_periph.c](#)
- file [cam_periph.h](#)
- file [cam_queue.c](#)
- file [cam_queue.h](#)
- file [cam_sim.c](#)
- file [cam_sim.h](#)

- file [cam_xpt.c](#)
- file [cam_xpt.h](#)
- file [cam_xpt_periph.h](#)
- file [cam_xpt_sim.h](#)

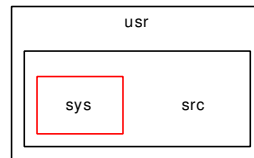
5.2 /usr/src/sys/cam/scsi/ Directory Reference



Files

- file [scsi_all.c](#)
- file [scsi_all.h](#)
- file [scsi_cd.c](#)
- file [scsi_cd.h](#)
- file [scsi_ch.c](#)
- file [scsi_ch.h](#)
- file [scsi_da.c](#)
- file [scsi_da.h](#)
- file [scsi_dvdfg.h](#)
- file [scsi_iu.h](#)
- file [scsi_low.c](#)
- file [scsi_low.h](#)
- file [scsi_low_pisa.c](#)
- file [scsi_low_pisa.h](#)
- file [scsi_message.h](#)
- file [scsi_pass.c](#)
- file [scsi_pass.h](#)
- file [scsi_pt.c](#)
- file [scsi_pt.h](#)
- file [scsi_sa.c](#)
- file [scsi_sa.h](#)
- file [scsi_ses.c](#)
- file [scsi_ses.h](#)
- file [scsi_targ_bh.c](#)
- file [scsi_target.c](#)
- file [scsi_targetio.h](#)

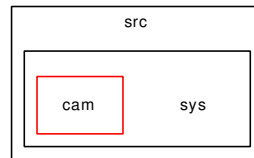
5.3 /usr/src/ Directory Reference



Directories

- directory [sys](#)

5.4 /usr/src/sys/ Directory Reference



Directories

- directory [cam](#)

5.5 /usr/ Directory Reference



Directories

- directory [src](#)

Chapter 6

FreeBSD kernel CAM code Data Structure Documentation

6.1 asc_key Struct Reference

Data Fields

- int [asc](#)
- int [ascq](#)

6.1.1 Detailed Description

Definition at line 1484 of file [scsi_all.c](#).

6.1.2 Field Documentation

6.1.2.1 int [asc_key::asc](#)

Definition at line 1486 of file [scsi_all.c](#).

Referenced by [fetchtableentries\(\)](#).

6.1.2.2 int [asc_key::ascq](#)

Definition at line 1487 of file [scsi_all.c](#).

Referenced by [fetchtableentries\(\)](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_all.c](#)

6.2 `asc_table_entry` Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- `u_int8_t asc`
- `u_int8_t ascq`
- `u_int32_t action`
- `const char * desc`

6.2.1 Detailed Description

Definition at line 878 of file `scsi_all.h`.

6.2.2 Field Documentation

6.2.2.1 `u_int32_t asc_table_entry::action`

Definition at line 881 of file `scsi_all.h`.

Referenced by `ascentrycomp()`, and `scsi_error_action()`.

6.2.2.2 `u_int8_t asc_table_entry::asc`

Definition at line 879 of file `scsi_all.h`.

Referenced by `ascentrycomp()`.

6.2.2.3 `u_int8_t asc_table_entry::ascq`

Definition at line 880 of file `scsi_all.h`.

Referenced by `ascentrycomp()`.

6.2.2.4 `const char* asc_table_entry::desc`

Definition at line 882 of file `scsi_all.h`.

Referenced by `scsi_sense_desc()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.3 `async_node` Struct Reference

6.3.1 Detailed Description

Definition at line 77 of file `cam_xpt.c`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/cam_xpt.c](#)

6.4 bus_match_pattern Struct Reference

```
#include <cam_ccb.h>
```

Data Fields

- [path_id_t path_id](#)
- char [dev_name](#) [DEV_IDLEN]
- [u_int32_t unit_number](#)
- [u_int32_t bus_id](#)
- [bus_pattern_flags flags](#)

6.4.1 Detailed Description

Definition at line 370 of file `cam_ccb.h`.

6.4.2 Field Documentation

6.4.2.1 [u_int32_t bus_match_pattern::bus_id](#)

Definition at line 374 of file `cam_ccb.h`.

6.4.2.2 [char bus_match_pattern::dev_name](#)[DEV_IDLEN]

Definition at line 372 of file `cam_ccb.h`.

6.4.2.3 [bus_pattern_flags bus_match_pattern::flags](#)

Definition at line 375 of file `cam_ccb.h`.

6.4.2.4 [path_id_t bus_match_pattern::path_id](#)

Definition at line 371 of file `cam_ccb.h`.

6.4.2.5 [u_int32_t bus_match_pattern::unit_number](#)

Definition at line 373 of file `cam_ccb.h`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/cam_ccb.h](#)

6.5 bus_match_result Struct Reference

```
#include <cam_ccb.h>
```

Data Fields

- [path_id_t path_id](#)
- char [dev_name](#) [DEV_IDLEN]
- [u_int32_t unit_number](#)
- [u_int32_t bus_id](#)

6.5.1 Detailed Description

Definition at line 416 of file cam_ccb.h.

6.5.2 Field Documentation

6.5.2.1 [u_int32_t bus_match_result::bus_id](#)

Definition at line 420 of file cam_ccb.h.

Referenced by [xptedtbusfunc\(\)](#).

6.5.2.2 [char bus_match_result::dev_name](#)[DEV_IDLEN]

Definition at line 418 of file cam_ccb.h.

Referenced by [xptedtbusfunc\(\)](#).

6.5.2.3 [path_id_t bus_match_result::path_id](#)

Definition at line 417 of file cam_ccb.h.

Referenced by [xptedtbusfunc\(\)](#).

6.5.2.4 [u_int32_t bus_match_result::unit_number](#)

Definition at line 419 of file cam_ccb.h.

Referenced by [xptedtbusfunc\(\)](#).

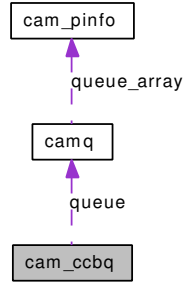
The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/cam_ccb.h](#)

6.6 cam_ccbq Struct Reference

```
#include <cam_queue.h>
```

Collaboration diagram for cam_ccbq:



Data Fields

- [camq queue](#)
- int [devq_openings](#)
- int [dev_openings](#)
- int [dev_active](#)
- int [held](#)
- `ccb_hdr_tailq` [active_ccbs](#)

6.6.1 Detailed Description

Definition at line 57 of file `cam_queue.h`.

6.6.2 Field Documentation

6.6.2.1 struct `ccb_hdr_tailq` `cam_ccbq::active_ccbs`

Definition at line 63 of file `cam_queue.h`.

Referenced by `cam_ccbq_ccb_done()`, and `cam_ccbq_send_ccb()`.

6.6.2.2 int `cam_ccbq::dev_active`

Definition at line 61 of file `cam_queue.h`.

Referenced by `cam_ccbq_ccb_done()`, `cam_ccbq_resize()`, and `cam_ccbq_send_ccb()`.

6.6.2.3 int `cam_ccbq::dev_openings`

Definition at line 60 of file `cam_queue.h`.

Referenced by `cam_ccbq_ccb_done()`, `cam_ccbq_resize()`, and `cam_ccbq_send_ccb()`.

6.6.2.4 int [cam_ccbq::devq_openings](#)

Definition at line 59 of file [cam_queue.h](#).

Referenced by [cam_ccbq_release_opening\(\)](#), [cam_ccbq_resize\(\)](#), and [cam_ccbq_take_opening\(\)](#).

6.6.2.5 int [cam_ccbq::held](#)

Definition at line 62 of file [cam_queue.h](#).

Referenced by [cam_ccbq_ccb_done\(\)](#), [cam_ccbq_insert_ccb\(\)](#), [cam_ccbq_release_opening\(\)](#), [cam_ccbq_resize\(\)](#), and [cam_ccbq_take_opening\(\)](#).

6.6.2.6 struct [camq](#) [cam_ccbq::queue](#)

Definition at line 58 of file [cam_queue.h](#).

Referenced by [cam_ccbq_free\(\)](#), [cam_ccbq_insert_ccb\(\)](#), [cam_ccbq_peek_ccb\(\)](#), [cam_ccbq_pending_ccb_count\(\)](#), [cam_ccbq_remove_ccb\(\)](#), and [cam_ccbq_resize\(\)](#).

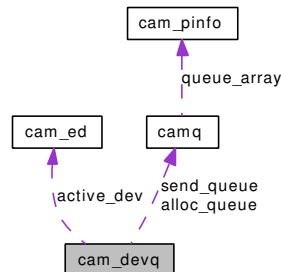
The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/cam_queue.h](#)

6.7 cam_devq Struct Reference

```
#include <cam_queue.h>
```

Collaboration diagram for cam_devq:



Data Fields

- [camq alloc_queue](#)
- [camq send_queue](#)
- [cam_ed * active_dev](#)
- [int alloc_openings](#)
- [int alloc_active](#)
- [int send_openings](#)
- [int send_active](#)

6.7.1 Detailed Description

Definition at line 68 of file cam_queue.h.

6.7.2 Field Documentation

6.7.2.1 struct [cam_ed*](#) [cam_devq::active_dev](#)

Definition at line 71 of file cam_queue.h.

Referenced by [xpt_bus_deregister\(\)](#), [xpt_freeze_simq\(\)](#), and [xpt_run_dev_sendq\(\)](#).

6.7.2.2 int [cam_devq::alloc_active](#)

Definition at line 73 of file cam_queue.h.

Referenced by [xpt_run_dev_allocq\(\)](#).

6.7.2.3 int [cam_devq::alloc_openings](#)

Definition at line 72 of file cam_queue.h.

Referenced by [dev_allocq_is_runnable\(\)](#), and [xpt_run_dev_allocq\(\)](#).

6.7.2.4 struct `camq cam_devq::alloc_queue`

Definition at line 69 of file `cam_queue.h`.

Referenced by `cam_devq_free()`, `cam_devq_resize()`, `dev_allocq_is_runnable()`, `xpt_alloc_device()`, `xpt_bus_deregister()`, `xpt_release_device()`, and `xpt_run_dev_allocq()`.

6.7.2.5 int `cam_devq::send_active`

Definition at line 75 of file `cam_queue.h`.

Referenced by `xpt_run_dev_sendq()`.

6.7.2.6 int `cam_devq::send_openings`

Definition at line 74 of file `cam_queue.h`.

Referenced by `xpt_polled_action()`, and `xpt_run_dev_sendq()`.

6.7.2.7 struct `camq cam_devq::send_queue`

Definition at line 70 of file `cam_queue.h`.

Referenced by `cam_devq_free()`, `cam_devq_resize()`, `xpt_bus_deregister()`, `xpt_freeze_simq()`, `xpt_release_simq()`, and `xpt_run_dev_sendq()`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/cam_queue.h](#)

6.8 `cam_eb` Struct Reference

6.8.1 Detailed Description

Definition at line 185 of file `cam_xpt.c`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/cam_xpt.c](#)

6.9 cam_ed Struct Reference

6.9.1 Detailed Description

Definition at line 115 of file cam_xpt.c.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/cam_xpt.c](#)

6.10 `cam_et` Struct Reference

6.10.1 Detailed Description

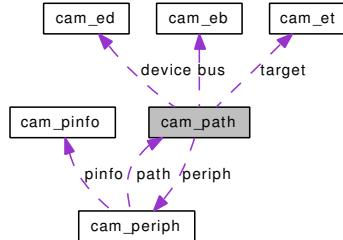
Definition at line 170 of file `cam_xpt.c`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/cam_xpt.c](#)

6.11 cam_path Struct Reference

Collaboration diagram for cam_path:



Data Fields

- `cam_periph * periph`
- `cam_eb * bus`
- `cam_et * target`
- `cam_ed * device`

6.11.1 Detailed Description

Definition at line 197 of file cam_xpt.c.

6.11.2 Field Documentation

6.11.2.1 struct `cam_eb`* `cam_path::bus`

Definition at line 199 of file cam_xpt.c.

Referenced by `camisr()`, `proberegister()`, `xpt_action()`, `xpt_announce_periph()`, `xpt_async()`, `xpt_bus_deregister()`, `xpt_compile_path()`, `xpt_dev_async()`, `xpt_path_comp()`, `xpt_path_path_id()`, `xpt_path_sim()`, `xpt_path_string()`, `xpt_polled_action()`, `xpt_print_path()`, `xpt_release_ccb()`, `xpt_release_path()`, `xpt_run_dev_sendq()`, `xpt_scan_lun()`, `xpt_schedule()`, `xpt_set_transfer_settings()`, `xpt_setup_ccb()`, `xpt_start_tags()`, `xptedtperiphfunc()`, `xptperiphmatch()`, and `xptplistperiphfunc()`.

6.11.2.2 struct `cam_ed`* `cam_path::device`

Definition at line 201 of file cam_xpt.c.

Referenced by `camisr()`, `probedone()`, `probestart()`, `xpt_action()`, `xpt_add_periph()`, `xpt_announce_periph()`, `xpt_async()`, `xpt_compile_path()`, `xpt_dev_async()`, `xpt_dev_ccbq_resize()`, `xpt_devise_transport()`, `xpt_freeze_devq()`, `xpt_path_comp()`, `xpt_path_lun_id()`, `xpt_path_string()`, `xpt_polled_action()`, `xpt_print_path()`, `xpt_release_ccb()`, `xpt_release_devq()`, `xpt_release_path()`, `xpt_remove_periph()`, `xpt_scan_bus()`, `xpt_scan_lun()`, `xpt_schedule()`, `xpt_setup_ccb()`, `xpt_start_tags()`, `xpt_toggle_tags()`, `xptedtperiphfunc()`, `xptperiphmatch()`, and `xptplistperiphfunc()`.

6.11.2.3 struct `cam_periph`* `cam_path::periph`

Definition at line 198 of file cam_xpt.c.

Referenced by `camisr()`, `probedone()`, `xpt_action()`, `xpt_announce_periph()`, `xpt_compile_path()`, `xpt_done()`, `xpt_path_periph()`, `xpt_path_string()`, and `xpt_print_path()`.

6.11.2.4 struct `cam_et*` `cam_path::target`

Definition at line 200 of file `cam_xpt.c`.

Referenced by `xpt_action()`, `xpt_announce_periph()`, `xpt_async()`, `xpt_compile_path()`, `xpt_dev_async()`, `xpt_devise_transport()`, `xpt_path_comp()`, `xpt_path_string()`, `xpt_path_target_id()`, `xpt_print_path()`, `xpt_release_path()`, `xpt_scan_bus()`, `xpt_scan_lun()`, `xpt_setup_ccb()`, `xptedtperiphfunc()`, `xptperiphmatch()`, and `xptplistperiphfunc()`.

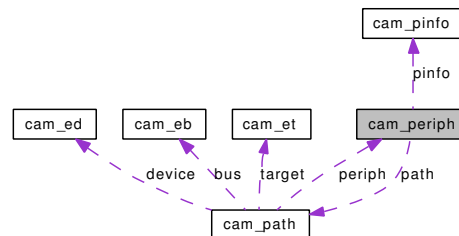
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/cam_xpt.c`

6.12 cam_periph Struct Reference

```
#include <cam_periph.h>
```

Collaboration diagram for cam_periph:



Public Member Functions

- [SLIST_HEAD](#) (, [ccb_hdr](#)) [ccb_list](#)
- [SLIST_ENTRY](#) ([cam_periph](#)) [periph_links](#)
- [TAILQ_ENTRY](#) ([cam_periph](#)) [unit_links](#)

Data Fields

- [cam_pinfo](#) [pinfo](#)
- [periph_start_t](#) * [periph_start](#)
- [periph_oninv_t](#) * [periph_oninval](#)
- [periph_dtor_t](#) * [periph_dtor](#)
- [char](#) * [periph_name](#)
- [cam_path](#) * [path](#)
- [void](#) * [softc](#)
- [u_int32_t](#) [unit_number](#)
- [cam_periph_type](#) [type](#)
- [u_int32_t](#) [flags](#)
- [u_int32_t](#) [immediate_priority](#)
- [u_int32_t](#) [refcount](#)
- [ac_callback_t](#) * [deferred_callback](#)
- [ac_code](#) [deferred_ac](#)

6.12.1 Detailed Description

Definition at line 99 of file [cam_periph.h](#).

6.12.2 Member Function Documentation

6.12.2.1 `cam_periph::SLIST_ENTRY` ([cam_periph](#))

6.12.2.2 `cam_periph::SLIST_HEAD` ([ccb_hdr](#))

6.12.2.3 `cam_periph::TAILQ_ENTRY` ([cam_periph](#))

6.12.3 Field Documentation

6.12.3.1 `ac_code` [cam_periph::deferred_ac](#)

Definition at line 122 of file `cam_periph.h`.

Referenced by `cam_periph_alloc()`.

6.12.3.2 `ac_callback_t*` [cam_periph::deferred_callback](#)

Definition at line 121 of file `cam_periph.h`.

Referenced by `cam_periph_alloc()`.

6.12.3.3 `u_int32_t` [cam_periph::flags](#)

Definition at line 109 of file `cam_periph.h`.

Referenced by `cam_periph_alloc()`, `cam_periph_invalidate()`, `cam_periph_lock()`, `cam_periph_release()`, `cam_periph_unlock()`, `camperiphdone()`, `camperiphscsisenseerror()`, and `saiocctl()`.

6.12.3.4 `u_int32_t` [cam_periph::immediate_priority](#)

Definition at line 116 of file `cam_periph.h`.

Referenced by `cam_periph_getccb()`, `cdstart()`, `chstart()`, `dastart()`, `passstart()`, `ptstart()`, `sastart()`, `sesstart()`, and `targbhstart()`.

6.12.3.5 `struct cam_path*` [cam_periph::path](#)

Definition at line 105 of file `cam_periph.h`.

Referenced by `cam_periph_alloc()`, `cam_periph_bus_settle()`, `cam_periph_find()`, `cam_periph_freeze_after_event()`, `cam_periph_getccb()`, `camperiphnextunit()`, `cd6byteworkaround()`, `cdcleanup()`, `cddone()`, `cdgetmode()`, `cdioctl()`, `cdoninvalidate()`, `cdopen()`, `cdprevent()`, `cdregister()`, `cdreportkey()`, `cdsize()`, `cdstart()`, `cdstrategy()`, `chcleanup()`, `chdone()`, `chgetelemstatus()`, `chgetparams()`, `chioctl()`, `choninvalidate()`, `chregister()`, `dacleanup()`, `daclose()`, `dadone()`, `dadump()`, `daoninvalidate()`, `daregister()`, `dasetgeom()`, `dashutdown()`, `passcleanup()`, `passioctl()`, `passoninvalidate()`, `passopen()`, `passregister()`, `probedone()`, `proberegister()`, `proberequestbackoff()`, `proberequestdefaultnegotiation()`, `probeschedule()`, `probstart()`, `ptctor()`, `ptdone()`, `ptdtor()`, `ptoninvalidate()`, `ptopen()`, `sacleanup()`, `saclose()`, `sadone()`, `saerror()`, `sagetparams()`, `saiocctl()`, `samount()`, `saoninvalidate()`, `saopen()`, `saprevent()`, `saregister()`, `sasetparams()`, `sastart()`, `sastrategy()`, `sescleanup()`, `sesioctl()`, `sesoninvalidate()`, `sesregister()`, `SLIST_HEAD()`, `targbhdislun()`, `targbhdone()`, `targbhendlun()`, `targbhstart()`, `targctor()`, `targdone()`, `targstart()`, `xpt_add_periph()`, `xpt_announce_periph()`, `xpt_async()`, `xpt_path_periph()`, `xpt_remove_periph()`, `xpt_run_dev_allocq()`, `xpt_scan_lun()`, `xpt_schedule()`, `xptedtperiphfunc()`, `xptioctl()`, `xptperiphmatch()`, and `xptplistperiphfunc()`.

6.12.3.6 `periph_dtor_t*` `cam_periph::periph_dtor`

Definition at line 103 of file `cam_periph.h`.

Referenced by `camperiphfree()`.

6.12.3.7 `char*` `cam_periph::periph_name`

Definition at line 104 of file `cam_periph.h`.

Referenced by `cam_periph_alloc()`, `camperiphfree()`, `camperiphnextunit()`, `camperiphunit()`, `chregister()`, `passregister()`, `ptctor()`, `saregister()`, `ses_log()`, `sesregister()`, `xpt_action()`, `xpt_announce_periph()`, `xpt_path_string()`, `xpt_print_path()`, `xptedtperiphfunc()`, `xptpassannouncefunc()`, `xptperiphmatch()`, and `xptplistperiphfunc()`.

6.12.3.8 `periph_oninv_t*` `cam_periph::periph_oninval`

Definition at line 102 of file `cam_periph.h`.

Referenced by `cam_periph_invalidate()`.

6.12.3.9 `periph_start_t*` `cam_periph::periph_start`

Definition at line 101 of file `cam_periph.h`.

Referenced by `xpt_run_dev_allocq()`, and `xpt_schedule()`.

6.12.3.10 `cam_pinfo` `cam_periph::pinfo`

Definition at line 100 of file `cam_periph.h`.

Referenced by `cdstart()`, `chstart()`, `dastart()`, `periph_is_queued()`, `probeschedule()`, `ptstart()`, `sastart()`, `sesstart()`, `targhstart()`, `xpt_bus_deregister()`, `xpt_run_dev_allocq()`, and `xpt_schedule()`.

6.12.3.11 `u_int32_t` `cam_periph::refcount`

Definition at line 117 of file `cam_periph.h`.

Referenced by `cam_periph_acquire()`, and `cam_periph_release()`.

6.12.3.12 `void*` `cam_periph::softc`

Definition at line 106 of file `cam_periph.h`.

Referenced by `cd6byteworkaround()`, `cdasync()`, `cdcheckmedia()`, `cdcleanup()`, `cdcclose()`, `cddone()`, `cderror()`, `cdgetccb()`, `cdgetmode()`, `cdioctl()`, `cdoninvalidate()`, `cdopen()`, `cdprevent()`, `cdregister()`, `cdrunccb()`, `cdschedule()`, `cdsetmode()`, `cdsize()`, `cdstart()`, `cdstrategy()`, `cdsysclinit()`, `chcleanup()`, `chclose()`, `chdone()`, `cherror()`, `chexchange()`, `chgetelemstatus()`, `chgetparams()`, `chielem()`, `chioctl()`, `chmove()`, `choninvalidate()`, `chopen()`, `chposition()`, `chregister()`, `chsetvoltage()`, `chstart()`, `cmd6workaround()`, `daasync()`, `dacleanup()`, `daclose()`, `dadone()`, `dadump()`, `daerror()`, `dagetcapacity()`, `daoninvalidate()`, `daprevent()`, `daregister()`, `dasetgeom()`, `dashutdown()`, `dastart()`, `dastrategy()`, `dasysclinit()`, `passcleanup()`, `passclose()`, `passdone()`, `passerror()`, `passioctl()`, `passoninvalidate()`, `passopen()`, `passregister()`, `passsendccb()`, `passstart()`,

probecleanup(), probedone(), proberegister(), probeschedule(), probestart(), ptasync(), ptclose(), ptc-
tor(), ptdone(), ptdtor(), pterror(), ptioctl(), ptoninvalidate(), ptopen(), ptstart(), ptstrategy(), sacleanup(),
saclose(), sadone(), saerase(), saerror(), sagetparams(), saioctl(), saloadunload(), samarkswanted(),
samount(), saoninvalidate(), saopen(), saprevent(), sardpos(), saregister(), sareservereleaseunit(), sareten-
sion(), sarewind(), sasetparams(), sasetpos(), saspace(), sastart(), sastrategy(), sawritefilemarks(), sescle-
anup(), sesclose(), seserror(), sesioctl(), sesoninvalidate(), sesopen(), sesregister(), SLIST_HEAD(), targb-
hctor(), targbhdislun(), targbhdone(), targbhdtor(), targbhenlun(), targbhstart(), targctor(), targdone(),
targdtor(), targetnable(), targstart(), xpt_scan_lun(), and xptregister().

6.12.3.13 `cam_periph_type` `cam_periph::type`

Definition at line 108 of file `cam_periph.h`.

Referenced by `xpt_done()`.

6.12.3.14 `u_int32_t` `cam_periph::unit_number`

Definition at line 107 of file `cam_periph.h`.

Referenced by `cam_periph_alloc()`, `camperiphnextunit()`, `cdregister()`, `cdsysctlinit()`, `chregister()`, `dareg-
ister()`, `dasyctlinit()`, `passregister()`, `ptctor()`, `saregister()`, `ses_log()`, `sesregister()`, `SLIST_HEAD()`, `targ-
enable()`, `xpt_action()`, `xpt_announce_periph()`, `xpt_path_string()`, `xpt_print_path()`, `xptedtperiphfunc()`,
`xptioctl()`, `xptperiphmatch()`, and `xptplistperiphfunc()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/cam_periph.h`

6.13 cam_periph_map_info Struct Reference

```
#include <cam_periph.h>
```

Data Fields

- int [num_bufs_used](#)
- buf * [bp](#) [CAM_PERIPH_MAXMAPS]

6.13.1 Detailed Description

Definition at line 127 of file [cam_periph.h](#).

6.13.2 Field Documentation

6.13.2.1 struct buf* [cam_periph_map_info::bp](#)[CAM_PERIPH_MAXMAPS]

Definition at line 129 of file [cam_periph.h](#).

Referenced by [cam_periph_mapmem\(\)](#), and [cam_periph_unmapmem\(\)](#).

6.13.2.2 int [cam_periph_map_info::num_bufs_used](#)

Definition at line 128 of file [cam_periph.h](#).

Referenced by [cam_periph_mapmem\(\)](#), [cam_periph_unmapmem\(\)](#), [targetreturnccb\(\)](#), and [targsendccb\(\)](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/cam_periph.h](#)

6.14 cam_pinfo Struct Reference

```
#include <cam.h>
```

Data Fields

- `u_int32_t` [priority](#)
- `u_int32_t` [generation](#)
- `int` [index](#)

6.14.1 Detailed Description

Definition at line 67 of file `cam.h`.

6.14.2 Field Documentation

6.14.2.1 `u_int32_t` [cam_pinfo::generation](#)

Definition at line 70 of file `cam.h`.

Referenced by `cdgetccb()`, `cdrunchangerqueue()`, `cdschedule()`, `xpt_schedule()`, `xpt_schedule_dev()`, and `xpt_setup_ccb()`.

6.14.2.2 `int` [cam_pinfo::index](#)

Definition at line 71 of file `cam.h`.

Referenced by `cam_ccbq_remove_ccb()`, `cam_ccbq_send_ccb()`, `cam_init_pinfo()`, `cam_periph_ccbwait()`, `camisr()`, `camq_insert()`, `camq_remove()`, `cdgetccb()`, `cdregister()`, `cdschedule()`, `periph_is_queued()`, `swap()`, `xpt_action()`, `xpt_done()`, `xpt_schedule()`, `xpt_schedule_dev()`, and `xpt_setup_ccb()`.

6.14.2.3 `u_int32_t` [cam_pinfo::priority](#)

Definition at line 68 of file `cam.h`.

Referenced by `cam_init_pinfo()`, `camq_change_priority()`, `cdgetccb()`, `cdschedule()`, `cdstart()`, `chstart()`, `dastart()`, `passioctl()`, `probedone()`, `probeschedule()`, `ptstart()`, `sastart()`, `sesstart()`, `targbhstart()`, `xpt_bus_deregister()`, `xpt_run_dev_allocq()`, `xpt_scan_bus()`, `xpt_schedule()`, `xpt_schedule_dev()`, `xpt_setup_ccb()`, and `xptioctl()`.

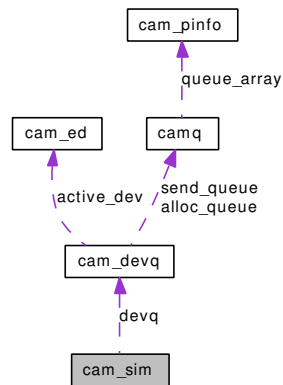
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/cam.h`

6.15 cam_sim Struct Reference

```
#include <cam_sim.h>
```

Collaboration diagram for cam_sim:



Data Fields

- `sim_action_func` `sim_action`
- `sim_poll_func` `sim_poll`
- `const char *` `sim_name`
- `void *` `softc`
- `u_int32_t` `path_id`
- `u_int32_t` `unit_number`
- `u_int32_t` `bus_id`
- `int` `max_tagged_dev_openings`
- `int` `max_dev_openings`
- `u_int32_t` `flags`
- `callout_handle` `c_handle`
- `cam_devq *` `devq`

6.15.1 Detailed Description

Definition at line 88 of file `cam_sim.h`.

6.15.2 Field Documentation

6.15.2.1 `u_int32_t` `cam_sim::bus_id`

Definition at line 95 of file `cam_sim.h`.

Referenced by `cam_sim_alloc()`, `cam_sim_bus()`, `xpt_bus_register()`, and `xptaction()`.

6.15.2.2 `struct callout_handle` `cam_sim::c_handle`

Definition at line 100 of file `cam_sim.h`.

Referenced by `cam_sim_alloc()`, and `xpt_release_simq()`.

6.15.2.3 struct `cam_devq*` `cam_sim::devq`

Definition at line 101 of file `cam_sim.h`.

Referenced by `cam_sim_alloc()`, `cam_sim_free()`, `xpt_freeze_simq()`, `xpt_polled_action()`, `xpt_release_simq()`, and `xpt_run_dev_sendq()`.

6.15.2.4 `u_int32_t` `cam_sim::flags`

Definition at line 98 of file `cam_sim.h`.

Referenced by `cam_sim_alloc()`, and `xpt_release_simq()`.

6.15.2.5 `int` `cam_sim::max_dev_openings`

Definition at line 97 of file `cam_sim.h`.

Referenced by `cam_sim_alloc()`, and `xpt_set_transfer_settings()`.

6.15.2.6 `int` `cam_sim::max_tagged_dev_openings`

Definition at line 96 of file `cam_sim.h`.

Referenced by `cam_sim_alloc()`, and `xpt_start_tags()`.

6.15.2.7 `u_int32_t` `cam_sim::path_id`

Definition at line 93 of file `cam_sim.h`.

Referenced by `cam_sim_alloc()`, `cam_sim_path()`, `cam_sim_set_path()`, `xpt_bus_register()`, and `xpt_release_simq()`.

6.15.2.8 `sim_action_func` `cam_sim::sim_action`

Definition at line 89 of file `cam_sim.h`.

Referenced by `cam_sim_alloc()`, `xpt_action()`, `xpt_bus_deregister()`, `xpt_run_dev_sendq()`, and `xpt_set_transfer_settings()`.

6.15.2.9 `const char*` `cam_sim::sim_name`

Definition at line 91 of file `cam_sim.h`.

Referenced by `cam_sim_alloc()`, `cam_sim_name()`, `xpt_bus_register()`, and `xptaction()`.

6.15.2.10 `sim_poll_func` `cam_sim::sim_poll`

Definition at line 90 of file `cam_sim.h`.

Referenced by `cam_sim_alloc()`, and `xpt_polled_action()`.

6.15.2.11 void* [cam_sim::softc](#)

Definition at line 92 of file [cam_sim.h](#).

Referenced by [cam_sim_alloc\(\)](#), [cam_sim_softc\(\)](#), and [xpt_init\(\)](#).

6.15.2.12 u_int32_t [cam_sim::unit_number](#)

Definition at line 94 of file [cam_sim.h](#).

Referenced by [cam_sim_alloc\(\)](#), [cam_sim_unit\(\)](#), [xpt_bus_register\(\)](#), and [xptaction\(\)](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/cam_sim.h](#)

6.16 `cam_status_entry` Struct Reference

```
#include <cam.h>
```

Data Fields

- [cam_status status_code](#)
- `const char *` [status_text](#)

6.16.1 Detailed Description

Definition at line 182 of file `cam.h`.

6.16.2 Field Documentation

6.16.2.1 [cam_status cam_status_entry::status_code](#)

Definition at line 184 of file `cam.h`.

Referenced by `camstatusentrycomp()`.

6.16.2.2 `const char*` [cam_status_entry::status_text](#)

Definition at line 185 of file `cam.h`.

Referenced by `cam_error_string()`, and `passasync()`.

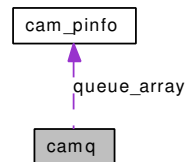
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/cam.h`

6.17 camq Struct Reference

```
#include <cam_queue.h>
```

Collaboration diagram for camq:



Data Fields

- [cam_pinfo ** queue_array](#)
- [int array_size](#)
- [int entries](#)
- [u_int32_t generation](#)
- [u_int32_t qfrozen_cnt](#)

6.17.1 Detailed Description

Definition at line 45 of file `cam_queue.h`.

6.17.2 Field Documentation

6.17.2.1 `int camq::array_size`

Definition at line 47 of file `cam_queue.h`.

Referenced by `camq_insert()`, `camq_resize()`, `xpt_alloc_device()`, and `xpt_release_device()`.

6.17.2.2 `int camq::entries`

Definition at line 48 of file `cam_queue.h`.

Referenced by `cam_ccbq_pending_ccb_count()`, `cam_ccbq_resize()`, `camq_change_priority()`, `camq_insert()`, `camq_remove()`, `camq_resize()`, `cdrunchangerqueue()`, `dev_allocq_is_runnable()`, `xpt_run_dev_allocq()`, and `xpt_run_dev_sendq()`.

6.17.2.3 `u_int32_t camq::generation`

Definition at line 49 of file `cam_queue.h`.

Referenced by `cdrunchangerqueue()`, and `xpt_schedule_dev()`.

6.17.2.4 `u_int32_t camq::qfrozen_cnt`

Definition at line 50 of file `cam_queue.h`.

Referenced by `cdrunchangerqueue()`, `dev_allocq_is_runnable()`, `xpt_freeze_simq()`, `xpt_release_simq()`, `xpt_run_dev_allocq()`, and `xpt_run_dev_sendq()`.

6.17.2.5 `cam_pinfo** camq::queue_array`

Definition at line 46 of file `cam_queue.h`.

Referenced by `cam_ccbq_peek_ccb()`, `camq_change_priority()`, `camq_fini()`, `camq_insert()`, `camq_remove()`, and `camq_resize()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/cam_queue.h`

6.18 camq_entry Union Reference

```
#include <cam_ccb.h>
```

Public Member Functions

- [LIST_ENTRY \(ccb_hdr\) le](#)
- [SLIST_ENTRY \(ccb_hdr\) sle](#)
- [TAILQ_ENTRY \(ccb_hdr\) tqe](#)
- [STAILQ_ENTRY \(ccb_hdr\) stqe](#)

6.18.1 Detailed Description

Definition at line 235 of file cam_ccb.h.

6.18.2 Member Function Documentation

6.18.2.1 [camq_entry::LIST_ENTRY \(ccb_hdr\)](#)

6.18.2.2 [camq_entry::SLIST_ENTRY \(ccb_hdr\)](#)

6.18.2.3 [camq_entry::STAILQ_ENTRY \(ccb_hdr\)](#)

6.18.2.4 [camq_entry::TAILQ_ENTRY \(ccb_hdr\)](#)

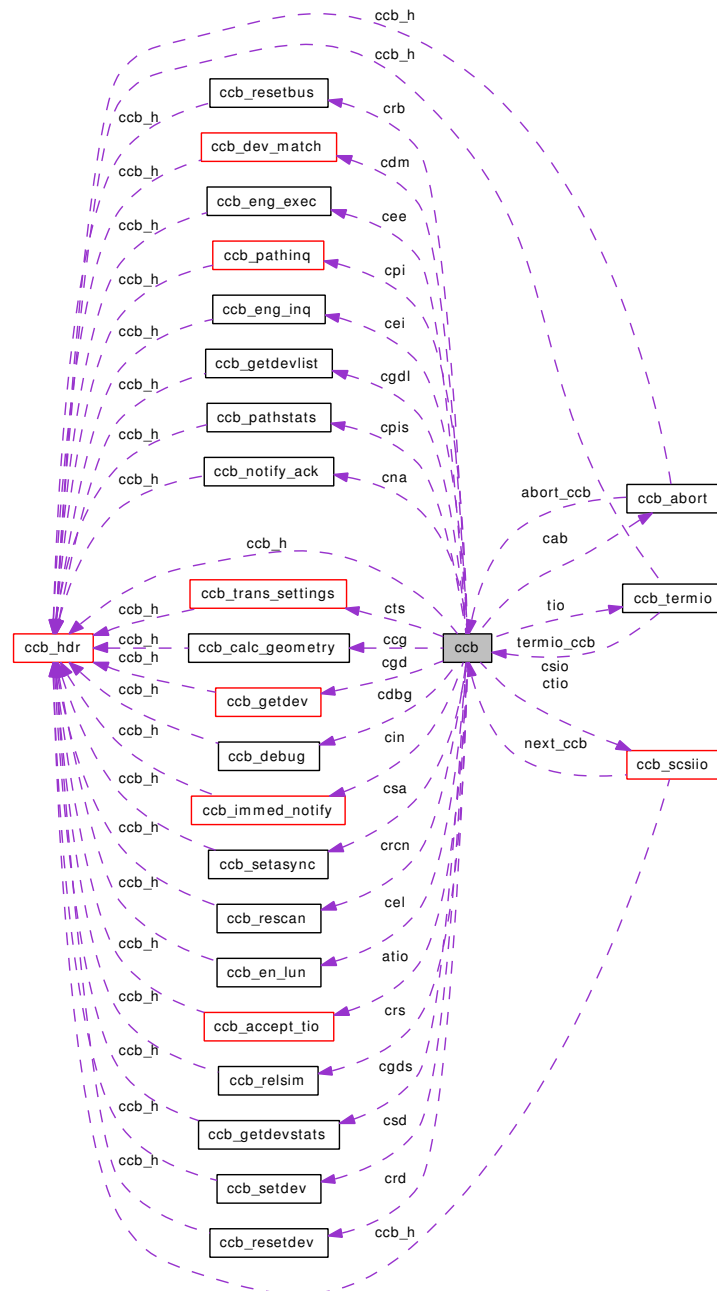
The documentation for this union was generated from the following file:

- [/usr/src/sys/cam/cam_ccb.h](#)

6.19 ccb Union Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for ccb:



Data Fields

- [ccb_hdr](#) `ccb_h`
- [ccb_scsiio](#) `csio`
- [ccb_getdev](#) `cgd`
- [ccb_getdevlist](#) `cgdl`
- [ccb_pathinq](#) `cpi`
- [ccb_relsim](#) `crs`
- [ccb_setasync](#) `csa`
- [ccb_setdev](#) `csd`
- [ccb_pathstats](#) `cpis`
- [ccb_getdevstats](#) `cgds`
- [ccb_dev_match](#) `cdm`
- [ccb_trans_settings](#) `cts`
- [ccb_calc_geometry](#) `cgc`
- [ccb_abort](#) `cab`
- [ccb_resetbus](#) `crb`
- [ccb_resetdev](#) `crd`
- [ccb_termio](#) `tio`
- [ccb_accept_tio](#) `atio`
- [ccb_scsiio](#) `ctio`
- [ccb_en_lun](#) `cel`
- [ccb_immed_notify](#) `cin`
- [ccb_notify_ack](#) `cna`
- [ccb_eng_inq](#) `cei`
- [ccb_eng_exec](#) `cee`
- [ccb_rescan](#) `cren`
- [ccb_debug](#) `cdbg`

6.19.1 Detailed Description

Definition at line 880 of file `cam_ccb.h`.

6.19.2 Field Documentation

6.19.2.1 struct [ccb_accept_tio](#) `ccb::atio`

Definition at line 898 of file `cam_ccb.h`.

Referenced by `targbhdslun()`, and `targbhdone()`.

6.19.2.2 struct [ccb_abort](#) `ccb::cab`

Definition at line 894 of file `cam_ccb.h`.

Referenced by `abort_all_pending()`, `targbhdslun()`, and `xpt_action()`.

6.19.2.3 struct `ccb_hdr ccb::ccb_h`

Definition at line 881 of file `cam_ccb.h`.

Referenced by `abort_all_pending()`, `cam_ccbq_ccb_done()`, `cam_ccbq_insert_ccb()`, `cam_ccbq_remove_ccb()`, `cam_ccbq_send_ccb()`, `cam_error_string()`, `cam_periph_ccbwait()`, `cam_periph_error()`, `cam_periph_ioctl()`, `cam_periph_mapmem()`, `cam_periph_unmapmem()`, `camisr()`, `camperiphdone()`, `camperiphfree()`, `camperiphscsisenseerror()`, `camperiphscsisistatuserror()`, `cd6byteworkaround()`, `cddone()`, `cderror()`, `cdregister()`, `cdstart()`, `chdone()`, `cherror()`, `chinit()`, `chstart()`, `cmd6workaround()`, `daclose()`, `dadone()`, `dadump()`, `daerror()`, `dagetcapacity()`, `dainit()`, `daregister()`, `dasetgeom()`, `dashutdown()`, `dastart()`, `dead_sim_action()`, `passdone()`, `passerror()`, `passinit()`, `passioctl()`, `passsendccb()`, `passstart()`, `probedone()`, `proberegister()`, `proberequestbackoff()`, `proberequestdefaultnegotiation()`, `probestart()`, `ptdone()`, `pterror()`, `ptinit()`, `ptstart()`, `sadone()`, `saerase()`, `saerror()`, `sagetparams()`, `sainit()`, `sardpos()`, `saretension()`, `sarewind()`, `sasetparams()`, `sasetpos()`, `saspace()`, `sastart()`, `sawritefilemarks()`, `scsi_command_string()`, `scsi_sense_sbuf()`, `ses_runcmd()`, `sesdone()`, `seserror()`, `sesinit()`, `sesstart()`, `STAILQ_HEAD()`, `TAILQ_HEAD()`, `targbhdislun()`, `targbhdone()`, `targbhenlun()`, `targbhinit()`, `targbhstart()`, `targdone()`, `targenable()`, `targendlun()`, `targfreeccb()`, `targioctl()`, `targreturnccb()`, `targsendccb()`, `targusermerge()`, `targwrite()`, `xpt_action()`, `xpt_announce_periph()`, `xpt_bus_deregister()`, `xpt_done()`, `xpt_finishconfig()`, `xpt_get_ccb()`, `xpt_merge_ccb()`, `xpt_polled_action()`, `xpt_release_ccb()`, `xpt_rescan()`, `xpt_run_dev_allocq()`, `xpt_run_dev_sendq()`, `xpt_scan_bus()`, `xpt_scan_lun()`, `xpt_schedule()`, `xpt_set_transfer_settings()`, `xptaction()`, `xptconfigfunc()`, `xptdone()`, `xptioctl()`, and `xptscandone()`.

6.19.2.4 struct `ccb_calc_geometry ccb::ccg`

Definition at line 893 of file `cam_ccb.h`.

Referenced by `xpt_action()`.

6.19.2.5 struct `ccb_debug ccb::cdbg`

Definition at line 906 of file `cam_ccb.h`.

Referenced by `xpt_action()`.

6.19.2.6 struct `ccb_dev_match ccb::cdm`

Definition at line 891 of file `cam_ccb.h`.

Referenced by `cam_periph_mapmem()`, `cam_periph_unmapmem()`, and `xpt_action()`.

6.19.2.7 struct `ccb_eng_exec ccb::cee`

Definition at line 904 of file `cam_ccb.h`.

6.19.2.8 struct `ccb_eng_inq ccb::cei`

Definition at line 903 of file `cam_ccb.h`.

6.19.2.9 struct `ccb_en_lun ccb::cel`

Definition at line 900 of file `cam_ccb.h`.

Referenced by targbhdislun(), and targbhenlun().

6.19.2.10 struct `ccb_getdev ccb::cgd`

Definition at line 883 of file cam_ccb.h.

Referenced by camperiphscsisenseerror(), and xpt_action().

6.19.2.11 struct `ccb_getdevlist ccb::cgdl`

Definition at line 884 of file cam_ccb.h.

Referenced by cam_periph_ioctl(), xpt_action(), and xptioctl().

6.19.2.12 struct `ccb_getdevstats ccb::cgds`

Definition at line 890 of file cam_ccb.h.

Referenced by xpt_action().

6.19.2.13 struct `ccb_immed_notify ccb::cin`

Definition at line 901 of file cam_ccb.h.

6.19.2.14 struct `ccb_notify_ack ccb::cna`

Definition at line 902 of file cam_ccb.h.

6.19.2.15 struct `ccb_pathinq ccb::cpi`

Definition at line 885 of file cam_ccb.h.

Referenced by probeschedule(), xpt_scan_bus(), xpt_set_transfer_settings(), xptaction(), and xptconfigfunc().

6.19.2.16 struct `ccb_pathstats ccb::cpis`

Definition at line 889 of file cam_ccb.h.

Referenced by xpt_action().

6.19.2.17 struct `ccb_resetbus ccb::crb`

Definition at line 895 of file cam_ccb.h.

6.19.2.18 struct `ccb_rescan ccb::crcn`

Definition at line 905 of file cam_ccb.h.

Referenced by xpt_action(), xpt_finishconfig(), xpt_scan_bus(), and xpt_scan_lun().

6.19.2.19 struct `ccb_resetdev ccb::crd`

Definition at line 896 of file `cam_ccb.h`.

6.19.2.20 struct `ccb_relsim ccb::crs`

Definition at line 886 of file `cam_ccb.h`.

Referenced by `xpt_action()`.

6.19.2.21 struct `ccb_setasync ccb::csa`

Definition at line 887 of file `cam_ccb.h`.

Referenced by `xpt_action()`.

6.19.2.22 struct `ccb_setdev ccb::csd`

Definition at line 888 of file `cam_ccb.h`.

6.19.2.23 struct `ccb_scsiio ccb::csio`

Definition at line 882 of file `cam_ccb.h`.

Referenced by `cam_error_string()`, `cam_periph_mapmem()`, `cam_periph_unmapmem()`, `camperiphdone()`, `camperiphscsisenseerror()`, `camperiphscsisatuserror()`, `cd6byteworkaround()`, `cddone()`, `cderror()`, `cdgetmode()`, `cdpause()`, `cdplay()`, `cdplaymsf()`, `cdplaytracks()`, `cdprevent()`, `cdreadvdstructure()`, `cdreadsubchannel()`, `cdreadtoc()`, `cdreportkey()`, `cdsendkey()`, `cdsetmode()`, `cdsetspeed()`, `cdsize()`, `cdstart()`, `cdstartunit()`, `cdstopunit()`, `chdone()`, `chexchange()`, `chgetelemstatus()`, `chgetparams()`, `chielem()`, `chmove()`, `chposition()`, `chstart()`, `cmd6workaround()`, `daclose()`, `dadone()`, `daerror()`, `dagetcapacity()`, `daprevent()`, `dashutdown()`, `dastart()`, `passdone()`, `probedone()`, `probestart()`, `ptdone()`, `ptstart()`, `sadone()`, `saerase()`, `saerror()`, `sagetparams()`, `saloadunload()`, `samount()`, `saprevent()`, `sardpos()`, `sareservereleaseunit()`, `saretension()`, `sarewind()`, `sasetparams()`, `sasetpos()`, `saspace()`, `sastart()`, `sawritefilemarks()`, `ses_runcmd()`, `targbhstart()`, `xpt_action()`, and `xpt_run_dev_sendq()`.

6.19.2.24 struct `ccb_scsiio ccb::ctio`

Definition at line 899 of file `cam_ccb.h`.

6.19.2.25 struct `ccb_trans_settings ccb::cts`

Definition at line 892 of file `cam_ccb.h`.

Referenced by `xpt_action()`, and `xpt_set_transfer_settings()`.

6.19.2.26 struct `ccb_termio ccb::tio`

Definition at line 897 of file `cam_ccb.h`.

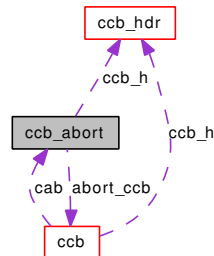
The documentation for this union was generated from the following file:

- [/usr/src/sys/cam/cam_ccb.h](#)

6.20 ccb_abort Struct Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for ccb_abort:



Data Fields

- [ccb_hdr ccb_h](#)
- [ccb * abort_ccb](#)

6.20.1 Detailed Description

Definition at line 678 of file cam_ccb.h.

6.20.2 Field Documentation

6.20.2.1 union [ccb* ccb_abort::abort_ccb](#)

Definition at line 680 of file cam_ccb.h.

Referenced by [abort_all_pending\(\)](#), [targbhdslun\(\)](#), [targusermerge\(\)](#), and [xpt_action\(\)](#).

6.20.2.2 struct [ccb_hdr ccb_abort::ccb_h](#)

Definition at line 679 of file cam_ccb.h.

Referenced by [abort_all_pending\(\)](#), [targbhdslun\(\)](#), and [targusermerge\(\)](#).

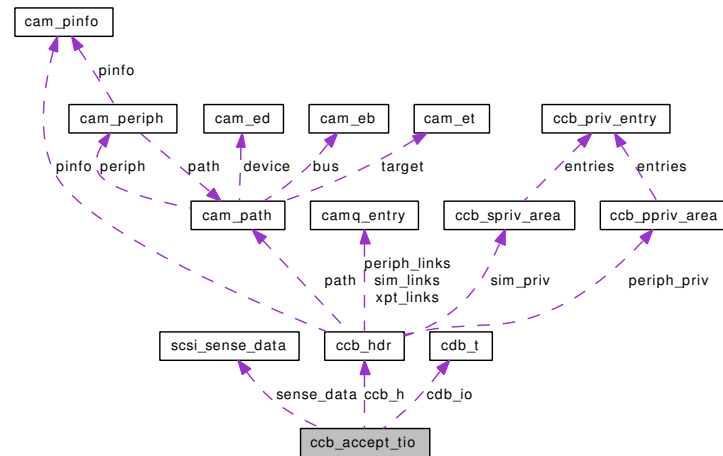
The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/cam_ccb.h](#)

6.21 ccb_accept_tio Struct Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for ccb_accept_tio:



Data Fields

- [ccb_hdr](#) `ccb_h`
- [ccb_t](#) `ccb_io`
- [u_int8_t](#) `ccb_len`
- [u_int8_t](#) `tag_action`
- [u_int8_t](#) `sense_len`
- [u_int](#) `tag_id`
- [u_int](#) `init_id`
- [scsi_sense_data](#) `sense_data`

6.21.1 Detailed Description

Definition at line 617 of file `cam_ccb.h`.

6.21.2 Field Documentation

6.21.2.1 struct `ccb_hdr` `ccb_accept_tio::ccb_h`

Definition at line 618 of file `cam_ccb.h`.

Referenced by `targbhdslun()`, `targbhdone()`, and `targbhstart()`.

6.21.2.2 `ccb_t` `ccb_accept_tio::ccb_io`

Definition at line 619 of file `cam_ccb.h`.

Referenced by `targbhdone()`.

6.21.2.3 `u_int8_t ccb_accept_tio::cdb_len`

Definition at line 620 of file `cam_ccb.h`.

6.21.2.4 `u_int ccb_accept_tio::init_id`

Definition at line 624 of file `cam_ccb.h`.

Referenced by `targbhstart()`.

6.21.2.5 `struct scsi_sense_data ccb_accept_tio::sense_data`

Definition at line 625 of file `cam_ccb.h`.

Referenced by `targbhdone()`, and `targbhstart()`.

6.21.2.6 `u_int8_t ccb_accept_tio::sense_len`

Definition at line 622 of file `cam_ccb.h`.

Referenced by `targbhdone()`, and `targbhstart()`.

6.21.2.7 `u_int8_t ccb_accept_tio::tag_action`

Definition at line 621 of file `cam_ccb.h`.

6.21.2.8 `u_int ccb_accept_tio::tag_id`

Definition at line 623 of file `cam_ccb.h`.

Referenced by `targbhstart()`.

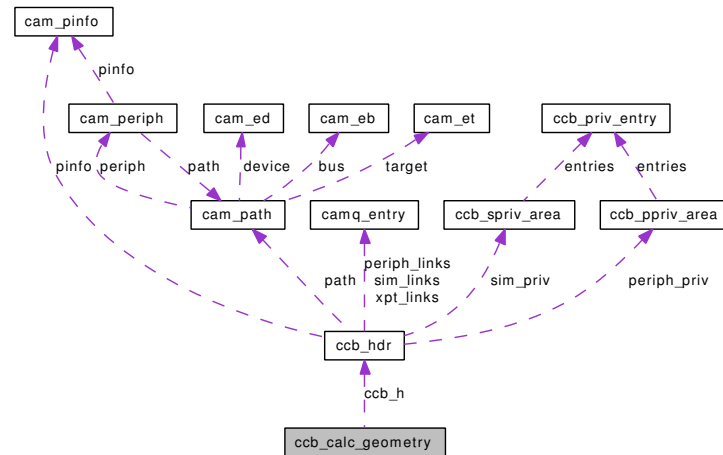
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/cam_ccb.h`

6.22 ccb_calc_geometry Struct Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for ccb_calc_geometry:



Data Fields

- [ccb_hdr](#) [ccb_h](#)
- [u_int32_t](#) [block_size](#)
- [u_int64_t](#) [volume_size](#)
- [u_int32_t](#) [cylinders](#)
- [u_int8_t](#) [heads](#)
- [u_int8_t](#) [secs_per_track](#)

6.22.1 Detailed Description

Definition at line 772 of file `cam_ccb.h`.

6.22.2 Field Documentation

6.22.2.1 [u_int32_t](#) [ccb_calc_geometry::block_size](#)

Definition at line 774 of file `cam_ccb.h`.

Referenced by `cam_calc_geometry()`, `dasetgeom()`, and `xpt_action()`.

6.22.2.2 [struct](#) [ccb_hdr](#) [ccb_calc_geometry::ccb_h](#)

Definition at line 773 of file `cam_ccb.h`.

Referenced by `cam_calc_geometry()`, and `dasetgeom()`.

6.22.2.3 [u_int32_t ccb_calc_geometry::cylinders](#)

Definition at line 776 of file `cam_ccb.h`.

Referenced by `cam_calc_geometry()`, `dasetgeom()`, and `xpt_action()`.

6.22.2.4 [u_int8_t ccb_calc_geometry::heads](#)

Definition at line 777 of file `cam_ccb.h`.

Referenced by `cam_calc_geometry()`, `dasetgeom()`, and `xpt_action()`.

6.22.2.5 [u_int8_t ccb_calc_geometry::secs_per_track](#)

Definition at line 778 of file `cam_ccb.h`.

Referenced by `cam_calc_geometry()`, `dasetgeom()`, and `xpt_action()`.

6.22.2.6 [u_int64_t ccb_calc_geometry::volume_size](#)

Definition at line 775 of file `cam_ccb.h`.

Referenced by `cam_calc_geometry()`, `dasetgeom()`, and `xpt_action()`.

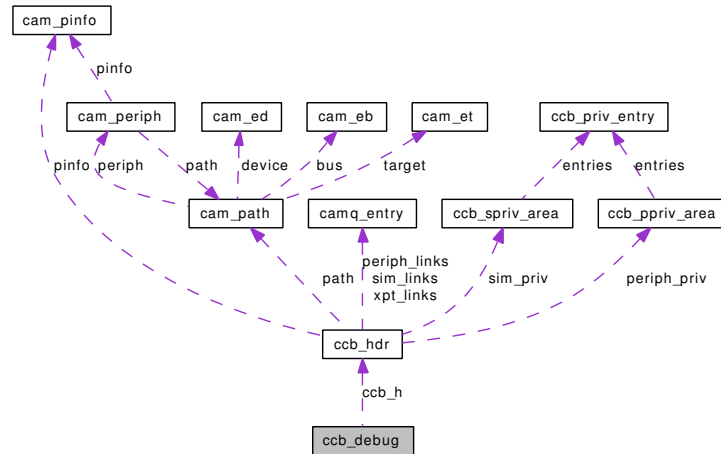
The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/cam_ccb.h](#)

6.23 ccb_debug Struct Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for ccb_debug:



Data Fields

- [ccb_hdr ccb_h](#)
- [cam_debug_flags flags](#)

6.23.1 Detailed Description

Definition at line 792 of file `cam_ccb.h`.

6.23.2 Field Documentation

6.23.2.1 struct `ccb_hdr ccb_debug::ccb_h`

Definition at line 793 of file `cam_ccb.h`.

6.23.2.2 `cam_debug_flags ccb_debug::flags`

Definition at line 794 of file `cam_ccb.h`.

Referenced by `xpt_action()`.

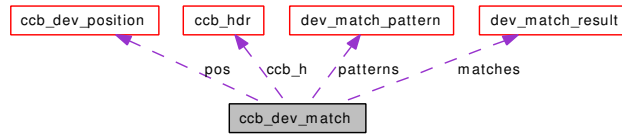
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/cam_ccb.h`

6.24 ccb_dev_match Struct Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for ccb_dev_match:



Data Fields

- [ccb_hdr](#) [ccb_h](#)
- [ccb_dev_match_status](#) [status](#)
- [u_int32_t](#) [num_patterns](#)
- [u_int32_t](#) [pattern_buf_len](#)
- [dev_match_pattern](#) * [patterns](#)
- [u_int32_t](#) [num_matches](#)
- [u_int32_t](#) [match_buf_len](#)
- [dev_match_result](#) * [matches](#)
- [ccb_dev_position](#) [pos](#)

6.24.1 Detailed Description

Definition at line 472 of file `cam_ccb.h`.

6.24.2 Field Documentation

6.24.2.1 struct [ccb_hdr](#) [ccb_dev_match::ccb_h](#)

Definition at line 473 of file `cam_ccb.h`.

6.24.2.2 [u_int32_t](#) [ccb_dev_match::match_buf_len](#)

Definition at line 479 of file `cam_ccb.h`.

Referenced by `cam_periph_mapmem()`, `xptedtbusfunc()`, `xptedtdevicefunc()`, `xptedtperiphfunc()`, and `xptplistperiphfunc()`.

6.24.2.3 struct [dev_match_result](#)* [ccb_dev_match::matches](#)

Definition at line 480 of file `cam_ccb.h`.

Referenced by `cam_periph_mapmem()`, `cam_periph_unmapmem()`, `xptedtbusfunc()`, `xptedtdevicefunc()`, `xptedtperiphfunc()`, and `xptplistperiphfunc()`.

6.24.2.4 `u_int32_t ccb_dev_match::num_matches`

Definition at line 478 of file `cam_ccb.h`.

Referenced by `xptedtbusfunc()`, `xptedtdevicefunc()`, `xptedtmatch()`, `xptedtperiphfunc()`, `xptperiphlistmatch()`, and `xptplistperiphfunc()`.

6.24.2.5 `u_int32_t ccb_dev_match::num_patterns`

Definition at line 475 of file `cam_ccb.h`.

Referenced by `xpt_action()`, `xptedtbusfunc()`, `xptedtdevicefunc()`, `xptedtperiphfunc()`, and `xptplistperiphfunc()`.

6.24.2.6 `u_int32_t ccb_dev_match::pattern_buf_len`

Definition at line 476 of file `cam_ccb.h`.

Referenced by `cam_periph_mapmem()`.

6.24.2.7 `struct dev_match_pattern* ccb_dev_match::patterns`

Definition at line 477 of file `cam_ccb.h`.

Referenced by `cam_periph_mapmem()`, `cam_periph_unmapmem()`, `xpt_action()`, `xptedtbusfunc()`, `xptedtdevicefunc()`, `xptedtperiphfunc()`, and `xptplistperiphfunc()`.

6.24.2.8 `struct ccb_dev_position ccb_dev_match::pos`

Definition at line 481 of file `cam_ccb.h`.

Referenced by `xpt_action()`, `xptedtbusfunc()`, `xptedtdevicefunc()`, `xptedtmatch()`, `xptedtperiphfunc()`, `xptedttargetfunc()`, `xptperiphlistmatch()`, `xptlistpdrvfunc()`, and `xptplistperiphfunc()`.

6.24.2.9 `ccb_dev_match_status ccb_dev_match::status`

Definition at line 474 of file `cam_ccb.h`.

Referenced by `xptedtbusfunc()`, `xptedtdevicefunc()`, `xptedtmatch()`, `xptedtperiphfunc()`, `xptedttargetfunc()`, `xptperiphlistmatch()`, `xptlistpdrvfunc()`, and `xptplistperiphfunc()`.

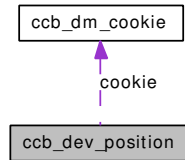
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/cam_ccb.h`

6.25 ccb_dev_position Struct Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for ccb_dev_position:



Data Fields

- [u_int generations](#) [4]
- [dev_pos_type position_type](#)
- [ccb_dm_cookie cookie](#)

6.25.1 Detailed Description

Definition at line 462 of file `cam_ccb.h`.

6.25.2 Field Documentation

6.25.2.1 struct `ccb_dm_cookie` `ccb_dev_position::cookie`

Definition at line 469 of file `cam_ccb.h`.

Referenced by `xptedtbusfunc()`, `xptedtdevicefunc()`, `xptedtmatch()`, `xptedtperiphfunc()`, `xptedttargetfunc()`, `xptperiphlistmatch()`, `xptplistdrvfunc()`, and `xptplistperiphfunc()`.

6.25.2.2 u_int `ccb_dev_position::generations`[4]

Definition at line 463 of file `cam_ccb.h`.

Referenced by `xptedtbusfunc()`, `xptedtdevicefunc()`, `xptedtmatch()`, `xptedtperiphfunc()`, `xptedttargetfunc()`, `xptplistdrvfunc()`, and `xptplistperiphfunc()`.

6.25.2.3 dev_pos_type `ccb_dev_position::position_type`

Definition at line 468 of file `cam_ccb.h`.

Referenced by `xpt_action()`, `xptedtbusfunc()`, `xptedtdevicefunc()`, `xptedtmatch()`, `xptedtperiphfunc()`, `xptedttargetfunc()`, `xptperiphlistmatch()`, `xptplistdrvfunc()`, and `xptplistperiphfunc()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/cam_ccb.h`

6.26 ccb_dm_cookie Struct Reference

```
#include <cam_ccb.h>
```

Data Fields

- void * [bus](#)
- void * [target](#)
- void * [device](#)
- void * [periph](#)
- void * [pdrv](#)

6.26.1 Detailed Description

Definition at line 454 of file cam_ccb.h.

6.26.2 Field Documentation

6.26.2.1 void* [ccb_dm_cookie::bus](#)

Definition at line 455 of file cam_ccb.h.

Referenced by [xptedtbusfunc\(\)](#), [xptedtdevicefunc\(\)](#), [xptedtmatch\(\)](#), [xptedtperiphfunc\(\)](#), and [xptedttargetfunc\(\)](#).

6.26.2.2 void* [ccb_dm_cookie::device](#)

Definition at line 457 of file cam_ccb.h.

Referenced by [xptedtdevicefunc\(\)](#), [xptedtperiphfunc\(\)](#), and [xptedttargetfunc\(\)](#).

6.26.2.3 void* [ccb_dm_cookie::pdrv](#)

Definition at line 459 of file cam_ccb.h.

Referenced by [xptperiphlistmatch\(\)](#), [xptplistpdrvfunc\(\)](#), and [xptplistperiphfunc\(\)](#).

6.26.2.4 void* [ccb_dm_cookie::periph](#)

Definition at line 458 of file cam_ccb.h.

Referenced by [xptedtdevicefunc\(\)](#), [xptedtperiphfunc\(\)](#), [xptplistpdrvfunc\(\)](#), and [xptplistperiphfunc\(\)](#).

6.26.2.5 void* [ccb_dm_cookie::target](#)

Definition at line 456 of file cam_ccb.h.

Referenced by [xptedtbusfunc\(\)](#), [xptedtdevicefunc\(\)](#), [xptedtperiphfunc\(\)](#), and [xptedttargetfunc\(\)](#).

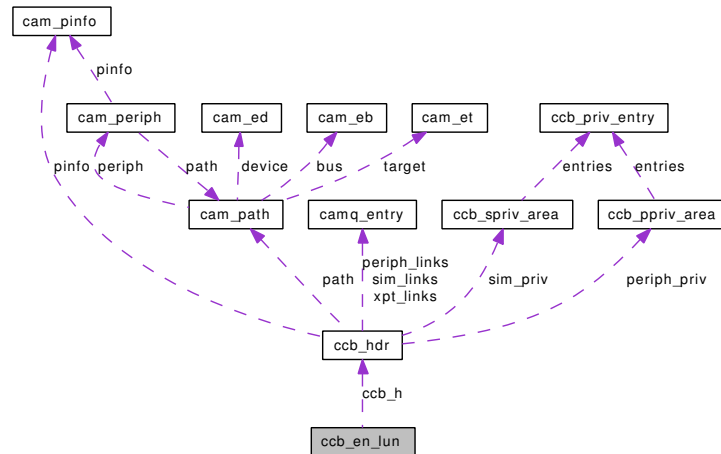
The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/cam_ccb.h](#)

6.27 ccb_en_lun Struct Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for ccb_en_lun:



Data Fields

- [ccb_hdr](#) [ccb_h](#)
- [u_int16_t](#) [grp6_len](#)
- [u_int16_t](#) [grp7_len](#)
- [u_int8_t](#) [enable](#)

6.27.1 Detailed Description

Definition at line 799 of file `cam_ccb.h`.

6.27.2 Field Documentation

6.27.2.1 struct [ccb_hdr](#) [ccb_en_lun::ccb_h](#)

Definition at line 800 of file `cam_ccb.h`.

Referenced by `targbhdslun()`, and `targendslun()`.

6.27.2.2 [u_int8_t](#) [ccb_en_lun::enable](#)

Definition at line 803 of file `cam_ccb.h`.

Referenced by `targbhdslun()`, `targbhenlun()`, and `targendslun()`.

6.27.2.3 [u_int16_t](#) [ccb_en_lun::grp6_len](#)

Definition at line 801 of file `cam_ccb.h`.

Referenced by `targbhenlun()`, and `targendislnun()`.

6.27.2.4 `u_int16_t ccb_en_lun::grp7_len`

Definition at line 802 of file `cam_ccb.h`.

Referenced by `targbhenlun()`, and `targendislnun()`.

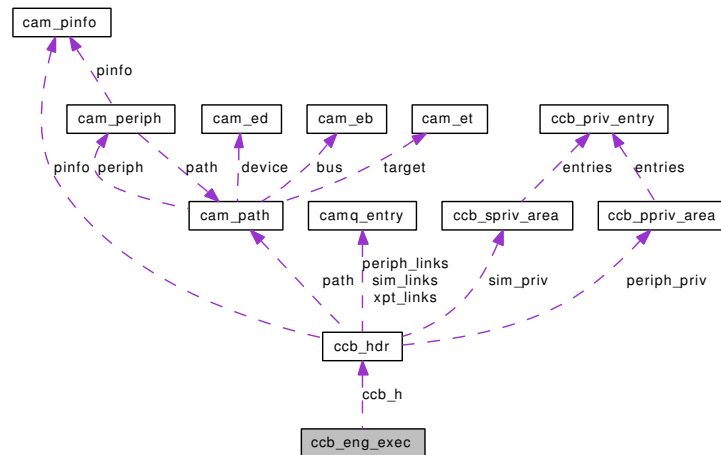
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/cam_ccb.h`

6.28 ccb_eng_exec Struct Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for ccb_eng_exec:



Data Fields

- [ccb_hdr](#) [ccb_h](#)
- [u_int8_t](#) * [pdrv_ptr](#)
- [u_int8_t](#) * [req_map](#)
- [u_int8_t](#) * [data_ptr](#)
- [u_int32_t](#) [dxfer_len](#)
- [u_int8_t](#) * [engdata_ptr](#)
- [u_int16_t](#) [sglist_cnt](#)
- [u_int32_t](#) [dmax_len](#)
- [u_int32_t](#) [dest_len](#)
- [int32_t](#) [src_resid](#)
- [u_int32_t](#) [timeout](#)
- [u_int16_t](#) [eng_num](#)
- [u_int16_t](#) [vu_flags](#)

6.28.1 Detailed Description

Definition at line 844 of file [cam_ccb.h](#).

6.28.2 Field Documentation

6.28.2.1 struct [ccb_hdr](#) [ccb_eng_exec::ccb_h](#)

Definition at line 845 of file [cam_ccb.h](#).

6.28.2.2 `u_int8_t* ccb_eng_exec::data_ptr`

Definition at line 848 of file cam_ccb.h.

6.28.2.3 `u_int32_t ccb_eng_exec::dest_len`

Definition at line 853 of file cam_ccb.h.

6.28.2.4 `u_int32_t ccb_eng_exec::dmax_len`

Definition at line 852 of file cam_ccb.h.

6.28.2.5 `u_int32_t ccb_eng_exec::dxfer_len`

Definition at line 849 of file cam_ccb.h.

6.28.2.6 `u_int16_t ccb_eng_exec::eng_num`

Definition at line 856 of file cam_ccb.h.

6.28.2.7 `u_int8_t* ccb_eng_exec::engdata_ptr`

Definition at line 850 of file cam_ccb.h.

6.28.2.8 `u_int8_t* ccb_eng_exec::pdrv_ptr`

Definition at line 846 of file cam_ccb.h.

6.28.2.9 `u_int8_t* ccb_eng_exec::req_map`

Definition at line 847 of file cam_ccb.h.

6.28.2.10 `u_int16_t ccb_eng_exec::sglist_cnt`

Definition at line 851 of file cam_ccb.h.

6.28.2.11 `int32_t ccb_eng_exec::src_resid`

Definition at line 854 of file cam_ccb.h.

6.28.2.12 `u_int32_t ccb_eng_exec::timeout`

Definition at line 855 of file cam_ccb.h.

6.28.2.13 `u_int16_t ccb_eng_exec::vu_flags`

Definition at line 857 of file `cam_ccb.h`.

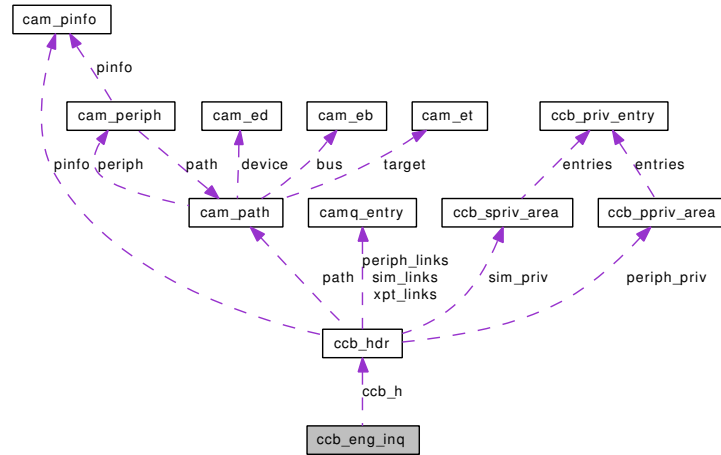
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/cam_ccb.h`

6.29 ccb_eng_inq Struct Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for ccb_eng_inq:



Data Fields

- [ccb_hdr](#) [ccb_h](#)
- [u_int16_t](#) [eng_num](#)
- [ei_type](#) [eng_type](#)
- [ei_algo](#) [eng_algo](#)
- [u_int32_t](#) [eng_memeory](#)

6.29.1 Detailed Description

Definition at line 836 of file `cam_ccb.h`.

6.29.2 Field Documentation

6.29.2.1 struct [ccb_hdr](#) [ccb_eng_inq::ccb_h](#)

Definition at line 837 of file `cam_ccb.h`.

6.29.2.2 [ei_algo](#) [ccb_eng_inq::eng_algo](#)

Definition at line 840 of file `cam_ccb.h`.

6.29.2.3 [u_int32_t](#) [ccb_eng_inq::eng_memeory](#)

Definition at line 841 of file `cam_ccb.h`.

6.29.2.4 `u_int16_t ccb_eng_inq::eng_num`

Definition at line 838 of file `cam_ccb.h`.

6.29.2.5 `ei_type ccb_eng_inq::eng_type`

Definition at line 839 of file `cam_ccb.h`.

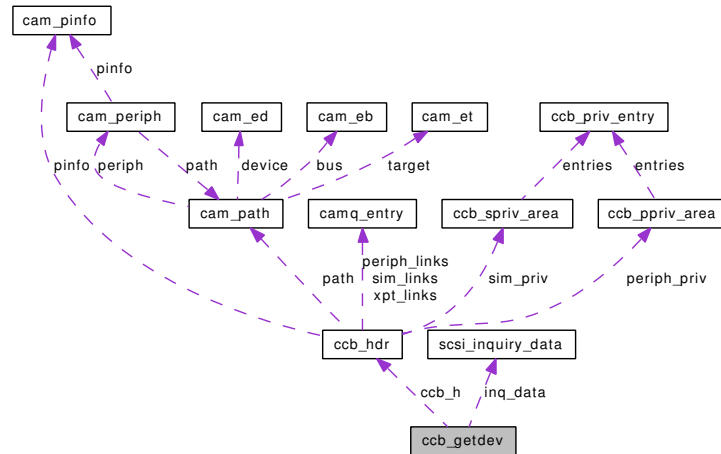
The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/cam_ccb.h](#)

6.30 ccb_getdev Struct Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for ccb_getdev:



Data Fields

- [ccb_hdr](#) [ccb_h](#)
- [scsi_inquiry_data](#) [inq_data](#)
- [u_int8_t](#) [serial_num](#) [252]
- [u_int8_t](#) [reserved](#)
- [u_int8_t](#) [serial_num_len](#)

6.30.1 Detailed Description

Definition at line 282 of file `cam_ccb.h`.

6.30.2 Field Documentation

6.30.2.1 struct [ccb_hdr](#) `ccb_getdev::ccb_h`

Definition at line 283 of file `cam_ccb.h`.

Referenced by `camperiphdone()`, `camperiphscsisenseerror()`, `cdasync()`, `cddone()`, `cdregister()`, `chasync()`, `daasync()`, `dadone()`, `passasync()`, `ptasync()`, `saasync()`, `scsi_command_string()`, `scsi_sense_sbuf()`, `sesasync()`, `xpt_action()`, and `xptsetasyncfunc()`.

6.30.2.2 struct [scsi_inquiry_data](#) `ccb_getdev::inq_data`

Definition at line 284 of file `cam_ccb.h`.

Referenced by `camperiphscsisenseerror()`, `cdasync()`, `cdregister()`, `chasync()`, `chregister()`, `daasync()`, `daregister()`, `passregister()`, `ptasync()`, `pctor()`, `saasync()`, `saregister()`, `scsi_command_string()`, `scsi_sense_sbuf()`, `sesasync()`, `sesregister()`, and `xpt_action()`.

6.30.2.3 `u_int8_t ccb_getdev::reserved`

Definition at line 286 of file `cam_ccb.h`.

6.30.2.4 `u_int8_t ccb_getdev::serial_num[252]`

Definition at line 285 of file `cam_ccb.h`.

Referenced by `xpt_action()`.

6.30.2.5 `u_int8_t ccb_getdev::serial_num_len`

Definition at line 287 of file `cam_ccb.h`.

Referenced by `xpt_action()`.

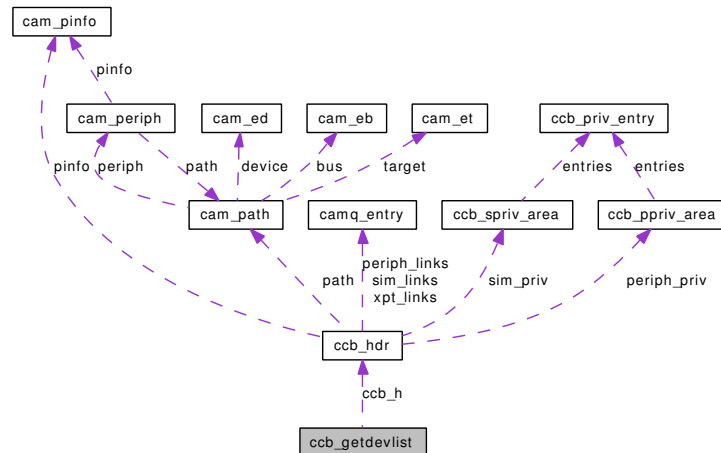
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/cam_ccb.h`

6.31 ccb_getdevlist Struct Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for ccb_getdevlist:



Data Fields

- `ccb_hdr` `ccb_h`
- `char` `periph_name` [DEV_IDLEN]
- `u_int32_t` `unit_number`
- `unsigned int` `generation`
- `u_int32_t` `index`
- `ccb_getdevlist_status_e` `status`

6.31.1 Detailed Description

Definition at line 316 of file `cam_ccb.h`.

6.31.2 Field Documentation

6.31.2.1 struct `ccb_hdr` `ccb_getdevlist::ccb_h`

Definition at line 317 of file `cam_ccb.h`.

Referenced by `xpt_action()`.

6.31.2.2 `unsigned int` `ccb_getdevlist::generation`

Definition at line 320 of file `cam_ccb.h`.

Referenced by `xpt_action()`.

6.31.2.3 `u_int32_t ccb_getdevlist::index`

Definition at line 321 of file `cam_ccb.h`.

Referenced by `cam_periph_ioctl()`, and `xpt_action()`.

6.31.2.4 `char ccb_getdevlist::periph_name[DEV_IDLEN]`

Definition at line 318 of file `cam_ccb.h`.

Referenced by `cam_periph_ioctl()`, `xpt_action()`, and `xptioctl()`.

6.31.2.5 `ccb_getdevlist_status_e ccb_getdevlist::status`

Definition at line 322 of file `cam_ccb.h`.

Referenced by `cam_periph_ioctl()`, `xpt_action()`, and `xptioctl()`.

6.31.2.6 `u_int32_t ccb_getdevlist::unit_number`

Definition at line 319 of file `cam_ccb.h`.

Referenced by `cam_periph_ioctl()`, `xpt_action()`, and `xptioctl()`.

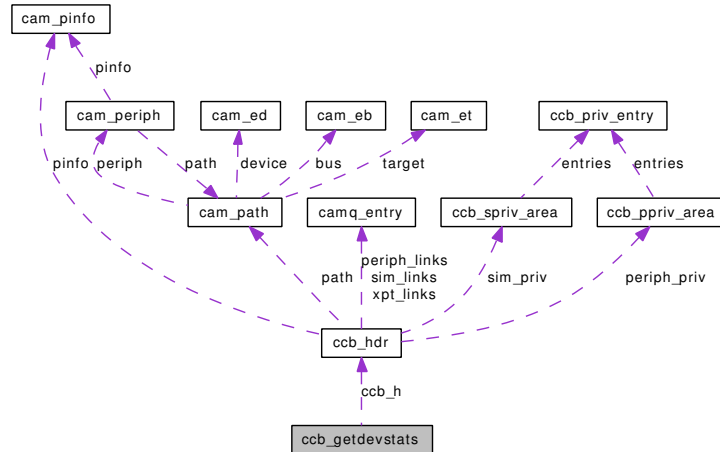
The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/cam_ccb.h](#)

6.32 ccb_getdevstats Struct Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for ccb_getdevstats:



Data Fields

- [ccb_hdr](#) [ccb_h](#)
- [int](#) [dev_openings](#)
- [int](#) [dev_active](#)
- [int](#) [devq_openings](#)
- [int](#) [devq_queued](#)
- [int](#) [held](#)
- [int](#) [maxtags](#)
- [int](#) [mintags](#)
- [timeval](#) [last_reset](#)

6.32.1 Detailed Description

Definition at line 291 of file `cam_ccb.h`.

6.32.2 Field Documentation

6.32.2.1 struct [ccb_hdr](#) [ccb_getdevstats::ccb_h](#)

Definition at line 292 of file `cam_ccb.h`.

Referenced by `cam_periph_bus_settle()`, `camperiphscsistatuserror()`, and `xpt_action()`.

6.32.2.2 [int](#) [ccb_getdevstats::dev_active](#)

Definition at line 294 of file `cam_ccb.h`.

Referenced by `xpt_action()`.

6.32.2.3 `int ccb_getdevstats::dev_openings`

Definition at line 293 of file `cam_ccb.h`.

Referenced by `xpt_action()`.

6.32.2.4 `int ccb_getdevstats::devq_openings`

Definition at line 295 of file `cam_ccb.h`.

Referenced by `xpt_action()`.

6.32.2.5 `int ccb_getdevstats::devq_queued`

Definition at line 296 of file `cam_ccb.h`.

Referenced by `xpt_action()`.

6.32.2.6 `int ccb_getdevstats::held`

Definition at line 297 of file `cam_ccb.h`.

Referenced by `xpt_action()`.

6.32.2.7 `struct timeval ccb_getdevstats::last_reset`

Definition at line 306 of file `cam_ccb.h`.

Referenced by `xpt_action()`.

6.32.2.8 `int ccb_getdevstats::maxtags`

Definition at line 301 of file `cam_ccb.h`.

Referenced by `xpt_action()`.

6.32.2.9 `int ccb_getdevstats::mintags`

Definition at line 305 of file `cam_ccb.h`.

Referenced by `xpt_action()`.

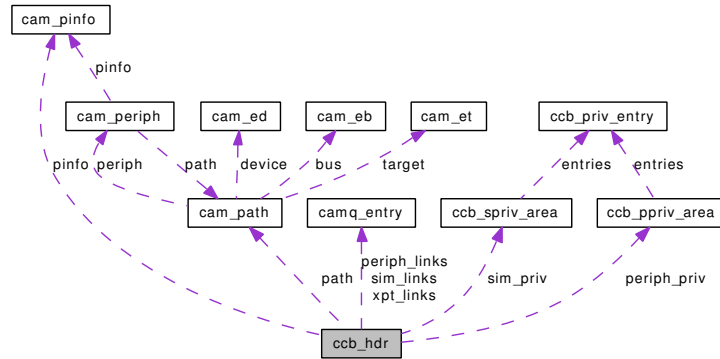
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/cam_ccb.h`

6.33 ccb_hdr Struct Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for ccb_hdr:



Data Fields

- `cam_pinfo` `pinfo`
- `camq_entry` `xpt_links`
- `camq_entry` `sim_links`
- `camq_entry` `periph_links`
- `u_int32_t` `retry_count`
- `void(* cbfcn)(struct cam_periph *, union ccb *)`
- `xpt_opcode` `func_code`
- `u_int32_t` `status`
- `cam_path` * `path`
- `path_id_t` `path_id`
- `target_id_t` `target_id`
- `lun_id_t` `target_lun`
- `u_int32_t` `flags`
- `ccb_ppriv_area` `periph_priv`
- `ccb_spriv_area` `sim_priv`
- `u_int32_t` `timeout`
- `callout_handle` `timeout_ch`

6.33.1 Detailed Description

Definition at line 259 of file `cam_ccb.h`.

6.33.2 Field Documentation

6.33.2.1 `void(* ccb_hdr::cbfcn)(struct cam_periph *, union ccb *)`

Referenced by `cam_fill_csio()`, `cam_fill_ctio()`, `cam_periph_ccbwait()`, `camisr()`, `camperiphdone()`, `cd-done()`, `chdone()`, `dadone()`, `passdone()`, `passsendccb()`, `ptdone()`, `sadone()`, `sesdone()`, `targbhdone()`, `targsendccb()`, `xpt_finishconfig()`, `xpt_scan_bus()`, `xpt_scan_lun()`, `xptconfigfunc()`, `xptdone()`, and `xptioctl()`.

6.33.2.2 `u_int32_t ccb_hdr::flags`

Definition at line 273 of file `cam_ccb.h`.

Referenced by `cam_fill_csio()`, `cam_fill_ctio()`, `cam_freeze_devq()`, `cam_periph_mapmem()`, `cam_release_devq()`, `camisr()`, `cd6byteworkaround()`, `cddone()`, `cderror()`, `cmd6workaround()`, `dadone()`, `daerror()`, `passsendccb()`, `scsi_command_string()`, `scsi_sense_sbuf()`, `targbhdone()`, `targbhstart()`, `targsendccb()`, `targusermerge()`, `xpt_action()`, `xpt_merge_ccb()`, `xpt_run_dev_sendq()`, `xpt_setup_ccb()`, and `xptioctl()`.

6.33.2.3 `xpt_opcode ccb_hdr::func_code`

Definition at line 267 of file `cam_ccb.h`.

Referenced by `abort_all_pending()`, `cam_error_string()`, `cam_fill_csio()`, `cam_fill_ctio()`, `cam_freeze_devq()`, `cam_periph_bus_settle()`, `cam_periph_ioctl()`, `cam_periph_mapmem()`, `cam_periph_unmapmem()`, `cam_release_devq()`, `camisr()`, `camperiphdone()`, `camperiphfree()`, `camperiphscsisenseerror()`, `camperiphscsisistatuserror()`, `cddone()`, `cdoninvalidate()`, `cdregister()`, `chinit()`, `choninvalidate()`, `chregister()`, `dadone()`, `dainit()`, `daoninvalidate()`, `daregister()`, `dasetgeom()`, `passinit()`, `passioctl()`, `passoninvalidate()`, `passregister()`, `passsendccb()`, `probedone()`, `proberequestbackoff()`, `proberequestdefaultnegotiation()`, `probeschedule()`, `ptctor()`, `ptinit()`, `ptoninvalidate()`, `sainit()`, `saoninvalidate()`, `saregister()`, `scsi_command_string()`, `scsi_sense_sbuf()`, `sesinit()`, `sesoninvalidate()`, `sesregister()`, `STAILQ_HEAD()`, `targbhdislun()`, `targbhdone()`, `targbhenlun()`, `targbhinit()`, `targdone()`, `targenable()`, `targendislun()`, `targfreeccb()`, `targreturnccb()`, `targsendccb()`, `targusermerge()`, `xpt_action()`, `xpt_announce_periph()`, `xpt_bus_register()`, `xpt_devise_transport()`, `xpt_done()`, `xpt_finishconfig()`, `xpt_merge_ccb()`, `xpt_run_dev_sendq()`, `xpt_scan_bus()`, `xpt_scan_lun()`, `xpt_set_transfer_settings()`, `xpt_start_tags()`, `xptaction()`, `xptconfigbuscountfunc()`, `xptconfigfunc()`, `xptioctl()`, `xptsetasynbusfunc()`, and `xptsetasynfunc()`.

6.33.2.4 `struct cam_path* ccb_hdr::path`

Definition at line 269 of file `cam_ccb.h`.

Referenced by `cam_error_string()`, `cam_freeze_devq()`, `cam_periph_error()`, `cam_periph_ioctl()`, `camisr()`, `camperiphdone()`, `camperiphfree()`, `camperiphscsisenseerror()`, `camperiphscsisistatuserror()`, `cd6byteworkaround()`, `cdasync()`, `cddone()`, `cderror()`, `chasync()`, `chdone()`, `cherror()`, `cmd6workaround()`, `daasync()`, `daclose()`, `dadone()`, `daerror()`, `dagetcapacity()`, `dashutdown()`, `passasync()`, `passerror()`, `probedone()`, `probestart()`, `ptasync()`, `ptdone()`, `pterror()`, `saasync()`, `sadone()`, `saerase()`, `saerror()`, `sardpos()`, `saretension()`, `sarewind()`, `sasetpos()`, `saspace()`, `sastart()`, `sawritefilemarks()`, `scsi_command_string()`, `scsi_read_write()`, `scsi_sense_sbuf()`, `ses_runcmd()`, `sesasync()`, `seserror()`, `targbhdone()`, `xpt_action()`, `xpt_bus_deregister()`, `xpt_done()`, `xpt_finishconfig()`, `xpt_freeze_devq()`, `xpt_polled_action()`, `xpt_release_ccb()`, `xpt_rescan()`, `xpt_run_dev_sendq()`, `xpt_scan_bus()`, `xpt_scan_lun()`, `xpt_set_transfer_settings()`, `xpt_setup_ccb()`, `xptaction()`, `xptioctl()`, and `xptscandone()`.

6.33.2.5 `path_id_t ccb_hdr::path_id`

Definition at line 270 of file `cam_ccb.h`.

Referenced by `cdregister()`, `targbhasync()`, `xpt_action()`, `xpt_scan_bus()`, `xpt_setup_ccb()`, and `xptioctl()`.

6.33.2.6 `camq_entry ccb_hdr::periph_links`

Definition at line 263 of file `cam_ccb.h`.

Referenced by `abort_all_pending()`, `cam_periph_getccb()`, and `targsendccb()`.

6.33.2.7 `ccb_ppriv_area ccb_hdr::periph_priv`

Definition at line 274 of file `cam_ccb.h`.

Referenced by `passsendccb()`.

6.33.2.8 `cam_pinfo ccb_hdr::pinfo`

Definition at line 260 of file `cam_ccb.h`.

Referenced by `cam_ccbq_insert_ccb()`, `cam_ccbq_remove_ccb()`, `cam_ccbq_send_ccb()`, `cam_periph_ccbwait()`, `camisr()`, `passioctl()`, `probedone()`, `xpt_action()`, `xpt_done()`, `xpt_scan_bus()`, `xpt_setup_ccb()`, and `xptioctl()`.

6.33.2.9 `u_int32_t ccb_hdr::retry_count`

Definition at line 264 of file `cam_ccb.h`.

Referenced by `cam_fill_csio()`, `cam_fill_ctio()`, `cam_periph_error()`, `camperiphdone()`, `camperiphscsistatuserror()`, `saerror()`, `sasetparams()`, `targreturnccb()`, `targusermerge()`, and `xpt_merge_ccb()`.

6.33.2.10 `camq_entry ccb_hdr::sim_links`

Definition at line 262 of file `cam_ccb.h`.

Referenced by `camisr()`, and `xpt_rescan()`.

6.33.2.11 `ccb_spriv_area ccb_hdr::sim_priv`

Definition at line 275 of file `cam_ccb.h`.

6.33.2.12 `u_int32_t ccb_hdr::status`

Definition at line 268 of file `cam_ccb.h`.

Referenced by `abort_all_pending()`, `cam_calc_geometry()`, `cam_error_string()`, `cam_periph_ccbwait()`, `cam_periph_error()`, `camisr()`, `camperiphdone()`, `cd6byteworkaround()`, `cddone()`, `cderror()`, `cdregister()`, `chdone()`, `chinit()`, `cmd6workaround()`, `daclose()`, `dadone()`, `dadump()`, `daerror()`, `dagetcapacity()`, `dainit()`, `daregister()`, `dasetgeom()`, `dashutdown()`, `dead_sim_action()`, `passinit()`, `probedone()`, `proberequestbackoff()`, `proberequestdefaultnegotiation()`, `ptdone()`, `ptinit()`, `sadone()`, `saerase()`, `saerror()`, `sagetparams()`, `sainit()`, `sardpos()`, `saretension()`, `sarewind()`, `sasetpos()`, `saspace()`, `sawritefilemarks()`, `scsi_command_string()`, `scsi_sense_sbuf()`, `ses_runcmd()`, `sesinit()`, `STAILQ_HEAD()`, `targbhdslun()`, `targbhdone()`, `targbhenlun()`, `targbhinit()`, `targbhstart()`, `targdone()`, `targenable()`, `targendislun()`, `targioctl()`, `targsendccb()`, `targusermerge()`, `xpt_action()`, `xpt_announce_periph()`, `xpt_finishconfig()`, `xpt_freeze_devq()`, `xpt_freeze_simq()`, `xpt_polled_action()`, `xpt_scan_bus()`, `xpt_scan_lun()`, `xpt_set_transfer_settings()`, `xptaction()`, `xpt-configfunc()`, and `xptioctl()`.

6.33.2.13 `target_id_t ccb_hdr::target_id`

Definition at line 271 of file `cam_ccb.h`.

Referenced by `cdregister()`, `targbhdone()`, `targbhstart()`, `xpt_action()`, `xpt_scan_bus()`, `xpt_setup_ccb()`, and `xptioctl()`.

6.33.2.14 [lun_id_t ccb_hdr::target_lun](#)

Definition at line 272 of file `cam_ccb.h`.

Referenced by `cdregister()`, `probedone()`, `targbhdone()`, `targbhstart()`, `xpt_action()`, `xpt_scan_bus()`, `xpt_setup_ccb()`, and `xptioctl()`.

6.33.2.15 [u_int32_t ccb_hdr::timeout](#)

Definition at line 276 of file `cam_ccb.h`.

Referenced by `cam_fill_csio()`, `cam_fill_ctio()`, `targusermerge()`, `xpt_merge_ccb()`, and `xpt_polled_action()`.

6.33.2.16 [struct callout_handle ccb_hdr::timeout_ch](#)

Definition at line 277 of file `cam_ccb.h`.

Referenced by `xpt_get_ccb()`.

6.33.2.17 [camq_entry ccb_hdr::xpt_links](#)

Definition at line 261 of file `cam_ccb.h`.

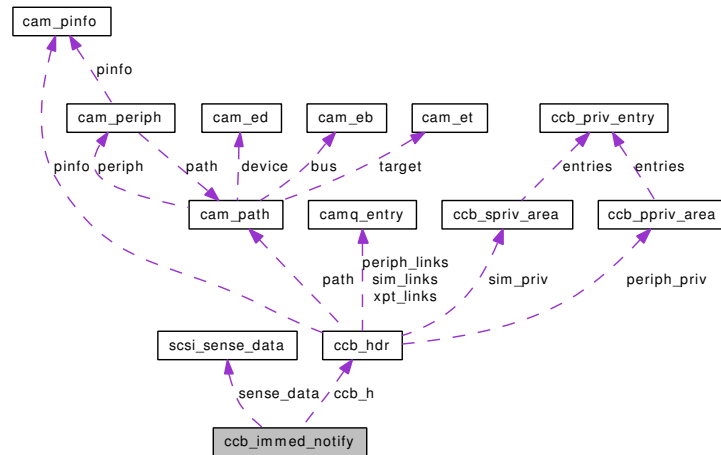
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/cam_ccb.h`

6.34 ccb_immed_notify Struct Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for ccb_immed_notify:



Data Fields

- [ccb_hdr](#) [ccb_h](#)
- [scsi_sense_data](#) [sense_data](#)
- [u_int8_t](#) [sense_len](#)
- [u_int8_t](#) [initiator_id](#)
- [u_int8_t](#) [message_args](#) [7]

6.34.1 Detailed Description

Definition at line 806 of file `cam_ccb.h`.

6.34.2 Field Documentation

6.34.2.1 struct `ccb_hdr` `ccb_immed_notify::ccb_h`

Definition at line 807 of file `cam_ccb.h`.

6.34.2.2 `u_int8_t` `ccb_immed_notify::initiator_id`

Definition at line 810 of file `cam_ccb.h`.

6.34.2.3 `u_int8_t` `ccb_immed_notify::message_args`[7]

Definition at line 811 of file `cam_ccb.h`.

6.34.2.4 struct `scsi_sense_data ccb_immed_notify::sense_data`

Definition at line 808 of file `cam_ccb.h`.

6.34.2.5 `u_int8_t ccb_immed_notify::sense_len`

Definition at line 809 of file `cam_ccb.h`.

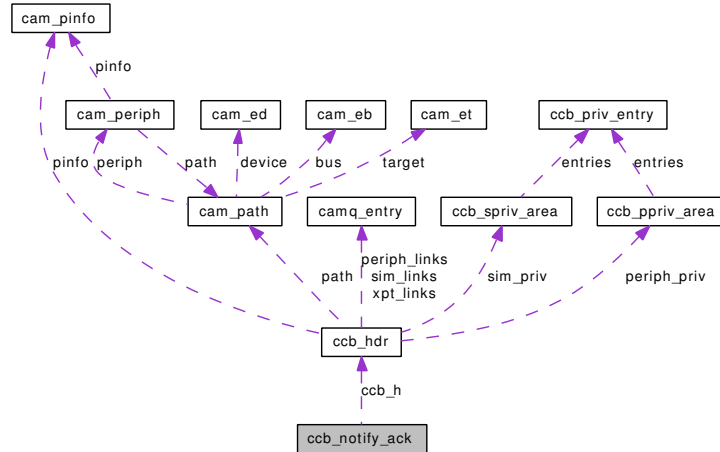
The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/cam_ccb.h](#)

6.35 ccb_notify_ack Struct Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for ccb_notify_ack:



Data Fields

- [ccb_hdr](#) [ccb_h](#)
- [u_int16_t](#) [seq_id](#)
- [u_int8_t](#) [event](#)

6.35.1 Detailed Description

Definition at line 814 of file `cam_ccb.h`.

6.35.2 Field Documentation

6.35.2.1 struct [ccb_hdr](#) [ccb_notify_ack::ccb_h](#)

Definition at line 815 of file `cam_ccb.h`.

6.35.2.2 [u_int8_t](#) [ccb_notify_ack::event](#)

Definition at line 817 of file `cam_ccb.h`.

6.35.2.3 [u_int16_t](#) [ccb_notify_ack::seq_id](#)

Definition at line 816 of file `cam_ccb.h`.

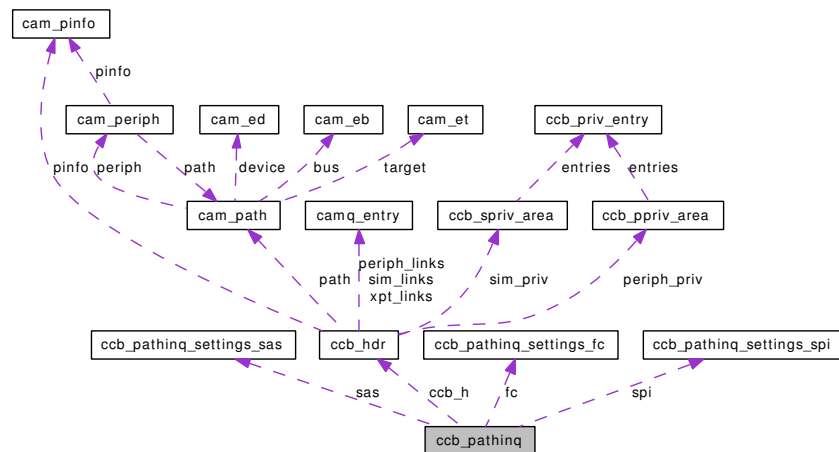
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/cam_ccb.h`

6.36 ccb_pathinq Struct Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for ccb_pathinq:



Data Fields

- [ccb_hdr](#) [ccb_h](#)
- [u_int8_t](#) [version_num](#)
- [u_int8_t](#) [hba_inquiry](#)
- [u_int8_t](#) [target_sprt](#)
- [u_int8_t](#) [hba_misc](#)
- [u_int16_t](#) [hba_eng_cnt](#)
- [u_int8_t](#) [vuhba_flags](#) [VUHBALEN]
- [u_int32_t](#) [max_target](#)
- [u_int32_t](#) [max_lun](#)
- [u_int32_t](#) [async_flags](#)
- [path_id_t](#) [hpath_id](#)
- [target_id_t](#) [initiator_id](#)
- [char](#) [sim_vid](#) [SIM_IDLEN]
- [char](#) [hba_vid](#) [HBA_IDLEN]
- [char](#) [dev_name](#) [DEV_IDLEN]
- [u_int32_t](#) [unit_number](#)
- [u_int32_t](#) [bus_id](#)
- [u_int32_t](#) [base_transfer_speed](#)
- [cam_proto](#) [protocol](#)
- [u_int](#) [protocol_version](#)
- [cam_xport](#) [transport](#)
- [u_int](#) [transport_version](#)
- [union](#) {
 - [ccb_pathinq_settings_spi](#) [spi](#)
 - [ccb_pathinq_settings_fc](#) [fc](#)
 - [ccb_pathinq_settings_sas](#) [sas](#)
 - [char](#) [ccb_pathinq_settings_opaque](#) [PATHINQ_SETTINGS_SIZE]
- [xport_specific](#)

6.36.1 Detailed Description

Definition at line 532 of file cam_ccb.h.

6.36.2 Field Documentation

6.36.2.1 `u_int32_t ccb_pathinq::async_flags`

Definition at line 543 of file cam_ccb.h.

6.36.2.2 `u_int32_t ccb_pathinq::base_transfer_speed`

Definition at line 551 of file cam_ccb.h.

Referenced by `xptaction()`.

6.36.2.3 `u_int32_t ccb_pathinq::bus_id`

Definition at line 550 of file cam_ccb.h.

Referenced by `xptaction()`.

6.36.2.4 `struct ccb_hdr ccb_pathinq::ccb_h`

Definition at line 533 of file cam_ccb.h.

Referenced by `cdregister()`, `daregister()`, `probeschedule()`, `targbhasync()`, `targenable()`, `xpt_announce_periph()`, `xpt_bus_register()`, `xpt_devise_transport()`, `xpt_scan_lun()`, `xpt_set_transfer_settings()`, `xptaction()`, `xptconfigbuscountfunc()`, and `xptsetasynbusfunc()`.

6.36.2.5 `char ccb_pathinq::ccb_pathinq_settings_opaque[PATHINQ_SETTINGS_SIZE]`

Definition at line 560 of file cam_ccb.h.

6.36.2.6 `char ccb_pathinq::dev_name[DEV_IDLEN]`

Definition at line 548 of file cam_ccb.h.

Referenced by `xptaction()`.

6.36.2.7 `struct ccb_pathinq_settings_fc ccb_pathinq::fc`

Definition at line 558 of file cam_ccb.h.

6.36.2.8 `u_int16_t ccb_pathinq::hba_eng_cnt`

Definition at line 538 of file cam_ccb.h.

Referenced by `xptaction()`.

6.36.2.9 `u_int8_t ccb_pathinq::hba_inquiry`

Definition at line 535 of file `cam_ccb.h`.

Referenced by `probeschedule()`, `xpt_set_transfer_settings()`, `xptaction()`, and `xptconfigfunc()`.

6.36.2.10 `u_int8_t ccb_pathinq::hba_misc`

Definition at line 537 of file `cam_ccb.h`.

Referenced by `probeschedule()`, `xpt_scan_bus()`, `xptaction()`, and `xptconfigfunc()`.

6.36.2.11 `char ccb_pathinq::hba_vid[HBA_IDLEN]`

Definition at line 547 of file `cam_ccb.h`.

Referenced by `xptaction()`.

6.36.2.12 `path_id_t ccb_pathinq::hpath_id`

Definition at line 544 of file `cam_ccb.h`.

6.36.2.13 `target_id_t ccb_pathinq::initiator_id`

Definition at line 545 of file `cam_ccb.h`.

Referenced by `xpt_scan_bus()`, and `xptaction()`.

6.36.2.14 `u_int32_t ccb_pathinq::max_lun`

Definition at line 542 of file `cam_ccb.h`.

Referenced by `xpt_scan_bus()`, and `xptaction()`.

6.36.2.15 `u_int32_t ccb_pathinq::max_target`

Definition at line 541 of file `cam_ccb.h`.

Referenced by `xpt_scan_bus()`, and `xptaction()`.

6.36.2.16 `cam_proto ccb_pathinq::protocol`

Definition at line 552 of file `cam_ccb.h`.

Referenced by `xptaction()`.

6.36.2.17 `u_int ccb_pathinq::protocol_version`

Definition at line 553 of file `cam_ccb.h`.

Referenced by `xptaction()`.

6.36.2.18 struct `ccb_pathinq::sas`

Definition at line 559 of file `cam_ccb.h`.

6.36.2.19 char `ccb_pathinq::sim_vid`[SIM_IDLEN]

Definition at line 546 of file `cam_ccb.h`.

Referenced by `xptaction()`.

6.36.2.20 struct `ccb_pathinq::spi`

Definition at line 557 of file `cam_ccb.h`.

6.36.2.21 u_int8_t `ccb_pathinq::target_sprt`

Definition at line 536 of file `cam_ccb.h`.

Referenced by `targbhasync()`, and `xptaction()`.

6.36.2.22 cam_xport `ccb_pathinq::transport`

Definition at line 554 of file `cam_ccb.h`.

Referenced by `xptaction()`.

6.36.2.23 u_int `ccb_pathinq::transport_version`

Definition at line 555 of file `cam_ccb.h`.

Referenced by `xptaction()`.

6.36.2.24 u_int32_t `ccb_pathinq::unit_number`

Definition at line 549 of file `cam_ccb.h`.

Referenced by `xptaction()`.

6.36.2.25 u_int8_t `ccb_pathinq::version_num`

Definition at line 534 of file `cam_ccb.h`.

Referenced by `xptaction()`.

6.36.2.26 u_int8_t `ccb_pathinq::vuhba_flags`[VUHBALEN]

Definition at line 540 of file `cam_ccb.h`.

6.36.2.27 union { ... } ccb_pathinq::xport_specific

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/cam_ccb.h](#)

6.37 ccb_pathinq_settings_fc Struct Reference

```
#include <cam_ccb.h>
```

Data Fields

- `u_int64_t` [wwnn](#)
- `u_int64_t` [wwpn](#)
- `u_int32_t` [port](#)
- `u_int32_t` [bitrate](#)

6.37.1 Detailed Description

Definition at line 521 of file `cam_ccb.h`.

6.37.2 Field Documentation

6.37.2.1 `u_int32_t` [ccb_pathinq_settings_fc::bitrate](#)

Definition at line 525 of file `cam_ccb.h`.

6.37.2.2 `u_int32_t` [ccb_pathinq_settings_fc::port](#)

Definition at line 524 of file `cam_ccb.h`.

6.37.2.3 `u_int64_t` [ccb_pathinq_settings_fc::wwnn](#)

Definition at line 522 of file `cam_ccb.h`.

6.37.2.4 `u_int64_t` [ccb_pathinq_settings_fc::wwpn](#)

Definition at line 523 of file `cam_ccb.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/cam_ccb.h`

6.38 ccb_pathinq_settings_sas Struct Reference

```
#include <cam_ccb.h>
```

Data Fields

- `u_int32_t` [bitrate](#)

6.38.1 Detailed Description

Definition at line 527 of file `cam_ccb.h`.

6.38.2 Field Documentation

6.38.2.1 `u_int32_t` [ccb_pathinq_settings_sas::bitrate](#)

Definition at line 528 of file `cam_ccb.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/cam_ccb.h`

6.39 ccb_pathinq_settings_spi Struct Reference

```
#include <cam_ccb.h>
```

Data Fields

- [u_int8_t ppr_options](#)

6.39.1 Detailed Description

Definition at line 518 of file `cam_ccb.h`.

6.39.2 Field Documentation

6.39.2.1 [u_int8_t ccb_pathinq_settings_spi::ppr_options](#)

Definition at line 519 of file `cam_ccb.h`.

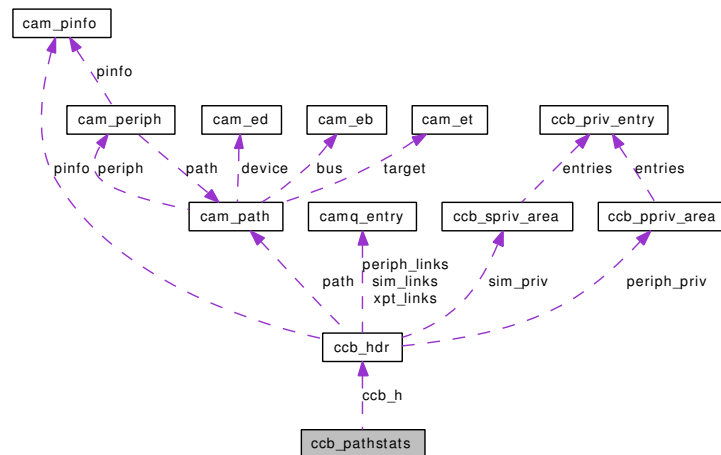
The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/cam_ccb.h](#)

6.40 ccb_pathstats Struct Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for ccb_pathstats:



Data Fields

- [ccb_hdr](#) [ccb_h](#)
- [timeval](#) [last_reset](#)

6.40.1 Detailed Description

Definition at line 565 of file `cam_ccb.h`.

6.40.2 Field Documentation

6.40.2.1 struct [ccb_hdr](#) [ccb_pathstats::ccb_h](#)

Definition at line 566 of file `cam_ccb.h`.

6.40.2.2 struct [timeval](#) [ccb_pathstats::last_reset](#)

Definition at line 567 of file `cam_ccb.h`.

Referenced by `xpt_action()`.

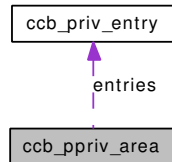
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/cam_ccb.h`

6.41 ccb_ppriv_area Union Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for ccb_ppriv_area:



Data Fields

- [ccb_priv_entry entries](#) [CCB_PERIPH_PRIV_SIZE]
- [u_int8_t bytes](#) [CCB_PERIPH_PRIV_SIZE *sizeof([ccb_priv_entry](#))]

6.41.1 Detailed Description

Definition at line 249 of file [cam_ccb.h](#).

6.41.2 Field Documentation

6.41.2.1 [u_int8_t ccb_ppriv_area::bytes](#)[CCB_PERIPH_PRIV_SIZE *sizeof([ccb_priv_entry](#))]

Definition at line 251 of file [cam_ccb.h](#).

6.41.2.2 [ccb_priv_entry ccb_ppriv_area::entries](#)[CCB_PERIPH_PRIV_SIZE]

Definition at line 250 of file [cam_ccb.h](#).

The documentation for this union was generated from the following file:

- [/usr/src/sys/cam/cam_ccb.h](#)

6.42 `ccb_priv_entry` Union Reference

```
#include <cam_ccb.h>
```

Data Fields

- `void * ptr`
- `u_long field`
- `u_int8_t bytes [sizeof(void *) > sizeof(u_long)?sizeof(void *):sizeof(u_long)]`

6.42.1 Detailed Description

Definition at line 242 of file `cam_ccb.h`.

6.42.2 Field Documentation

6.42.2.1 `u_int8_t ccb_priv_entry::bytes[sizeof(void *) > sizeof(u_long)?sizeof(void *):sizeof(u_long)]`

Definition at line 246 of file `cam_ccb.h`.

6.42.2.2 `u_long ccb_priv_entry::field`

Definition at line 244 of file `cam_ccb.h`.

6.42.2.3 `void* ccb_priv_entry::ptr`

Definition at line 243 of file `cam_ccb.h`.

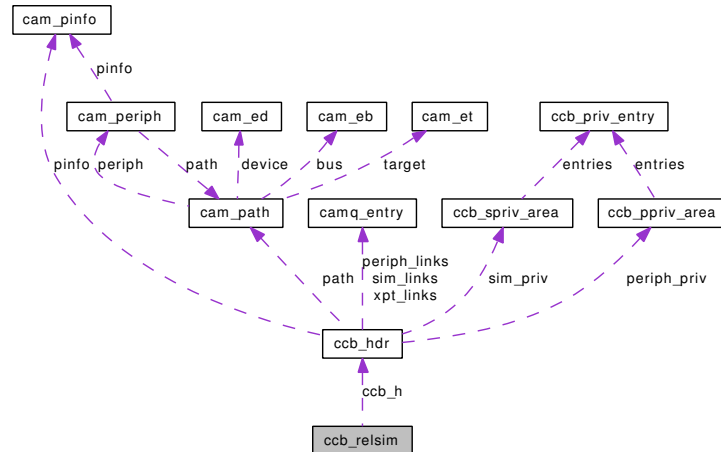
The documentation for this union was generated from the following file:

- `/usr/src/sys/cam/cam_ccb.h`

6.43 ccb_relsim Struct Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for ccb_relsim:



Data Fields

- [ccb_hdr](#) [ccb_h](#)
- [u_int32_t](#) [release_flags](#)
- [u_int32_t](#) [openings](#)
- [u_int32_t](#) [release_timeout](#)
- [u_int32_t](#) [qfrozen_cnt](#)

6.43.1 Detailed Description

Definition at line 629 of file `cam_ccb.h`.

6.43.2 Field Documentation

6.43.2.1 struct [ccb_hdr](#) [ccb_relsim::ccb_h](#)

Definition at line 630 of file `cam_ccb.h`.

Referenced by `cam_release_devq()`, `xpt_action()`, `xpt_set_transfer_settings()`, and `xpt_start_tags()`.

6.43.2.2 [u_int32_t](#) [ccb_relsim::openings](#)

Definition at line 636 of file `cam_ccb.h`.

Referenced by `cam_release_devq()`, `xpt_action()`, `xpt_set_transfer_settings()`, and `xpt_start_tags()`.

6.43.2.3 u_int32_t ccb_relsim::qfrozen_cnt

Definition at line 638 of file `cam_ccb.h`.

Referenced by `xpt_action()`, `xpt_set_transfer_settings()`, and `xpt_start_tags()`.

6.43.2.4 u_int32_t ccb_relsim::release_flags

Definition at line 631 of file `cam_ccb.h`.

Referenced by `cam_release_devq()`, `xpt_action()`, `xpt_set_transfer_settings()`, and `xpt_start_tags()`.

6.43.2.5 u_int32_t ccb_relsim::release_timeout

Definition at line 637 of file `cam_ccb.h`.

Referenced by `cam_release_devq()`, `xpt_action()`, `xpt_set_transfer_settings()`, and `xpt_start_tags()`.

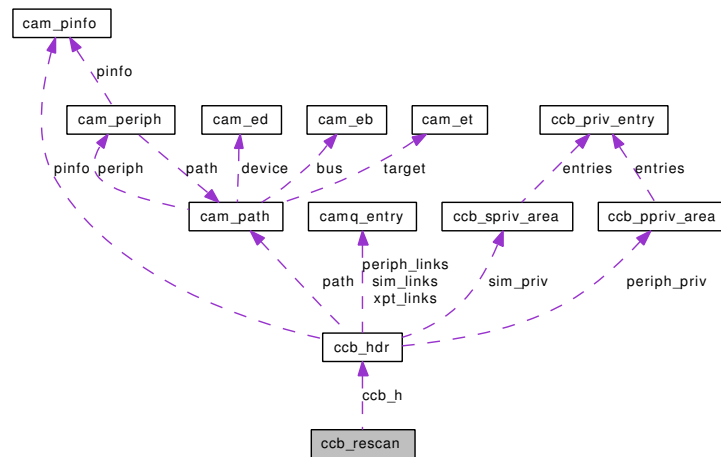
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/cam_ccb.h`

6.44 ccb_rescan Struct Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for ccb_rescan:



Data Fields

- [ccb_hdr ccb_h](#)
- [cam_flags flags](#)

6.44.1 Detailed Description

Definition at line 784 of file `cam_ccb.h`.

6.44.2 Field Documentation

6.44.2.1 struct `ccb_hdr ccb_rescan::ccb_h`

Definition at line 785 of file `cam_ccb.h`.

6.44.2.2 `cam_flags ccb_rescan::flags`

Definition at line 786 of file `cam_ccb.h`.

Referenced by `xpt_action()`, `xpt_finishconfig()`, `xpt_scan_bus()`, and `xpt_scan_lun()`.

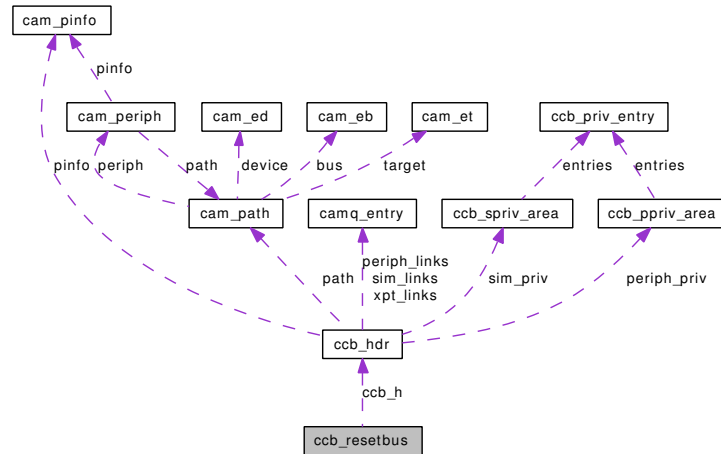
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/cam_ccb.h`

6.45 ccb_resetbus Struct Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for ccb_resetbus:



Data Fields

- [ccb_hdr](#) [ccb_h](#)

6.45.1 Detailed Description

Definition at line 684 of file `cam_ccb.h`.

6.45.2 Field Documentation

6.45.2.1 struct `ccb_hdr` `ccb_resetbus::ccb_h`

Definition at line 685 of file `cam_ccb.h`.

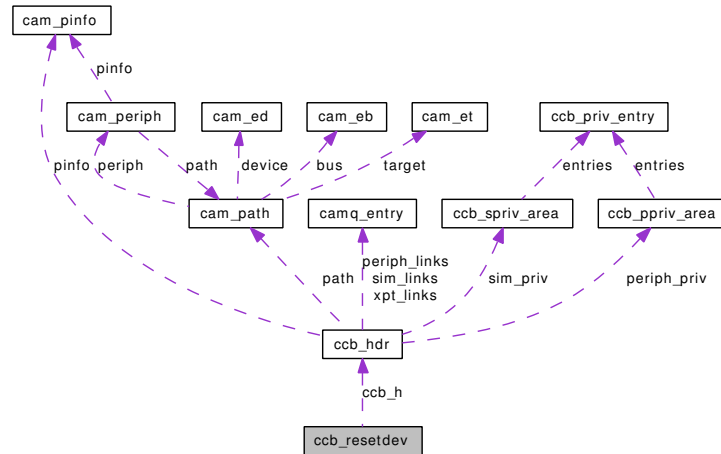
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/cam_ccb.h`

6.46 ccb_resetdev Struct Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for ccb_resetdev:



Data Fields

- [ccb_hdr](#) [ccb_h](#)

6.46.1 Detailed Description

Definition at line 689 of file `cam_ccb.h`.

6.46.2 Field Documentation

6.46.2.1 struct `ccb_hdr` `ccb_resetdev::ccb_h`

Definition at line 690 of file `cam_ccb.h`.

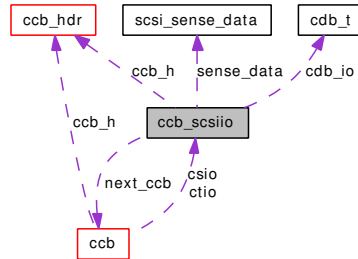
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/cam_ccb.h`

6.47 ccb_scsiio Struct Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for ccb_scsiio:



Data Fields

- [ccb_hdr](#) [ccb_h](#)
- [ccb](#) * [next_ccb](#)
- [u_int8_t](#) * [req_map](#)
- [u_int8_t](#) * [data_ptr](#)
- [u_int32_t](#) [dxfer_len](#)
- [scsi_sense_data](#) [sense_data](#)
- [u_int8_t](#) [sense_len](#)
- [u_int8_t](#) [cdb_len](#)
- [u_int16_t](#) [sglist_cnt](#)
- [u_int8_t](#) [scsi_status](#)
- [u_int8_t](#) [sense_resid](#)
- [u_int32_t](#) [resid](#)
- [cdb_t](#) [cdb_io](#)
- [u_int8_t](#) * [msg_ptr](#)
- [u_int16_t](#) [msg_len](#)
- [u_int8_t](#) [tag_action](#)
- [u_int](#) [tag_id](#)
- [u_int](#) [init_id](#)

6.47.1 Detailed Description

Definition at line 589 of file `cam_ccb.h`.

6.47.2 Field Documentation

6.47.2.1 struct `ccb_hdr` `ccb_scsiio::ccb_h`

Definition at line 590 of file `cam_ccb.h`.

Referenced by `cam_error_string()`, `cam_fill_csio()`, `cam_fill_ctio()`, `cddone()`, `daclose()`, `dadone()`, `dadump()`, `passdone()`, `probedone()`, `ptdone()`, `sadone()`, `saerror()`, `scsi_command_string()`, `scsi_read_write()`, and `scsi_sense_sbuf()`.

6.47.2.2 `cdb_t ccb_scsiio::cdb_io`

Definition at line 603 of file `cam_ccb.h`.

Referenced by `camperiphdone()`, `cd6byteworkaround()`, `cdpause()`, `cdplay()`, `cdplaymsf()`, `cdplaytracks()`, `cdreadsubchannel()`, `cdreadtoc()`, `cdsetspeed()`, `chdone()`, `chgetparams()`, `cmd6workaround()`, `saerror()`, `scsi_command_string()`, `scsi_erase()`, `scsi_exchange_medium()`, `scsi_initialize_element_status()`, `scsi_inquiry()`, `scsi_load_unload()`, `scsi_log_select()`, `scsi_log_sense()`, `scsi_mode_select_len()`, `scsi_mode_sense_len()`, `scsi_move_medium()`, `scsi_position_to_element()`, `scsi_prevent()`, `scsi_read_block_limits()`, `scsi_read_capacity()`, `scsi_read_capacity_16()`, `scsi_read_dvd_structure()`, `scsi_read_element_status()`, `scsi_read_position()`, `scsi_read_write()`, `scsi_report_key()`, `scsi_report_luns()`, `scsi_request_sense()`, `scsi_reserve_release_unit()`, `scsi_rewind()`, `scsi_sa_read_write()`, `scsi_send_key()`, `scsi_send_receive()`, `scsi_send_volume_tag()`, `scsi_set_position()`, `scsi_space()`, `scsi_start_stop()`, `scsi_synchronize_cache()`, `scsi_test_unit_ready()`, `scsi_write_filemarks()`, `ses_runcmd()`, and `xpt_action()`.

6.47.2.3 `u_int8_t ccb_scsiio::cdb_len`

Definition at line 598 of file `cam_ccb.h`.

Referenced by `cam_fill_csio()`, `cd6byteworkaround()`, `cmd6workaround()`, and `saerror()`.

6.47.2.4 `u_int8_t* ccb_scsiio::data_ptr`

Definition at line 593 of file `cam_ccb.h`.

Referenced by `cam_fill_csio()`, `cam_fill_ctio()`, `cam_periph_mapmem()`, `cam_periph_unmapmem()`, `cd6byteworkaround()`, `cddone()`, `chdone()`, `dadone()`, `probedone()`, and `probestart()`.

6.47.2.5 `u_int32_t ccb_scsiio::dxfer_len`

Definition at line 594 of file `cam_ccb.h`.

Referenced by `cam_fill_csio()`, `cam_fill_ctio()`, `cam_periph_mapmem()`, `cd6byteworkaround()`, and `saerror()`.

6.47.2.6 `u_int ccb_scsiio::init_id`

Definition at line 614 of file `cam_ccb.h`.

Referenced by `cam_fill_ctio()`.

6.47.2.7 `u_int16_t ccb_scsiio::msg_len`

Definition at line 605 of file `cam_ccb.h`.

6.47.2.8 `u_int8_t* ccb_scsiio::msg_ptr`

Definition at line 604 of file `cam_ccb.h`.

6.47.2.9 `union ccb* ccb_scsiio::next_ccb`

Definition at line 591 of file `cam_ccb.h`.

6.47.2.10 `u_int8_t* ccb_scsiio::req_map`

Definition at line 592 of file `cam_ccb.h`.

6.47.2.11 `u_int32_t ccb_scsiio::resid`

Definition at line 602 of file `cam_ccb.h`.

Referenced by `cddone()`, `cdreadvdstructure()`, `cdreportkey()`, `dadone()`, `ptdone()`, `sadone()`, `saerror()`, `ses_runcmd()`, and `xpt_action()`.

6.47.2.12 `u_int8_t ccb_scsiio::scsi_status`

Definition at line 600 of file `cam_ccb.h`.

Referenced by `cam_error_string()`, `cam_fill_ctio()`, `camperiphscsisstatuserror()`, `cddone()`, `cderror()`, `daclose()`, `daerror()`, `dashutdown()`, `scsi_status_string()`, and `xpt_action()`.

6.47.2.13 `struct scsi_sense_data ccb_scsiio::sense_data`

Definition at line 596 of file `cam_ccb.h`.

Referenced by `camperiphdone()`, `camperiphscsisenseerror()`, `cddone()`, `cderror()`, `daclose()`, `dadone()`, `daerror()`, `dashutdown()`, `saerror()`, `scsi_error_action()`, `scsi_sense_sbuf()`, and `targbhstart()`.

6.47.2.14 `u_int8_t ccb_scsiio::sense_len`

Definition at line 597 of file `cam_ccb.h`.

Referenced by `cam_fill_csio()`, and `targbhstart()`.

6.47.2.15 `u_int8_t ccb_scsiio::sense_resid`

Definition at line 601 of file `cam_ccb.h`.

Referenced by `xpt_action()`.

6.47.2.16 `u_int16_t ccb_scsiio::sglist_cnt`

Definition at line 599 of file `cam_ccb.h`.

6.47.2.17 `u_int8_t ccb_scsiio::tag_action`

Definition at line 606 of file `cam_ccb.h`.

Referenced by `cam_fill_csio()`, `cam_fill_ctio()`, and `xpt_run_dev_sendq()`.

6.47.2.18 `u_int ccb_scsiio::tag_id`

Definition at line 613 of file `cam_ccb.h`.

Referenced by `cam_fill_ctio()`.

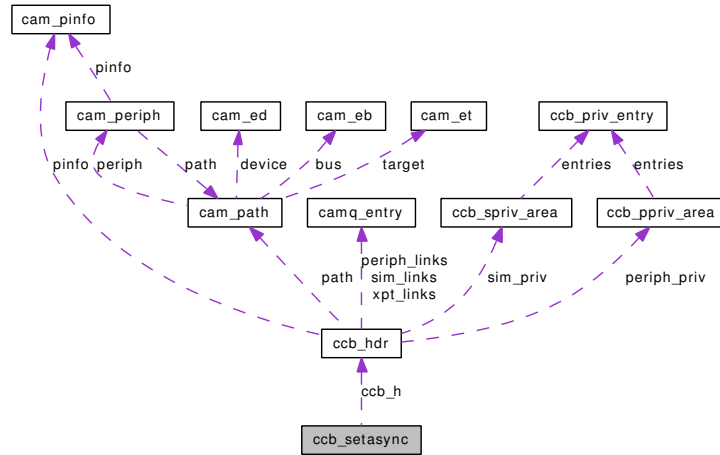
The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/cam_ccb.h](#)

6.48 ccb_setasync Struct Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for ccb_setasync:



Data Fields

- [ccb_hdr](#) [ccb_h](#)
- [u_int32_t](#) [event_enable](#)
- [ac_callback_t](#) * [callback](#)
- [void](#) * [callback_arg](#)

6.48.1 Detailed Description

Definition at line 662 of file `cam_ccb.h`.

6.48.2 Field Documentation

6.48.2.1 [ac_callback_t](#)* [ccb_setasync::callback](#)

Definition at line 665 of file `cam_ccb.h`.

Referenced by `cdoninvalidate()`, `cdregister()`, `chinit()`, `choninvalidate()`, `chregister()`, `dainit()`, `daoninvalidate()`, `daregister()`, `passinit()`, `passoninvalidate()`, `passregister()`, `ptctor()`, `ptinit()`, `ptoninvalidate()`, `sainit()`, `saoninvalidate()`, `saregister()`, `sesinit()`, `sesoninvalidate()`, `sesregister()`, `STAILQ_HEAD()`, `targbhinit()`, and `xpt_action()`.

6.48.2.2 [void](#)* [ccb_setasync::callback_arg](#)

Definition at line 666 of file `cam_ccb.h`.

Referenced by `cdoninvalidate()`, `cdregister()`, `chinit()`, `choninvalidate()`, `chregister()`, `dainit()`, `daoninvalidate()`, `daregister()`, `passinit()`, `passoninvalidate()`, `passregister()`, `ptctor()`, `ptinit()`, `ptoninvalidate()`, `sainit()`,

saoninvalidate(), saregister(), sesinit(), sesoninvalidate(), sesregister(), STAILQ_HEAD(), targbhinit(), and xpt_action().

6.48.2.3 struct [ccb_hdr](#) [ccb_setasync::ccb_h](#)

Definition at line 663 of file [cam_ccb.h](#).

Referenced by [cdoninvalidate\(\)](#), [cdregister\(\)](#), [chinit\(\)](#), [choninvalidate\(\)](#), [chregister\(\)](#), [dainit\(\)](#), [daoninvalidate\(\)](#), [daregister\(\)](#), [passinit\(\)](#), [passoninvalidate\(\)](#), [passregister\(\)](#), [ptctor\(\)](#), [ptinit\(\)](#), [ptoninvalidate\(\)](#), [sainit\(\)](#), [saoninvalidate\(\)](#), [saregister\(\)](#), [sesinit\(\)](#), [sesoninvalidate\(\)](#), [sesregister\(\)](#), [STAILQ_HEAD\(\)](#), [targbhinit\(\)](#), and [xpt_action\(\)](#).

6.48.2.4 u_int32_t [ccb_setasync::event_enable](#)

Definition at line 664 of file [cam_ccb.h](#).

Referenced by [cdoninvalidate\(\)](#), [cdregister\(\)](#), [chinit\(\)](#), [choninvalidate\(\)](#), [chregister\(\)](#), [dainit\(\)](#), [daoninvalidate\(\)](#), [daregister\(\)](#), [passinit\(\)](#), [passoninvalidate\(\)](#), [passregister\(\)](#), [ptctor\(\)](#), [ptinit\(\)](#), [ptoninvalidate\(\)](#), [sainit\(\)](#), [saoninvalidate\(\)](#), [saregister\(\)](#), [sesinit\(\)](#), [sesoninvalidate\(\)](#), [sesregister\(\)](#), [STAILQ_HEAD\(\)](#), [targbhinit\(\)](#), and [xpt_action\(\)](#).

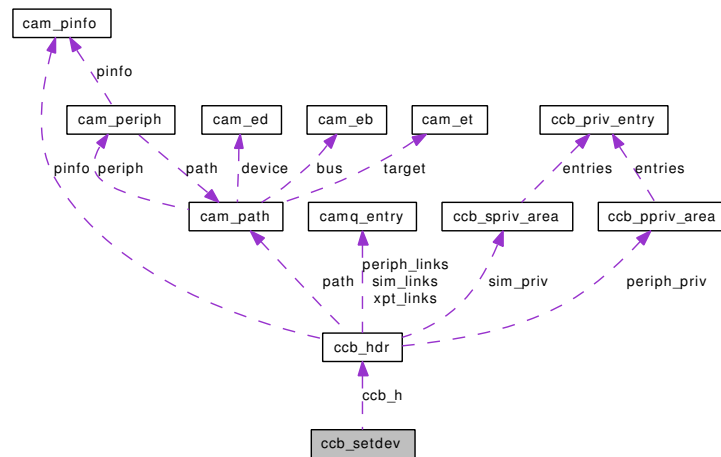
The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/cam_ccb.h](#)

6.49 ccb_setdev Struct Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for ccb_setdev:



Data Fields

- [ccb_hdr](#) [ccb_h](#)
- [u_int8_t](#) [dev_type](#)

6.49.1 Detailed Description

Definition at line 670 of file `cam_ccb.h`.

6.49.2 Field Documentation

6.49.2.1 struct [ccb_hdr](#) [ccb_setdev::ccb_h](#)

Definition at line 671 of file `cam_ccb.h`.

6.49.2.2 [u_int8_t](#) [ccb_setdev::dev_type](#)

Definition at line 672 of file `cam_ccb.h`.

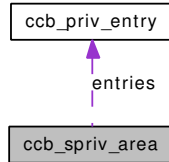
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/cam_ccb.h`

6.50 ccb_spriv_area Union Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for ccb_spriv_area:



Data Fields

- [ccb_priv_entry entries](#) [CCB_SIM_PRIV_SIZE]
- [u_int8_t bytes](#) [CCB_SIM_PRIV_SIZE *sizeof(ccb_priv_entry)]

6.50.1 Detailed Description

Definition at line 254 of file cam_ccb.h.

6.50.2 Field Documentation

6.50.2.1 [u_int8_t ccb_spriv_area::bytes](#)[CCB_SIM_PRIV_SIZE *sizeof(ccb_priv_entry)]

Definition at line 256 of file cam_ccb.h.

6.50.2.2 [ccb_priv_entry ccb_spriv_area::entries](#)[CCB_SIM_PRIV_SIZE]

Definition at line 255 of file cam_ccb.h.

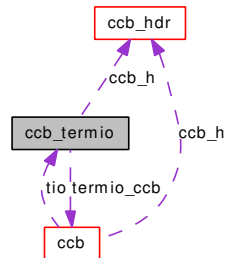
The documentation for this union was generated from the following file:

- [/usr/src/sys/cam/cam_ccb.h](#)

6.51 ccb_termio Struct Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for ccb_termio:



Data Fields

- `ccb_hdr ccb_h`
- `ccb * termio_ccb`

6.51.1 Detailed Description

Definition at line 694 of file `cam_ccb.h`.

6.51.2 Field Documentation

6.51.2.1 struct `ccb_hdr ccb_termio::ccb_h`

Definition at line 695 of file `cam_ccb.h`.

6.51.2.2 union `ccb* ccb_termio::termio_ccb`

Definition at line 696 of file `cam_ccb.h`.

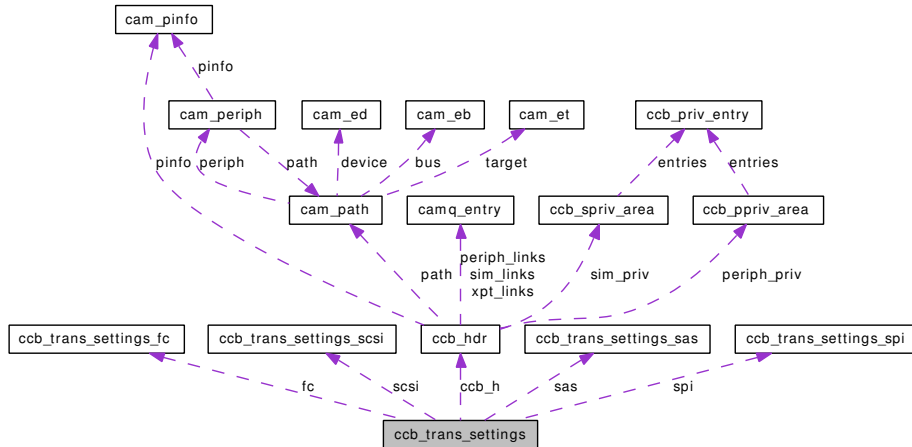
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/cam_ccb.h`

6.52 ccb_trans_settings Struct Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for ccb_trans_settings:



Data Fields

- `ccb_hdr` `ccb_h`
- `cts_type` type
- `cam_proto` protocol
- `u_int` `protocol_version`
- `cam_xport` transport
- `u_int` `transport_version`
- union {
 - `u_int` `valid`
 - `ccb_trans_settings_scsi` `scsi`
- } `proto_specific`
- union {
 - `u_int` `valid`
 - `ccb_trans_settings_spi` `spi`
 - `ccb_trans_settings_fc` `fc`
 - `ccb_trans_settings_sas` `sas`
- } `xport_specific`

6.52.1 Detailed Description

Definition at line 748 of file `cam_ccb.h`.

6.52.2 Field Documentation

6.52.2.1 struct [ccb_hdr](#) [ccb_trans_settings::ccb_h](#)

Definition at line 749 of file `cam_ccb.h`.

Referenced by `proberequestdefaultnegotiation()`, `xpt_announce_periph()`, `xpt_devise_transport()`, `xpt_set_transfer_settings()`, and `xpt_toggle_tags()`.

6.52.2.2 struct [ccb_trans_settings_fc](#) [ccb_trans_settings::fc](#)

Definition at line 762 of file `cam_ccb.h`.

6.52.2.3 union { ... } [ccb_trans_settings::proto_specific](#)

Referenced by `xpt_devise_transport()`, `xpt_set_transfer_settings()`, and `xpt_toggle_tags()`.

6.52.2.4 [cam_proto](#) [ccb_trans_settings::protocol](#)

Definition at line 751 of file `cam_ccb.h`.

Referenced by `xpt_devise_transport()`, `xpt_set_transfer_settings()`, and `xpt_toggle_tags()`.

6.52.2.5 [u_int](#) [ccb_trans_settings::protocol_version](#)

Definition at line 752 of file `cam_ccb.h`.

Referenced by `xpt_devise_transport()`, `xpt_set_transfer_settings()`, and `xpt_toggle_tags()`.

6.52.2.6 struct [ccb_trans_settings_sas](#) [ccb_trans_settings::sas](#)

Definition at line 763 of file `cam_ccb.h`.

6.52.2.7 struct [ccb_trans_settings_scsi](#) [ccb_trans_settings::scsi](#)

Definition at line 757 of file `cam_ccb.h`.

Referenced by `xpt_set_transfer_settings()`, and `xpt_toggle_tags()`.

6.52.2.8 struct [ccb_trans_settings_spi](#) [ccb_trans_settings::spi](#)

Definition at line 761 of file `cam_ccb.h`.

Referenced by `xpt_set_transfer_settings()`.

6.52.2.9 [cam_xport](#) [ccb_trans_settings::transport](#)

Definition at line 753 of file `cam_ccb.h`.

Referenced by `xpt_devise_transport()`, `xpt_set_transfer_settings()`, and `xpt_toggle_tags()`.

6.52.2.10 `u_int ccb_trans_settings::transport_version`

Definition at line 754 of file `cam_ccb.h`.

Referenced by `xpt_devise_transport()`, `xpt_set_transfer_settings()`, and `xpt_toggle_tags()`.

6.52.2.11 `cts_type ccb_trans_settings::type`

Definition at line 750 of file `cam_ccb.h`.

Referenced by `proberequestdefaultnegotiation()`, `xpt_announce_periph()`, `xpt_devise_transport()`, and `xpt_set_transfer_settings()`.

6.52.2.12 `u_int ccb_trans_settings::valid`

Definition at line 756 of file `cam_ccb.h`.

Referenced by `xpt_devise_transport()`.

6.52.2.13 `union { ... } ccb_trans_settings::xport_specific`

Referenced by `xpt_devise_transport()`, and `xpt_set_transfer_settings()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/cam_ccb.h`

6.53 `ccb_trans_settings_fc` Struct Reference

```
#include <cam_ccb.h>
```

Data Fields

- `u_int` [valid](#)
- `u_int64_t` [wwnn](#)
- `u_int64_t` [wwpn](#)
- `u_int32_t` [port](#)
- `u_int32_t` [bitrate](#)

6.53.1 Detailed Description

Definition at line 728 of file `cam_ccb.h`.

6.53.2 Field Documentation

6.53.2.1 `u_int32_t` [ccb_trans_settings_fc::bitrate](#)

Definition at line 737 of file `cam_ccb.h`.

Referenced by `xpt_announce_periph()`.

6.53.2.2 `u_int32_t` [ccb_trans_settings_fc::port](#)

Definition at line 736 of file `cam_ccb.h`.

Referenced by `xpt_announce_periph()`.

6.53.2.3 `u_int` [ccb_trans_settings_fc::valid](#)

Definition at line 729 of file `cam_ccb.h`.

Referenced by `xpt_announce_periph()`.

6.53.2.4 `u_int64_t` [ccb_trans_settings_fc::wwnn](#)

Definition at line 734 of file `cam_ccb.h`.

Referenced by `xpt_announce_periph()`.

6.53.2.5 `u_int64_t` [ccb_trans_settings_fc::wwpn](#)

Definition at line 735 of file `cam_ccb.h`.

Referenced by `xpt_announce_periph()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/cam_ccb.h`

6.54 ccb_trans_settings_sas Struct Reference

```
#include <cam_ccb.h>
```

Data Fields

- [u_int valid](#)
- [u_int32_t bitrate](#)

6.54.1 Detailed Description

Definition at line 740 of file `cam_ccb.h`.

6.54.2 Field Documentation

6.54.2.1 [u_int32_t ccb_trans_settings_sas::bitrate](#)

Definition at line 743 of file `cam_ccb.h`.

Referenced by `xpt_announce_periph()`.

6.54.2.2 [u_int ccb_trans_settings_sas::valid](#)

Definition at line 741 of file `cam_ccb.h`.

Referenced by `xpt_announce_periph()`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/cam_ccb.h](#)

6.55 `ccb_trans_settings_scsi` Struct Reference

```
#include <cam_ccb.h>
```

Data Fields

- `u_int` [valid](#)
- `u_int` [flags](#)

6.55.1 Detailed Description

Definition at line 704 of file `cam_ccb.h`.

6.55.2 Field Documentation

6.55.2.1 `u_int` [ccb_trans_settings_scsi::flags](#)

Definition at line 708 of file `cam_ccb.h`.

Referenced by `xpt_set_transfer_settings()`, and `xpt_toggle_tags()`.

6.55.2.2 `u_int` [ccb_trans_settings_scsi::valid](#)

Definition at line 706 of file `cam_ccb.h`.

Referenced by `xpt_set_transfer_settings()`, and `xpt_toggle_tags()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/cam_ccb.h`

6.56 ccb_trans_settings_spi Struct Reference

```
#include <cam_ccb.h>
```

Data Fields

- [u_int valid](#)
- [u_int flags](#)
- [u_int sync_period](#)
- [u_int sync_offset](#)
- [u_int bus_width](#)
- [u_int ppr_options](#)

6.56.1 Detailed Description

Definition at line 712 of file cam_ccb.h.

6.56.2 Field Documentation

6.56.2.1 [u_int ccb_trans_settings_spi::bus_width](#)

Definition at line 724 of file cam_ccb.h.

Referenced by [xpt_announce_periph\(\)](#), and [xpt_set_transfer_settings\(\)](#).

6.56.2.2 [u_int ccb_trans_settings_spi::flags](#)

Definition at line 720 of file cam_ccb.h.

Referenced by [xpt_set_transfer_settings\(\)](#).

6.56.2.3 [u_int ccb_trans_settings_spi::ppr_options](#)

Definition at line 725 of file cam_ccb.h.

Referenced by [xpt_announce_periph\(\)](#), and [xpt_set_transfer_settings\(\)](#).

6.56.2.4 [u_int ccb_trans_settings_spi::sync_offset](#)

Definition at line 723 of file cam_ccb.h.

Referenced by [proberequestbackoff\(\)](#), [xpt_announce_periph\(\)](#), and [xpt_set_transfer_settings\(\)](#).

6.56.2.5 [u_int ccb_trans_settings_spi::sync_period](#)

Definition at line 722 of file cam_ccb.h.

Referenced by [proberequestbackoff\(\)](#), [xpt_announce_periph\(\)](#), and [xpt_set_transfer_settings\(\)](#).

6.56.2.6 `u_int ccb_trans_settings_spi::valid`

Definition at line 714 of file `cam_ccb.h`.

Referenced by `proberequestbackoff()`, `xpt_announce_periph()`, and `xpt_set_transfer_settings()`.

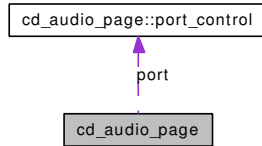
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/cam_ccb.h`

6.57 cd_audio_page Struct Reference

```
#include <scsi_cd.h>
```

Collaboration diagram for cd_audio_page:



Data Fields

- [u_int8_t page_code](#)
- [u_int8_t param_len](#)
- [u_int8_t flags](#)
- [u_int8_t unused](#) [2]
- [u_int8_t format_lba](#)
- [u_int8_t lb_per_sec](#) [2]
- [cd_audio_page::port_control port](#) [4]

Data Structures

- [struct port_control](#)

6.57.1 Detailed Description

Definition at line 658 of file `scsi_cd.h`.

6.57.2 Field Documentation

6.57.2.1 u_int8_t cd_audio_page::flags

Definition at line 665 of file `scsi_cd.h`.

Referenced by `cdioctl()`.

6.57.2.2 u_int8_t cd_audio_page::format_lba

Definition at line 669 of file `scsi_cd.h`.

6.57.2.3 u_int8_t cd_audio_page::lb_per_sec[2]

Definition at line 672 of file `scsi_cd.h`.

6.57.2.4 `u_int8_t cd_audio_page::page_code`

Definition at line 660 of file `scsi_cd.h`.

6.57.2.5 `u_int8_t cd_audio_page::param_len`

Definition at line 664 of file `scsi_cd.h`.

6.57.2.6 `struct cd_audio_page::port_control cd_audio_page::port[4]`

Referenced by `cdioctl()`.

6.57.2.7 `u_int8_t cd_audio_page::unused[2]`

Definition at line 668 of file `scsi_cd.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.h`

6.58 `cd_audio_page::port_control` Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- `u_int8_t channels`
- `u_int8_t volume`

6.58.1 Detailed Description

Definition at line 673 of file `scsi_cd.h`.

6.58.2 Field Documentation

6.58.2.1 `u_int8_t cd_audio_page::port_control::channels`

Definition at line 675 of file `scsi_cd.h`.

Referenced by `cdioctl()`.

6.58.2.2 `u_int8_t cd_audio_page::port_control::volume`

Definition at line 683 of file `scsi_cd.h`.

Referenced by `cdioctl()`.

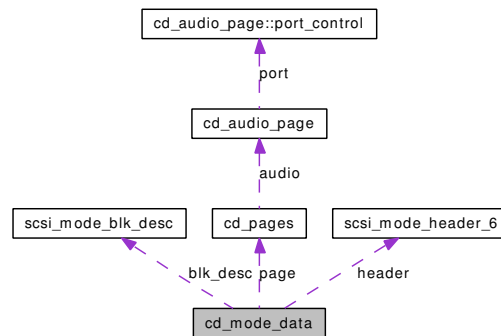
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.h`

6.59 cd_mode_data Struct Reference

```
#include <scsi_cd.h>
```

Collaboration diagram for cd_mode_data:



Data Fields

- [scsi_mode_header_6](#) header
- [scsi_mode_blk_desc](#) blk_desc
- [cd_pages](#) page

6.59.1 Detailed Description

Definition at line 701 of file `scsi_cd.h`.

6.59.2 Field Documentation

6.59.2.1 struct [scsi_mode_blk_desc](#) `cd_mode_data::blk_desc`

Definition at line 704 of file `scsi_cd.h`.

6.59.2.2 struct [scsi_mode_header_6](#) `cd_mode_data::header`

Definition at line 703 of file `scsi_cd.h`.

6.59.2.3 union [cd_pages](#) `cd_mode_data::page`

Definition at line 705 of file `scsi_cd.h`.

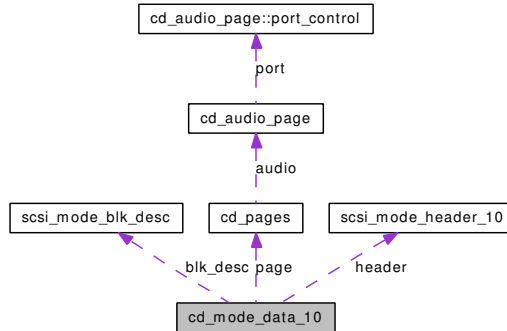
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.h`

6.60 cd_mode_data_10 Struct Reference

```
#include <scsi_cd.h>
```

Collaboration diagram for cd_mode_data_10:



Data Fields

- [scsi_mode_header_10](#) header
- [scsi_mode_blk_desc](#) blk_desc
- [cd_pages](#) page

6.60.1 Detailed Description

Definition at line 694 of file `scsi_cd.h`.

6.60.2 Field Documentation

6.60.2.1 struct [scsi_mode_blk_desc](#) `cd_mode_data_10::blk_desc`

Definition at line 697 of file `scsi_cd.h`.

6.60.2.2 struct [scsi_mode_header_10](#) `cd_mode_data_10::header`

Definition at line 696 of file `scsi_cd.h`.

6.60.2.3 union [cd_pages](#) `cd_mode_data_10::page`

Definition at line 698 of file `scsi_cd.h`.

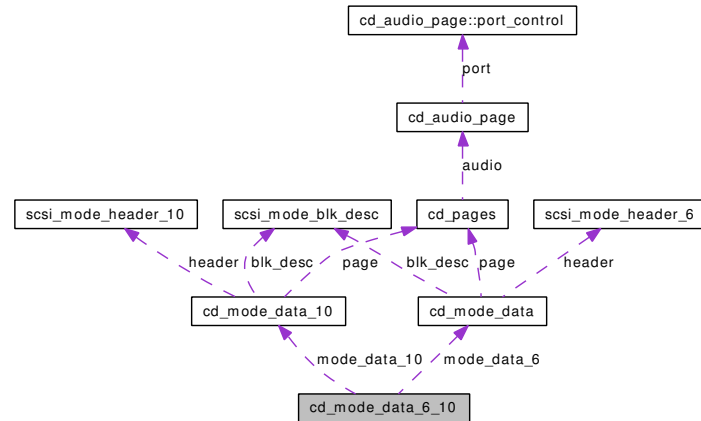
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.h`

6.61 cd_mode_data_6_10 Union Reference

```
#include <scsi_cd.h>
```

Collaboration diagram for cd_mode_data_6_10:



Data Fields

- [cd_mode_data mode_data_6](#)
- [cd_mode_data_10 mode_data_10](#)

6.61.1 Detailed Description

Definition at line 708 of file scsi_cd.h.

6.61.2 Field Documentation

6.61.2.1 struct [cd_mode_data_10](#) [cd_mode_data_6_10::mode_data_10](#)

Definition at line 711 of file scsi_cd.h.

6.61.2.2 struct [cd_mode_data](#) [cd_mode_data_6_10::mode_data_6](#)

Definition at line 710 of file scsi_cd.h.

The documentation for this union was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_cd.h](#)

6.62 cd_mode_params Struct Reference

```
#include <scsi_cd.h>
```

Public Member Functions

- [STAILQ_ENTRY \(cd_mode_params\)](#) links

Data Fields

- int [cdb_size](#)
- int [alloc_len](#)
- u_int8_t * [mode_buf](#)

6.62.1 Detailed Description

Definition at line 714 of file `scsi_cd.h`.

6.62.2 Member Function Documentation

6.62.2.1 [cd_mode_params::STAILQ_ENTRY \(cd_mode_params\)](#)

6.62.3 Field Documentation

6.62.3.1 int [cd_mode_params::alloc_len](#)

Definition at line 718 of file `scsi_cd.h`.

Referenced by `cd6byteworkaround()`, `cdgetmode()`, `cdioctl()`, and `cdsetmode()`.

6.62.3.2 int [cd_mode_params::cdb_size](#)

Definition at line 717 of file `scsi_cd.h`.

Referenced by `cd6byteworkaround()`, `cdgetmode()`, `cdgetpage()`, and `cdsetmode()`.

6.62.3.3 u_int8_t* [cd_mode_params::mode_buf](#)

Definition at line 719 of file `scsi_cd.h`.

Referenced by `cd6byteworkaround()`, `cdgetmode()`, `cdgetpage()`, `cdioctl()`, and `cdsetmode()`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_cd.h](#)

6.63 `cd_page_sizes` Struct Reference

Data Fields

- int `page`
- int `page_size`

6.63.1 Detailed Description

Definition at line 165 of file `scsi_cd.c`.

6.63.2 Field Documentation

6.63.2.1 int `cd_page_sizes::page`

Definition at line 166 of file `scsi_cd.c`.

6.63.2.2 int `cd_page_sizes::page_size`

Definition at line 167 of file `scsi_cd.c`.

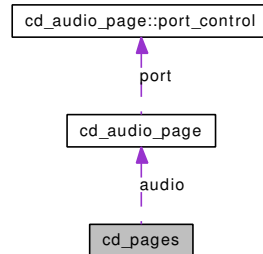
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.c`

6.64 cd_pages Union Reference

```
#include <scsi_cd.h>
```

Collaboration diagram for cd_pages:



Data Fields

- [cd_audio_page audio](#)

6.64.1 Detailed Description

Definition at line 689 of file scsi_cd.h.

6.64.2 Field Documentation

6.64.2.1 struct [cd_audio_page](#) [cd_pages::audio](#)

Definition at line 691 of file scsi_cd.h.

Referenced by [cdioctl\(\)](#).

The documentation for this union was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_cd.h](#)

6.65 cd_params Struct Reference

Data Fields

- [u_int32_t blksize](#)
- [u_long disksize](#)

6.65.1 Detailed Description

Definition at line 80 of file `scsi_cd.c`.

6.65.2 Field Documentation

6.65.2.1 [u_int32_t cd_params::blksize](#)

Definition at line 81 of file `scsi_cd.c`.

Referenced by `cddone()`.

6.65.2.2 [u_long cd_params::disksize](#)

Definition at line 82 of file `scsi_cd.c`.

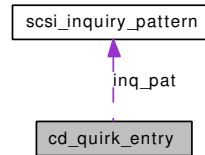
Referenced by `cddone()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.c`

6.66 cd_quirk_entry Struct Reference

Collaboration diagram for cd_quirk_entry:



Data Fields

- [scsi_inquiry_pattern inq_pat](#)
- [cd_quirks quirks](#)

6.66.1 Detailed Description

Definition at line 175 of file scsi_cd.c.

6.66.2 Field Documentation

6.66.2.1 struct [scsi_inquiry_pattern](#) [cd_quirk_entry::inq_pat](#)

Definition at line 176 of file scsi_cd.c.

6.66.2.2 [cd_quirks](#) [cd_quirk_entry::quirks](#)

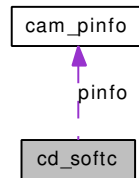
Definition at line 177 of file scsi_cd.c.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_cd.c](#)

6.67 cd_softc Struct Reference

Collaboration diagram for cd_softc:



Data Fields

- [cam_pinfo pinfo](#)
- [cd_state state](#)
- volatile [cd_flags flags](#)
- [bio_queue_head bio_queue](#)

6.67.1 Detailed Description

Definition at line 142 of file scsi_cd.c.

6.67.2 Field Documentation

6.67.2.1 struct [bio_queue_head](#) [cd_softc::bio_queue](#)

Definition at line 146 of file scsi_cd.c.

Referenced by [cdchangerschedule\(\)](#), [cddone\(\)](#), [cdrunchangerqueue\(\)](#), [cdshorttimeout\(\)](#), [cdstart\(\)](#), and [cdstrategy\(\)](#).

6.67.2.2 volatile [cd_flags](#) [cd_softc::flags](#)

Definition at line 145 of file scsi_cd.c.

Referenced by [cdasync\(\)](#), [cdchangerschedule\(\)](#), [cdcheckmedia\(\)](#), [cdcleanup\(\)](#), [cdcloses\(\)](#), [cddone\(\)](#), [cdgetccb\(\)](#), [cdioctl\(\)](#), [cdopen\(\)](#), [cdprevent\(\)](#), [cdregister\(\)](#), [cdruncb\(\)](#), [cdrunchangerqueue\(\)](#), [cdschedule\(\)](#), [cdstart\(\)](#), [cdstrategy\(\)](#), and [cdsysctlnit\(\)](#).

6.67.2.3 [cam_pinfo](#) [cd_softc::pinfo](#)

Definition at line 143 of file scsi_cd.c.

Referenced by [cdgetccb\(\)](#), [cdregister\(\)](#), [cdrunchangerqueue\(\)](#), and [cdschedule\(\)](#).

6.67.2.4 [cd_state](#) [cd_softc::state](#)

Definition at line 144 of file scsi_cd.c.

Referenced by [cddone\(\)](#), and [cdstart\(\)](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_cd.c](#)

6.68 cd_toc_single Struct Reference

Data Fields

- `ioc_toc_header` [header](#)
- `cd_toc_entry` [entry](#)

6.68.1 Detailed Description

Definition at line 132 of file `scsi_cd.c`.

6.68.2 Field Documentation

6.68.2.1 struct cd_toc_entry [cd_toc_single::entry](#)

Definition at line 134 of file `scsi_cd.c`.

6.68.2.2 struct ioc_toc_header [cd_toc_single::header](#)

Definition at line 133 of file `scsi_cd.c`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.c`

6.69 cd_tocdata Struct Reference

Data Fields

- `ioc_toc_header` [header](#)
- `cd_toc_entry` [entries](#) [100]

6.69.1 Detailed Description

Definition at line 127 of file `scsi_cd.c`.

6.69.2 Field Documentation

6.69.2.1 struct cd_toc_entry [cd_tocdata::entries](#)[100]

Definition at line 129 of file `scsi_cd.c`.

6.69.2.2 struct ioc_toc_header [cd_tocdata::header](#)

Definition at line 128 of file `scsi_cd.c`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.c`

6.70 cdb_t Union Reference

```
#include <cam_ccb.h>
```

Data Fields

- `u_int8_t * cdb_ptr`
- `u_int8_t cdb_bytes [IOCDBLEN]`

6.70.1 Detailed Description

Definition at line 579 of file `cam_ccb.h`.

6.70.2 Field Documentation

6.70.2.1 `u_int8_t cdb_t::cdb_bytes[IOCDBLEN]`

Definition at line 582 of file `cam_ccb.h`.

Referenced by `camperiphdone()`, `cd6byteworkaround()`, `cdpause()`, `cdplay()`, `cdplaymsf()`, `cdplaytracks()`, `cdreadsubchannel()`, `cdreadtoc()`, `cdsetspeed()`, `chdone()`, `chgetparams()`, `cmd6workaround()`, `saerror()`, `scsi_command_string()`, `scsi_erase()`, `scsi_exchange_medium()`, `scsi_initialize_element_status()`, `scsi_inquiry()`, `scsi_load_unload()`, `scsi_log_select()`, `scsi_log_sense()`, `scsi_mode_select_len()`, `scsi_mode_sense_len()`, `scsi_move_medium()`, `scsi_position_to_element()`, `scsi_prevent()`, `scsi_read_block_limits()`, `scsi_read_capacity()`, `scsi_read_capacity_16()`, `scsi_read_dvd_structure()`, `scsi_read_element_status()`, `scsi_read_position()`, `scsi_read_write()`, `scsi_report_key()`, `scsi_report_luns()`, `scsi_request_sense()`, `scsi_reserve_release_unit()`, `scsi_rewind()`, `scsi_sa_read_write()`, `scsi_send_key()`, `scsi_send_receive()`, `scsi_send_volume_tag()`, `scsi_set_position()`, `scsi_space()`, `scsi_start_stop()`, `scsi_synchronize_cache()`, `scsi_test_unit_ready()`, `scsi_write_filemarks()`, `ses_runcmd()`, `targbhdone()`, and `xpt_action()`.

6.70.2.2 `u_int8_t* cdb_t::cdb_ptr`

Definition at line 580 of file `cam_ccb.h`.

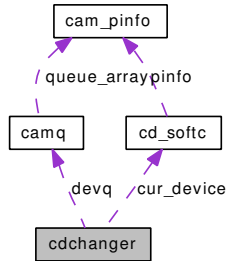
Referenced by `scsi_command_string()`.

The documentation for this union was generated from the following file:

- `/usr/src/sys/cam/cam_ccb.h`

6.71 cdchanger Struct Reference

Collaboration diagram for cdchanger:



Data Fields

- `path_id_t path_id`
- `target_id_t target_id`
- `int num_devices`
- `camq devq`
- `timeval start_time`
- `cd_softc * cur_device`
- `callout_handle short_handle`
- `callout_handle long_handle`
- `volatile cd_changer_flags flags`

6.71.1 Detailed Description

Definition at line 315 of file `scsi_cd.c`.

6.71.2 Field Documentation

6.71.2.1 struct `cd_softc* cdchanger::cur_device`

Definition at line 321 of file `scsi_cd.c`.

Referenced by `cdrunchangerqueue()`, and `cdshorttimeout()`.

6.71.2.2 struct `camq cdchanger::devq`

Definition at line 319 of file `scsi_cd.c`.

Referenced by `cdregister()`, and `cdrunchangerqueue()`.

6.71.2.3 volatile `cd_changer_flags cdchanger::flags`

Definition at line 324 of file `scsi_cd.c`.

Referenced by `cdchangerschedule()`, `cdrunchangerqueue()`, and `cdshorttimeout()`.

6.71.2.4 struct callout_handle cdchanger::long_handle

Definition at line 323 of file scsi_cd.c.

Referenced by cdchangerschedule(), and cdrunchangerqueue().

6.71.2.5 int cdchanger::num_devices

Definition at line 318 of file scsi_cd.c.

Referenced by cdregister().

6.71.2.6 path_id_t cdchanger::path_id

Definition at line 316 of file scsi_cd.c.

Referenced by cdregister().

6.71.2.7 struct callout_handle cdchanger::short_handle

Definition at line 322 of file scsi_cd.c.

Referenced by cdchangerschedule(), and cdrunchangerqueue().

6.71.2.8 struct timeval cdchanger::start_time

Definition at line 320 of file scsi_cd.c.

6.71.2.9 target_id_t cdchanger::target_id

Definition at line 317 of file scsi_cd.c.

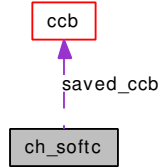
Referenced by cdregister().

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_cd.c](#)

6.72 ch_softc Struct Reference

Collaboration diagram for ch_softc:



Data Fields

- [ch_flags](#) flags
- [ch_state](#) state
- [ch_quirks](#) quirks
- [ccb saved_ccb](#)
- devstat * [device_stats](#)
- cdev * [dev](#)
- int [sc_picker](#)
- int [sc_firsts](#) [CHET_MAX+1]
- int [sc_counts](#) [CHET_MAX+1]
- u_int8_t [sc_movemask](#) [CHET_MAX+1]
- u_int8_t [sc_exchangemask](#) [CHET_MAX+1]
- int [sc_settle delay](#)

6.72.1 Detailed Description

Definition at line 142 of file scsi_ch.c.

6.72.2 Field Documentation

6.72.2.1 struct cdev* [ch_softc::dev](#)

Definition at line 148 of file scsi_ch.c.

Referenced by [chcleanup\(\)](#).

6.72.2.2 struct devstat* [ch_softc::device_stats](#)

Definition at line 147 of file scsi_ch.c.

Referenced by [chcleanup\(\)](#), [chexchange\(\)](#), [chgetelemstatus\(\)](#), [chgetparams\(\)](#), [chielem\(\)](#), [chmove\(\)](#), and [chposition\(\)](#).

6.72.2.3 [ch_flags](#) [ch_softc::flags](#)

Definition at line 143 of file scsi_ch.c.

Referenced by [chclose\(\)](#), and [chopen\(\)](#).

6.72.2.4 `ch_quirks` `ch_softc::quirks`

Definition at line 145 of file `scsi_ch.c`.

Referenced by `chdone()`, `chgetparams()`, and `chstart()`.

6.72.2.5 `union ccb` `ch_softc::saved_ccb`

Definition at line 146 of file `scsi_ch.c`.

Referenced by `cherror()`.

6.72.2.6 `int` `ch_softc::sc_counts`[`CHET_MAX+1`]

Definition at line 157 of file `scsi_ch.c`.

Referenced by `chdone()`, `chexchange()`, `chgetelemstatus()`, `chgetparams()`, `chioctl()`, `chmove()`, `chposition()`, `chsetvtag()`, and `copy_element_status()`.

6.72.2.7 `u_int8_t` `ch_softc::sc_exchangemask`[`CHET_MAX+1`]

Definition at line 168 of file `scsi_ch.c`.

Referenced by `chexchange()`, and `chgetparams()`.

6.72.2.8 `int` `ch_softc::sc_firsts`[`CHET_MAX+1`]

Definition at line 156 of file `scsi_ch.c`.

Referenced by `chdone()`, `chexchange()`, `chgetelemstatus()`, `chgetparams()`, `chioctl()`, `chmove()`, `chposition()`, `chsetvtag()`, and `copy_element_status()`.

6.72.2.9 `u_int8_t` `ch_softc::sc_movemask`[`CHET_MAX+1`]

Definition at line 163 of file `scsi_ch.c`.

Referenced by `chgetparams()`, and `chmove()`.

6.72.2.10 `int` `ch_softc::sc_picker`

Definition at line 150 of file `scsi_ch.c`.

Referenced by `chdone()`, `chexchange()`, `chioctl()`, `chmove()`, and `chposition()`.

6.72.2.11 `int` `ch_softc::sc_settledelay`

Definition at line 173 of file `scsi_ch.c`.

6.72.2.12 `ch_state` `ch_softc::state`

Definition at line 144 of file `scsi_ch.c`.

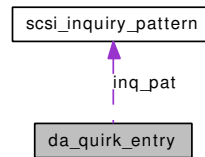
Referenced by `chdone()`, and `chstart()`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_ch.c](#)

6.73 da_quirk_entry Struct Reference

Collaboration diagram for da_quirk_entry:



Data Fields

- [scsi_inquiry_pattern inq_pat](#)
- [da_quirks quirks](#)

6.73.1 Detailed Description

Definition at line 138 of file `scsi_da.c`.

6.73.2 Field Documentation

6.73.2.1 struct [scsi_inquiry_pattern](#) `da_quirk_entry::inq_pat`

Definition at line 139 of file `scsi_da.c`.

6.73.2.2 [da_quirks](#) `da_quirk_entry::quirks`

Definition at line 140 of file `scsi_da.c`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_da.c`

6.74 da_softc Struct Reference

Data Fields

- bio_queue_head [bio_queue](#)

6.74.1 Detailed Description

Definition at line 120 of file scsi_da.c.

6.74.2 Field Documentation

6.74.2.1 struct bio_queue_head [da_softc::bio_queue](#)

Definition at line 121 of file scsi_da.c.

Referenced by [dadone\(\)](#), [dastart\(\)](#), and [dastrategy\(\)](#).

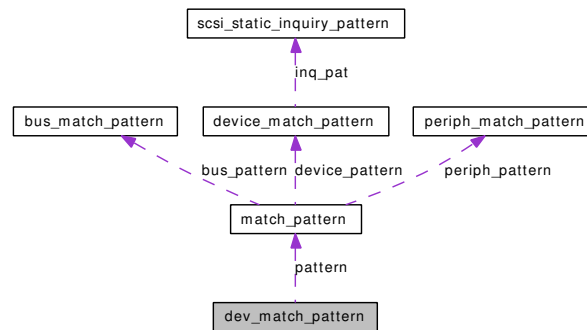
The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_da.c](#)

6.75 dev_match_pattern Struct Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for dev_match_pattern:



Data Fields

- [dev_match_type](#) type
- [match_pattern](#) pattern

6.75.1 Detailed Description

Definition at line 390 of file cam_ccb.h.

6.75.2 Field Documentation

6.75.2.1 union [match_pattern](#) dev_match_pattern::pattern

Definition at line 392 of file cam_ccb.h.

Referenced by [xptbusmatch\(\)](#), [xptdevicematch\(\)](#), and [xptperiphmatch\(\)](#).

6.75.2.2 [dev_match_type](#) dev_match_pattern::type

Definition at line 391 of file cam_ccb.h.

Referenced by [xpt_action\(\)](#).

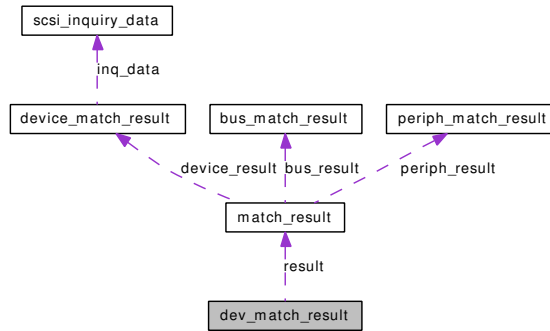
The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/cam_ccb.h](#)

6.76 dev_match_result Struct Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for dev_match_result:



Data Fields

- [dev_match_type](#) type
- [match_result](#) result

6.76.1 Detailed Description

Definition at line 429 of file cam_ccb.h.

6.76.2 Field Documentation

6.76.2.1 union [match_result](#) dev_match_result::result

Definition at line 431 of file cam_ccb.h.

Referenced by [xptedtbusfunc\(\)](#), [xptedtdevicefunc\(\)](#), [xptedtperiphfunc\(\)](#), and [xptplistperiphfunc\(\)](#).

6.76.2.2 [dev_match_type](#) dev_match_result::type

Definition at line 430 of file cam_ccb.h.

Referenced by [xptedtbusfunc\(\)](#), [xptedtdevicefunc\(\)](#), [xptedtperiphfunc\(\)](#), and [xptplistperiphfunc\(\)](#).

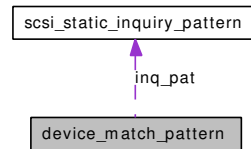
The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/cam_ccb.h](#)

6.77 device_match_pattern Struct Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for device_match_pattern:



Data Fields

- [path_id_t path_id](#)
- [target_id_t target_id](#)
- [lun_id_t target_lun](#)
- [scsi_static_inquiry_pattern inq_pat](#)
- [dev_pattern_flags flags](#)

6.77.1 Detailed Description

Definition at line 353 of file cam_ccb.h.

6.77.2 Field Documentation

6.77.2.1 [dev_pattern_flags device_match_pattern::flags](#)

Definition at line 358 of file cam_ccb.h.

6.77.2.2 [struct scsi_static_inquiry_pattern device_match_pattern::inq_pat](#)

Definition at line 357 of file cam_ccb.h.

6.77.2.3 [path_id_t device_match_pattern::path_id](#)

Definition at line 354 of file cam_ccb.h.

6.77.2.4 [target_id_t device_match_pattern::target_id](#)

Definition at line 355 of file cam_ccb.h.

6.77.2.5 [lun_id_t device_match_pattern::target_lun](#)

Definition at line 356 of file cam_ccb.h.

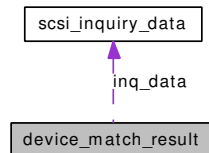
The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/cam_ccb.h](#)

6.78 device_match_result Struct Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for device_match_result:



Data Fields

- [path_id_t path_id](#)
- [target_id_t target_id](#)
- [lun_id_t target_lun](#)
- [scsi_inquiry_data inq_data](#)
- [dev_result_flags flags](#)

6.78.1 Detailed Description

Definition at line 408 of file cam_ccb.h.

6.78.2 Field Documentation

6.78.2.1 [dev_result_flags device_match_result::flags](#)

Definition at line 413 of file cam_ccb.h.

Referenced by [xptedtdevicefunc\(\)](#).

6.78.2.2 [struct scsi_inquiry_data device_match_result::inq_data](#)

Definition at line 412 of file cam_ccb.h.

Referenced by [xptedtdevicefunc\(\)](#).

6.78.2.3 [path_id_t device_match_result::path_id](#)

Definition at line 409 of file cam_ccb.h.

Referenced by [xptedtdevicefunc\(\)](#).

6.78.2.4 [target_id_t device_match_result::target_id](#)

Definition at line 410 of file cam_ccb.h.

Referenced by [xptedtdevicefunc\(\)](#).

6.78.2.5 [lun_id_t device_match_result::target_lun](#)

Definition at line 411 of file [cam_ccb.h](#).

Referenced by [xptedtdevicefunc\(\)](#).

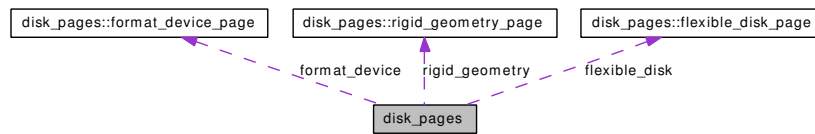
The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/cam_ccb.h](#)

6.79 disk_pages Union Reference

```
#include <scsi_da.h>
```

Collaboration diagram for disk_pages:



Data Fields

- [disk_pages::format_device_page](#) [format_device](#)
- [disk_pages::rigid_geometry_page](#) [rigid_geometry](#)
- [disk_pages::flexible_disk_page](#) [flexible_disk](#)

Data Structures

- struct [flexible_disk_page](#)
- struct [format_device_page](#)
- struct [rigid_geometry_page](#)

6.79.1 Detailed Description

Definition at line 324 of file [scsi_da.h](#).

6.79.2 Field Documentation

6.79.2.1 struct [disk_pages::flexible_disk_page](#) [disk_pages::flexible_disk](#)

6.79.2.2 struct [disk_pages::format_device_page](#) [disk_pages::format_device](#)

6.79.2.3 struct [disk_pages::rigid_geometry_page](#) [disk_pages::rigid_geometry](#)

The documentation for this union was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_da.h](#)

6.80 `disk_pages::flexible_disk_page` Struct Reference

```
#include <scsi_da.h>
```

Data Fields

- `u_int8_t pg_code`
- `u_int8_t pg_length`
- `u_int8_t xfr_rate_1`
- `u_int8_t xfr_rate_0`
- `u_int8_t nheads`
- `u_int8_t sec_per_track`
- `u_int8_t bytes_s_1`
- `u_int8_t bytes_s_0`
- `u_int8_t ncyl_1`
- `u_int8_t ncyl_0`
- `u_int8_t st_cyl_wp_1`
- `u_int8_t st_cyl_wp_0`
- `u_int8_t st_cyl_rwc_1`
- `u_int8_t st_cyl_rwc_0`
- `u_int8_t driv_step_1`
- `u_int8_t driv_step_0`
- `u_int8_t driv_step_pw`
- `u_int8_t head_stl_del_1`
- `u_int8_t head_stl_del_0`
- `u_int8_t motor_on_del`
- `u_int8_t motor_off_del`
- `u_int8_t trdy_ssn_mo`
- `u_int8_t spc`
- `u_int8_t write_comp`
- `u_int8_t head_load_del`
- `u_int8_t head_uoload_del`
- `u_int8_t pin32_pin2`
- `u_int8_t pin4_pint1`
- `u_int8_t medium_rot_rate_1`
- `u_int8_t medium_rot_rate_0`
- `u_int8_t reserved30`
- `u_int8_t reserved31`

6.80.1 Detailed Description

Definition at line 386 of file `scsi_da.h`.

6.80.2 Field Documentation

6.80.2.1 `u_int8_t disk_pages::flexible_disk_page::bytes_s_0`

Definition at line 396 of file `scsi_da.h`.

6.80.2.2 `u_int8_t disk_pages::flexible_disk_page::bytes_s_1`

Definition at line 395 of file `scsi_da.h`.

6.80.2.3 `u_int8_t disk_pages::flexible_disk_page::driv_step_0`

Definition at line 404 of file `scsi_da.h`.

6.80.2.4 `u_int8_t disk_pages::flexible_disk_page::driv_step_1`

Definition at line 403 of file `scsi_da.h`.

6.80.2.5 `u_int8_t disk_pages::flexible_disk_page::driv_step_pw`

Definition at line 405 of file `scsi_da.h`.

6.80.2.6 `u_int8_t disk_pages::flexible_disk_page::head_load_del`

Definition at line 413 of file `scsi_da.h`.

6.80.2.7 `u_int8_t disk_pages::flexible_disk_page::head_stl_del_0`

Definition at line 407 of file `scsi_da.h`.

6.80.2.8 `u_int8_t disk_pages::flexible_disk_page::head_stl_del_1`

Definition at line 406 of file `scsi_da.h`.

6.80.2.9 `u_int8_t disk_pages::flexible_disk_page::head_uoload_del`

Definition at line 414 of file `scsi_da.h`.

6.80.2.10 `u_int8_t disk_pages::flexible_disk_page::medium_rot_rate_0`

Definition at line 418 of file `scsi_da.h`.

6.80.2.11 `u_int8_t disk_pages::flexible_disk_page::medium_rot_rate_1`

Definition at line 417 of file `scsi_da.h`.

6.80.2.12 `u_int8_t disk_pages::flexible_disk_page::motor_off_del`

Definition at line 409 of file `scsi_da.h`.

6.80.2.13 `u_int8_t disk_pages::flexible_disk_page::motor_on_del`

Definition at line 408 of file scsi_da.h.

6.80.2.14 `u_int8_t disk_pages::flexible_disk_page::ncyl_0`

Definition at line 398 of file scsi_da.h.

6.80.2.15 `u_int8_t disk_pages::flexible_disk_page::ncyl_1`

Definition at line 397 of file scsi_da.h.

6.80.2.16 `u_int8_t disk_pages::flexible_disk_page::nheads`

Definition at line 393 of file scsi_da.h.

6.80.2.17 `u_int8_t disk_pages::flexible_disk_page::pg_code`

Definition at line 387 of file scsi_da.h.

6.80.2.18 `u_int8_t disk_pages::flexible_disk_page::pg_length`

Definition at line 389 of file scsi_da.h.

6.80.2.19 `u_int8_t disk_pages::flexible_disk_page::pin32_pin2`

Definition at line 415 of file scsi_da.h.

6.80.2.20 `u_int8_t disk_pages::flexible_disk_page::pin4_pint1`

Definition at line 416 of file scsi_da.h.

6.80.2.21 `u_int8_t disk_pages::flexible_disk_page::reserved30`

Definition at line 419 of file scsi_da.h.

6.80.2.22 `u_int8_t disk_pages::flexible_disk_page::reserved31`

Definition at line 420 of file scsi_da.h.

6.80.2.23 `u_int8_t disk_pages::flexible_disk_page::sec_per_track`

Definition at line 394 of file scsi_da.h.

6.80.2.24 `u_int8_t disk_pages::flexible_disk_page::spc`

Definition at line 411 of file `scsi_da.h`.

6.80.2.25 `u_int8_t disk_pages::flexible_disk_page::st_cyl_rwc_0`

Definition at line 402 of file `scsi_da.h`.

6.80.2.26 `u_int8_t disk_pages::flexible_disk_page::st_cyl_rwc_1`

Definition at line 401 of file `scsi_da.h`.

6.80.2.27 `u_int8_t disk_pages::flexible_disk_page::st_cyl_wp_0`

Definition at line 400 of file `scsi_da.h`.

6.80.2.28 `u_int8_t disk_pages::flexible_disk_page::st_cyl_wp_1`

Definition at line 399 of file `scsi_da.h`.

6.80.2.29 `u_int8_t disk_pages::flexible_disk_page::trdy_ssn_mo`

Definition at line 410 of file `scsi_da.h`.

6.80.2.30 `u_int8_t disk_pages::flexible_disk_page::write_comp`

Definition at line 412 of file `scsi_da.h`.

6.80.2.31 `u_int8_t disk_pages::flexible_disk_page::xfr_rate_0`

Definition at line 392 of file `scsi_da.h`.

6.80.2.32 `u_int8_t disk_pages::flexible_disk_page::xfr_rate_1`

Definition at line 391 of file `scsi_da.h`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_da.h](#)

6.81 `disk_pages::format_device_page` Struct Reference

```
#include <scsi_da.h>
```

Data Fields

- `u_int8_t pg_code`
- `u_int8_t pg_length`
- `u_int8_t trk_z_1`
- `u_int8_t trk_z_0`
- `u_int8_t alt_sec_1`
- `u_int8_t alt_sec_0`
- `u_int8_t alt_trk_z_1`
- `u_int8_t alt_trk_z_0`
- `u_int8_t alt_trk_v_1`
- `u_int8_t alt_trk_v_0`
- `u_int8_t ph_sec_t_1`
- `u_int8_t ph_sec_t_0`
- `u_int8_t bytes_s_1`
- `u_int8_t bytes_s_0`
- `u_int8_t interleave_1`
- `u_int8_t interleave_0`
- `u_int8_t trk_skew_1`
- `u_int8_t trk_skew_0`
- `u_int8_t cyl_skew_1`
- `u_int8_t cyl_skew_0`
- `u_int8_t flags`
- `u_int8_t reserved21`
- `u_int8_t reserved22`
- `u_int8_t reserved23`

6.81.1 Detailed Description

Definition at line 326 of file `scsi_da.h`.

6.81.2 Field Documentation

6.81.2.1 `u_int8_t disk_pages::format_device_page::alt_sec_0`

Definition at line 334 of file `scsi_da.h`.

6.81.2.2 `u_int8_t disk_pages::format_device_page::alt_sec_1`

Definition at line 333 of file `scsi_da.h`.

6.81.2.3 `u_int8_t disk_pages::format_device_page::alt_trk_v_0`

Definition at line 338 of file `scsi_da.h`.

6.81.2.4 `u_int8_t disk_pages::format_device_page::alt_trk_v_1`

Definition at line 337 of file `scsi_da.h`.

6.81.2.5 `u_int8_t disk_pages::format_device_page::alt_trk_z_0`

Definition at line 336 of file `scsi_da.h`.

6.81.2.6 `u_int8_t disk_pages::format_device_page::alt_trk_z_1`

Definition at line 335 of file `scsi_da.h`.

6.81.2.7 `u_int8_t disk_pages::format_device_page::bytes_s_0`

Definition at line 342 of file `scsi_da.h`.

6.81.2.8 `u_int8_t disk_pages::format_device_page::bytes_s_1`

Definition at line 341 of file `scsi_da.h`.

6.81.2.9 `u_int8_t disk_pages::format_device_page::cyl_skew_0`

Definition at line 348 of file `scsi_da.h`.

6.81.2.10 `u_int8_t disk_pages::format_device_page::cyl_skew_1`

Definition at line 347 of file `scsi_da.h`.

6.81.2.11 `u_int8_t disk_pages::format_device_page::flags`

Definition at line 349 of file `scsi_da.h`.

6.81.2.12 `u_int8_t disk_pages::format_device_page::interleave_0`

Definition at line 344 of file `scsi_da.h`.

6.81.2.13 `u_int8_t disk_pages::format_device_page::interleave_1`

Definition at line 343 of file `scsi_da.h`.

6.81.2.14 `u_int8_t disk_pages::format_device_page::pg_code`

Definition at line 327 of file `scsi_da.h`.

6.81.2.15 `u_int8_t disk_pages::format_device_page::pg_length`

Definition at line 329 of file `scsi_da.h`.

6.81.2.16 `u_int8_t disk_pages::format_device_page::ph_sec_t_0`

Definition at line 340 of file `scsi_da.h`.

6.81.2.17 `u_int8_t disk_pages::format_device_page::ph_sec_t_1`

Definition at line 339 of file `scsi_da.h`.

6.81.2.18 `u_int8_t disk_pages::format_device_page::reserved21`

Definition at line 354 of file `scsi_da.h`.

6.81.2.19 `u_int8_t disk_pages::format_device_page::reserved22`

Definition at line 355 of file `scsi_da.h`.

6.81.2.20 `u_int8_t disk_pages::format_device_page::reserved23`

Definition at line 356 of file `scsi_da.h`.

6.81.2.21 `u_int8_t disk_pages::format_device_page::trk_skew_0`

Definition at line 346 of file `scsi_da.h`.

6.81.2.22 `u_int8_t disk_pages::format_device_page::trk_skew_1`

Definition at line 345 of file `scsi_da.h`.

6.81.2.23 `u_int8_t disk_pages::format_device_page::trk_z_0`

Definition at line 332 of file `scsi_da.h`.

6.81.2.24 `u_int8_t disk_pages::format_device_page::trk_z_1`

Definition at line 331 of file `scsi_da.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_da.h`

6.82 disk_pages::rigid_geometry_page Struct Reference

```
#include <scsi_da.h>
```

Data Fields

- u_int8_t [pg_code](#)
- u_int8_t [pg_length](#)
- u_int8_t [ncyl_2](#)
- u_int8_t [ncyl_1](#)
- u_int8_t [ncyl_0](#)
- u_int8_t [nheads](#)
- u_int8_t [st_cyl_wp_2](#)
- u_int8_t [st_cyl_wp_1](#)
- u_int8_t [st_cyl_wp_0](#)
- u_int8_t [st_cyl_rwc_2](#)
- u_int8_t [st_cyl_rwc_1](#)
- u_int8_t [st_cyl_rwc_0](#)
- u_int8_t [driv_step_1](#)
- u_int8_t [driv_step_0](#)
- u_int8_t [land_zone_2](#)
- u_int8_t [land_zone_1](#)
- u_int8_t [land_zone_0](#)
- u_int8_t [rpl](#)
- u_int8_t [rot_offset](#)
- u_int8_t [reserved19](#)
- u_int8_t [medium_rot_rate_1](#)
- u_int8_t [medium_rot_rate_0](#)
- u_int8_t [reserved22](#)
- u_int8_t [reserved23](#)

6.82.1 Detailed Description

Definition at line 358 of file scsi_da.h.

6.82.2 Field Documentation

6.82.2.1 u_int8_t [disk_pages::rigid_geometry_page::driv_step_0](#)

Definition at line 374 of file scsi_da.h.

6.82.2.2 u_int8_t [disk_pages::rigid_geometry_page::driv_step_1](#)

Definition at line 373 of file scsi_da.h.

6.82.2.3 u_int8_t [disk_pages::rigid_geometry_page::land_zone_0](#)

Definition at line 377 of file scsi_da.h.

6.82.2.4 `u_int8_t disk_pages::rigid_geometry_page::land_zone_1`

Definition at line 376 of file scsi_da.h.

6.82.2.5 `u_int8_t disk_pages::rigid_geometry_page::land_zone_2`

Definition at line 375 of file scsi_da.h.

6.82.2.6 `u_int8_t disk_pages::rigid_geometry_page::medium_rot_rate_0`

Definition at line 382 of file scsi_da.h.

6.82.2.7 `u_int8_t disk_pages::rigid_geometry_page::medium_rot_rate_1`

Definition at line 381 of file scsi_da.h.

6.82.2.8 `u_int8_t disk_pages::rigid_geometry_page::ncyl_0`

Definition at line 365 of file scsi_da.h.

6.82.2.9 `u_int8_t disk_pages::rigid_geometry_page::ncyl_1`

Definition at line 364 of file scsi_da.h.

6.82.2.10 `u_int8_t disk_pages::rigid_geometry_page::ncyl_2`

Definition at line 363 of file scsi_da.h.

6.82.2.11 `u_int8_t disk_pages::rigid_geometry_page::nheads`

Definition at line 366 of file scsi_da.h.

6.82.2.12 `u_int8_t disk_pages::rigid_geometry_page::pg_code`

Definition at line 359 of file scsi_da.h.

6.82.2.13 `u_int8_t disk_pages::rigid_geometry_page::pg_length`

Definition at line 361 of file scsi_da.h.

6.82.2.14 `u_int8_t disk_pages::rigid_geometry_page::reserved19`

Definition at line 380 of file scsi_da.h.

6.82.2.15 `u_int8_t disk_pages::rigid_geometry_page::reserved22`

Definition at line 383 of file `scsi_da.h`.

6.82.2.16 `u_int8_t disk_pages::rigid_geometry_page::reserved23`

Definition at line 384 of file `scsi_da.h`.

6.82.2.17 `u_int8_t disk_pages::rigid_geometry_page::rot_offset`

Definition at line 379 of file `scsi_da.h`.

6.82.2.18 `u_int8_t disk_pages::rigid_geometry_page::rpl`

Definition at line 378 of file `scsi_da.h`.

6.82.2.19 `u_int8_t disk_pages::rigid_geometry_page::st_cyl_rwc_0`

Definition at line 372 of file `scsi_da.h`.

6.82.2.20 `u_int8_t disk_pages::rigid_geometry_page::st_cyl_rwc_1`

Definition at line 371 of file `scsi_da.h`.

6.82.2.21 `u_int8_t disk_pages::rigid_geometry_page::st_cyl_rwc_2`

Definition at line 370 of file `scsi_da.h`.

6.82.2.22 `u_int8_t disk_pages::rigid_geometry_page::st_cyl_wp_0`

Definition at line 369 of file `scsi_da.h`.

6.82.2.23 `u_int8_t disk_pages::rigid_geometry_page::st_cyl_wp_1`

Definition at line 368 of file `scsi_da.h`.

6.82.2.24 `u_int8_t disk_pages::rigid_geometry_page::st_cyl_wp_2`

Definition at line 367 of file `scsi_da.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_da.h`

6.83 disk_params Struct Reference

Data Fields

- [u_int8_t heads](#)
- [u_int32_t cylinders](#)
- [u_int8_t secs_per_track](#)
- [u_int32_t secksize](#)
- [u_int64_t sectors](#)

6.83.1 Detailed Description

Definition at line 112 of file `scsi_da.c`.

6.83.2 Field Documentation

6.83.2.1 [u_int32_t disk_params::cylinders](#)

Definition at line 114 of file `scsi_da.c`.

Referenced by `dadone()`, and `dasetgeom()`.

6.83.2.2 [u_int8_t disk_params::heads](#)

Definition at line 113 of file `scsi_da.c`.

Referenced by `dadone()`, and `dasetgeom()`.

6.83.2.3 [u_int8_t disk_params::secs_per_track](#)

Definition at line 115 of file `scsi_da.c`.

Referenced by `dadone()`, and `dasetgeom()`.

6.83.2.4 [u_int32_t disk_params::secksize](#)

Definition at line 116 of file `scsi_da.c`.

Referenced by `dadone()`, and `dasetgeom()`.

6.83.2.5 [u_int64_t disk_params::sectors](#)

Definition at line 117 of file `scsi_da.c`.

Referenced by `dadone()`, and `dasetgeom()`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_da.c](#)

6.84 encobj Struct Reference

Data Fields

- `uint32_t enctype`: 8
- `uint32_t subenclosure`: 8
- `uint32_t svalid`: 1
- `uint32_t priv`: 15
- `uint8_t encstat` [4]

6.84.1 Detailed Description

Definition at line 81 of file `scsi_ses.c`.

6.84.2 Field Documentation

6.84.2.1 `uint8_t encobj::encstat`[4]

Definition at line 87 of file `scsi_ses.c`.

Referenced by `saft_e_get_objstat()`, `saft_e_rdstat()`, and `ses_get_objstat()`.

6.84.2.2 `uint32_t encobj::enctype`

Definition at line 83 of file `scsi_ses.c`.

Referenced by `saft_e_set_objstat()`, `saft_e_softc_init()`, `ses_getconfig()`, `sesioctl()`, and `set_objstat_sel()`.

6.84.2.3 `uint32_t encobj::priv`

Definition at line 83 of file `scsi_ses.c`.

Referenced by `saft_e_set_objstat()`, `set_objstat_sel()`, and `wrslot_stat()`.

6.84.2.4 `uint32_t encobj::subenclosure`

Definition at line 83 of file `scsi_ses.c`.

Referenced by `ses_getconfig()`, and `sesioctl()`.

6.84.2.5 `uint32_t encobj::svalid`

Definition at line 83 of file `scsi_ses.c`.

Referenced by `saft_e_get_objstat()`, `saft_e_rdstat()`, `saft_e_set_objstat()`, `ses_get_objstat()`, `sesioctl()`, and `set_objstat_sel()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_ses.c`

6.85 encvec Struct Reference

Data Fields

- `int(* softc_init)(ses_softc_t *, int)`
- `int(* init_enc)(ses_softc_t *)`
- `int(* get_encstat)(ses_softc_t *, int)`
- `int(* set_encstat)(ses_softc_t *, ses_encstat, int)`
- `int(* get_objstat)(ses_softc_t *, ses_objstat *, int)`
- `int(* set_objstat)(ses_softc_t *, ses_objstat *, int)`

6.85.1 Detailed Description

Definition at line 70 of file `scsi_ses.c`.

6.85.2 Field Documentation

6.85.2.1 `int(* encvec::get_encstat)(ses_softc_t *, int)`

Referenced by `sesioctl()`, and `sesregister()`.

6.85.2.2 `int(* encvec::get_objstat)(ses_softc_t *, ses_objstat *, int)`

Referenced by `sesioctl()`, and `sesregister()`.

6.85.2.3 `int(* encvec::init_enc)(ses_softc_t *)`

Referenced by `sesioctl()`, and `sesregister()`.

6.85.2.4 `int(* encvec::set_encstat)(ses_softc_t *, ses_encstat, int)`

Referenced by `sesioctl()`, and `sesregister()`.

6.85.2.5 `int(* encvec::set_objstat)(ses_softc_t *, ses_objstat *, int)`

Referenced by `sesioctl()`, and `sesregister()`.

6.85.2.6 `int(* encvec::softc_init)(ses_softc_t *, int)`

Referenced by `sesopen()`, and `sesregister()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_ses.c`

6.86 format_capacity_descriptor Struct Reference

```
#include <scsi_da.h>
```

Data Fields

- uint8_t [nblocks](#) [4]
- uint8_t [byte4](#)
- uint8_t [block_length](#) [3]

6.86.1 Detailed Description

Definition at line 251 of file `scsi_da.h`.

6.86.2 Field Documentation

6.86.2.1 uint8_t [format_capacity_descriptor::block_length](#)[3]

Definition at line 261 of file `scsi_da.h`.

6.86.2.2 uint8_t [format_capacity_descriptor::byte4](#)

Definition at line 253 of file `scsi_da.h`.

6.86.2.3 uint8_t [format_capacity_descriptor::nblocks](#)[4]

Definition at line 252 of file `scsi_da.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_da.h`

6.87 format_capacity_list_header Struct Reference

```
#include <scsi_da.h>
```

Data Fields

- uint8_t [unused](#) [3]
- uint8_t [capacity_list_length](#)

6.87.1 Detailed Description

Definition at line 246 of file `scsi_da.h`.

6.87.2 Field Documentation

6.87.2.1 uint8_t [format_capacity_list_header::capacity_list_length](#)

Definition at line 248 of file `scsi_da.h`.

6.87.2.2 uint8_t [format_capacity_list_header::unused](#)[3]

Definition at line 247 of file `scsi_da.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_da.h`

6.88 `format_defect_list_header` Struct Reference

```
#include <scsi_da.h>
```

Data Fields

- `u_int8_t reserved`
- `u_int8_t byte2`
- `u_int8_t defect_list_length` [2]

6.88.1 Detailed Description

Definition at line 160 of file `scsi_da.h`.

6.88.2 Field Documentation

6.88.2.1 `u_int8_t format_defect_list_header::byte2`

Definition at line 163 of file `scsi_da.h`.

6.88.2.2 `u_int8_t format_defect_list_header::defect_list_length`[2]

Definition at line 172 of file `scsi_da.h`.

6.88.2.3 `u_int8_t format_defect_list_header::reserved`

Definition at line 162 of file `scsi_da.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_da.h`

6.89 format_ipat_descriptor Struct Reference

```
#include <scsi_da.h>
```

Data Fields

- [u_int8_t byte1](#)
- [u_int8_t pattern_type](#)
- [u_int8_t pat_length](#) [2]

6.89.1 Detailed Description

Definition at line 175 of file `scsi_da.h`.

6.89.2 Field Documentation

6.89.2.1 [u_int8_t format_ipat_descriptor::byte1](#)

Definition at line 177 of file `scsi_da.h`.

6.89.2.2 [u_int8_t format_ipat_descriptor::pat_length](#)[2]

Definition at line 185 of file `scsi_da.h`.

6.89.2.3 [u_int8_t format_ipat_descriptor::pattern_type](#)

Definition at line 182 of file `scsi_da.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_da.h`

6.90 `ioc_enable_lun` Struct Reference

```
#include <scsi_targetio.h>
```

Data Fields

- [path_id_t path_id](#)
- [target_id_t target_id](#)
- [lun_id_t lun_id](#)
- [int grp6_len](#)
- [int grp7_len](#)

6.90.1 Detailed Description

Definition at line 60 of file `scsi_targetio.h`.

6.90.2 Field Documentation

6.90.2.1 [int ioc_enable_lun::grp6_len](#)

Definition at line 64 of file `scsi_targetio.h`.

Referenced by `targioctl()`.

6.90.2.2 [int ioc_enable_lun::grp7_len](#)

Definition at line 65 of file `scsi_targetio.h`.

Referenced by `targioctl()`.

6.90.2.3 [lun_id_t ioc_enable_lun::lun_id](#)

Definition at line 63 of file `scsi_targetio.h`.

Referenced by `targioctl()`.

6.90.2.4 [path_id_t ioc_enable_lun::path_id](#)

Definition at line 61 of file `scsi_targetio.h`.

Referenced by `targioctl()`.

6.90.2.5 [target_id_t ioc_enable_lun::target_id](#)

Definition at line 62 of file `scsi_targetio.h`.

Referenced by `targioctl()`.

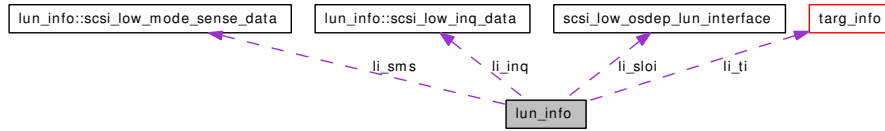
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_targetio.h`

6.91 lun_info Struct Reference

```
#include <scsi_low.h>
```

Collaboration diagram for lun_info:



Public Member Functions

- `LIST_ENTRY (lun_info) lun_chain`

Data Fields

- `scsi_low_osdep_lun_interface li_sloi`
- `int li_lun`
- `targ_info * li_ti`
- `slccbtab li_discq`
- `int li_maxnexus`
- `int li_maxnqio`
- `int li_nqio`
- `int li_disc`
- `u_int li_qtagbits`
- `u_int8_t li_qtagarray [SCSI_LOW_MAXNEXUS]`
- `u_int li_qd`
- `u_int li_qflags`
- `u_int li_state`
- `u_int li_flags_valid`
- `u_int li_flags`
- `u_int li_cfgflags`
- `u_int li_diskflags`
- `u_int li_quirks`
- `lun_info::scsi_low_inq_data li_inq`
- `lun_info::scsi_low_mode_sense_data li_sms`

Data Structures

- struct `scsi_low_inq_data`
- struct `scsi_low_mode_sense_data`

6.91.1 Detailed Description

Definition at line 301 of file `scsi_low.h`.

6.91.2 Member Function Documentation

6.91.2.1 lun_info::LIST_ENTRY (lun_info)

6.91.3 Field Documentation

6.91.3.1 u_int lun_info::li_cfgflags

Definition at line 352 of file scsi_low.h.

Referenced by scsi_low_alloc_li().

6.91.3.2 int lun_info::li_disc

Definition at line 317 of file scsi_low.h.

Referenced by scsi_low_disconnected(), scsi_low_establish_ccb(), scsi_low_revoke_ccb(), scsi_low_start(), and scsi_low_test_abort().

6.91.3.3 struct slcbtab lun_info::li_discq

Definition at line 309 of file scsi_low.h.

Referenced by scsi_low_alloc_li(), scsi_low_disconnected(), scsi_low_establish_ccb(), scsi_low_find_ccb(), scsi_low_print(), scsi_low_reset_nexus_lun(), scsi_low_revoke_ccb(), scsi_low_test_abort(), and scsi_low_timeout_check().

6.91.3.4 u_int lun_info::li_diskflags

Definition at line 353 of file scsi_low.h.

Referenced by scsi_low_alloc_li(), scsi_low_reset_nexus_target(), and scsi_low_setup_done().

6.91.3.5 u_int lun_info::li_flags

Definition at line 351 of file scsi_low.h.

Referenced by scsi_low_msginfunc_lcc(), scsi_low_reset_nexus_target(), and scsi_low_start().

6.91.3.6 u_int lun_info::li_flags_valid

Definition at line 343 of file scsi_low.h.

Referenced by scsi_low_alloc_li(), scsi_low_reset_nexus_target(), and scsi_low_setup_done().

6.91.3.7 struct lun_info::scsi_low_inq_data lun_info::li_inq

Referenced by scsi_low_setup_done(), and scsi_low_setup_start().

6.91.3.8 int lun_info::li_lun

Definition at line 304 of file scsi_low.h.

Referenced by `scsi_low_alloc_li()`, `scsi_low_msgfunc_identify()`, `scsi_low_msgin()`, and `scsi_low_setup_done()`.

6.91.3.9 `int lun_info::li_maxnexus`

Definition at line 314 of file `scsi_low.h`.

Referenced by `scsi_low_msginfunc_cc()`, and `scsi_low_msginfunc_lcc()`.

6.91.3.10 `int lun_info::li_maxnqio`

Definition at line 315 of file `scsi_low.h`.

Referenced by `scsi_low_msginfunc_cc()`, `scsi_low_msginfunc_lcc()`, `scsi_low_reset_nexus_target()`, and `scsi_low_start()`.

6.91.3.11 `int lun_info::li_nqio`

Definition at line 316 of file `scsi_low.h`.

Referenced by `scsi_low_activate_qtag()`, `scsi_low_deactivate_qtag()`, `scsi_low_msgin()`, `scsi_low_msginfunc_cc()`, and `scsi_low_start()`.

6.91.3.12 `u_int lun_info::li_qd`

Definition at line 324 of file `scsi_low.h`.

Referenced by `scsi_low_alloc_qtag()`.

6.91.3.13 `u_int lun_info::li_qflags`

Definition at line 328 of file `scsi_low.h`.

Referenced by `scsi_low_msginfunc_cc()`, and `scsi_low_setup_done()`.

6.91.3.14 `u_int8_t lun_info::li_qtagarray[SCSI_LOW_MAXNEXUS]`

Definition at line 323 of file `scsi_low.h`.

Referenced by `scsi_low_alloc_qtag()`, and `scsi_low_dealloc_qtag()`.

6.91.3.15 `u_int lun_info::li_qtagbits`

Definition at line 320 of file `scsi_low.h`.

Referenced by `scsi_low_alloc_li()`, `scsi_low_alloc_qtag()`, and `scsi_low_dealloc_qtag()`.

6.91.3.16 `u_int lun_info::li_quirks`

Definition at line 354 of file `scsi_low.h`.

Referenced by `scsi_low_alloc_li()`.

6.91.3.17 struct [scsi_low_osdep_lun_interface](#) [lun_info::li_sloi](#)

Definition at line 302 of file [scsi_low.h](#).

6.91.3.18 struct [lun_info::scsi_low_mode_sense_data](#) [lun_info::li_sms](#)

Referenced by [scsi_low_setup_done\(\)](#), and [scsi_low_setup_start\(\)](#).

6.91.3.19 u_int [lun_info::li_state](#)

Definition at line 338 of file [scsi_low.h](#).

Referenced by [scsi_low_reset_nexus_target\(\)](#), [scsi_low_setup_done\(\)](#), [scsi_low_setup_start\(\)](#), and [scsi_low_start\(\)](#).

6.91.3.20 struct [targ_info*](#) [lun_info::li_ti](#)

Definition at line 305 of file [scsi_low.h](#).

Referenced by [scsi_low_alloc_li\(\)](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_low.h](#)

6.92 lun_info::scsi_low_inq_data Struct Reference

```
#include <scsi_low.h>
```

Data Fields

- [u_int8_t sd_type](#)
- [u_int8_t sd_sp1](#)
- [u_int8_t sd_version](#)
- [u_int8_t sd_resp](#)
- [u_int8_t sd_len](#)
- [u_int8_t sd_sp2](#) [2]
- [u_int8_t sd_support](#)

6.92.1 Detailed Description

Definition at line 357 of file scsi_low.h.

6.92.2 Field Documentation

6.92.2.1 [u_int8_t lun_info::scsi_low_inq_data::sd_len](#)

Definition at line 362 of file scsi_low.h.

Referenced by `scsi_low_setup_done()`.

6.92.2.2 [u_int8_t lun_info::scsi_low_inq_data::sd_resp](#)

Definition at line 361 of file scsi_low.h.

6.92.2.3 [u_int8_t lun_info::scsi_low_inq_data::sd_sp1](#)

Definition at line 359 of file scsi_low.h.

6.92.2.4 [u_int8_t lun_info::scsi_low_inq_data::sd_sp2](#)[2]

Definition at line 363 of file scsi_low.h.

6.92.2.5 [u_int8_t lun_info::scsi_low_inq_data::sd_support](#)

Definition at line 364 of file scsi_low.h.

Referenced by `scsi_low_setup_done()`.

6.92.2.6 [u_int8_t lun_info::scsi_low_inq_data::sd_type](#)

Definition at line 358 of file scsi_low.h.

6.92.2.7 u_int8_t lun_info::scsi_low_inq_data::sd_version

Definition at line 360 of file scsi_low.h.

Referenced by scsi_low_setup_done().

The documentation for this struct was generated from the following file:

- /usr/src/sys/cam/scsi/scsi_low.h

6.93 lun_info::scsi_low_mode_sense_data Struct Reference

```
#include <scsi_low.h>
```

Data Fields

- `u_int8_t sms_header` [4]
- struct {
 - `u_int8_t cmp_page`
 - `u_int8_t cmp_length`
 - `u_int8_t cmp_rlec`
 - `u_int8_t cmp_qc`
 - `u_int8_t cmp_eca`
 - `u_int8_t cmp_spare` [3]
- `sms_cmp`

6.93.1 Detailed Description

Definition at line 368 of file `scsi_low.h`.

6.93.2 Field Documentation

6.93.2.1 `u_int8_t lun_info::scsi_low_mode_sense_data::cmp_eca`

Definition at line 375 of file `scsi_low.h`.

6.93.2.2 `u_int8_t lun_info::scsi_low_mode_sense_data::cmp_length`

Definition at line 372 of file `scsi_low.h`.

6.93.2.3 `u_int8_t lun_info::scsi_low_mode_sense_data::cmp_page`

Definition at line 371 of file `scsi_low.h`.

Referenced by `scsi_low_setup_done()`.

6.93.2.4 `u_int8_t lun_info::scsi_low_mode_sense_data::cmp_qc`

Definition at line 374 of file `scsi_low.h`.

Referenced by `scsi_low_setup_done()`.

6.93.2.5 `u_int8_t lun_info::scsi_low_mode_sense_data::cmp_rlec`

Definition at line 373 of file `scsi_low.h`.

6.93.2.6 `u_int8_t lun_info::scsi_low_mode_sense_data::cmp_spare[3]`

Definition at line 376 of file `scsi_low.h`.

6.93.2.7 `struct { ... } lun_info::scsi_low_mode_sense_data::sms_cmp`

Referenced by `scsi_low_setup_done()`.

6.93.2.8 `u_int8_t lun_info::scsi_low_mode_sense_data::sms_header[4]`

Definition at line 369 of file `scsi_low.h`.

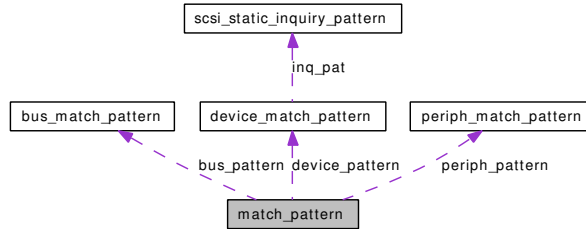
The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_low.h](#)

6.94 match_pattern Union Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for match_pattern:



Data Fields

- [periph_match_pattern](#) [periph_pattern](#)
- [device_match_pattern](#) [device_pattern](#)
- [bus_match_pattern](#) [bus_pattern](#)

6.94.1 Detailed Description

Definition at line 378 of file `cam_ccb.h`.

6.94.2 Field Documentation

6.94.2.1 struct [bus_match_pattern](#) `match_pattern::bus_pattern`

Definition at line 381 of file `cam_ccb.h`.

Referenced by `xptbusmatch()`.

6.94.2.2 struct [device_match_pattern](#) `match_pattern::device_pattern`

Definition at line 380 of file `cam_ccb.h`.

Referenced by `xptdevicematch()`.

6.94.2.3 struct [periph_match_pattern](#) `match_pattern::periph_pattern`

Definition at line 379 of file `cam_ccb.h`.

Referenced by `xptperiphmatch()`.

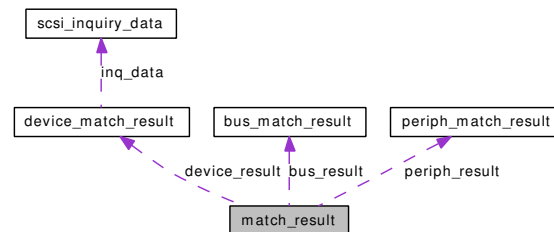
The documentation for this union was generated from the following file:

- [/usr/src/sys/cam/cam_ccb.h](#)

6.95 match_result Union Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for match_result:



Data Fields

- [periph_match_result](#) [periph_result](#)
- [device_match_result](#) [device_result](#)
- [bus_match_result](#) [bus_result](#)

6.95.1 Detailed Description

Definition at line 423 of file `cam_ccb.h`.

6.95.2 Field Documentation

6.95.2.1 struct [bus_match_result](#) `match_result::bus_result`

Definition at line 426 of file `cam_ccb.h`.

Referenced by `xptedtbusfunc()`.

6.95.2.2 struct [device_match_result](#) `match_result::device_result`

Definition at line 425 of file `cam_ccb.h`.

Referenced by `xptedtdevicefunc()`.

6.95.2.3 struct [periph_match_result](#) `match_result::periph_result`

Definition at line 424 of file `cam_ccb.h`.

Referenced by `xptdtperiphfunc()`, and `xptplistperiphfunc()`.

The documentation for this union was generated from the following file:

- [/usr/src/sys/cam/cam_ccb.h](#)

6.96 `op_table_entry` Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- `u_int8_t opcode`
- `u_int16_t opmask`
- `const char * desc`

6.96.1 Detailed Description

Definition at line 885 of file `scsi_all.h`.

6.96.2 Field Documentation

6.96.2.1 `const char* op_table_entry::desc`

Definition at line 888 of file `scsi_all.h`.

6.96.2.2 `u_int8_t op_table_entry::opcode`

Definition at line 886 of file `scsi_all.h`.

Referenced by `scsi_op_desc()`.

6.96.2.3 `u_int16_t op_table_entry::opmask`

Definition at line 887 of file `scsi_all.h`.

Referenced by `scsi_op_desc()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.97 page_device_capabilities Struct Reference

```
#include <scsi_ch.h>
```

Data Fields

- [u_int8_t pg_code](#)
- [u_int8_t pg_length](#)
- [u_int8_t stor](#)
- [u_int8_t reserved0](#)
- [u_int8_t move_from](#) [CHET_MAX+1]
- [u_int8_t reserved1](#) [4]
- [u_int8_t exchange_with](#) [CHET_MAX+1]

6.97.1 Detailed Description

Definition at line 318 of file scsi_ch.h.

6.97.2 Field Documentation

6.97.2.1 [u_int8_t page_device_capabilities::exchange_with](#)[CHET_MAX+1]

Definition at line 353 of file scsi_ch.h.

Referenced by `chgetparams()`.

6.97.2.2 [u_int8_t page_device_capabilities::move_from](#)[CHET_MAX+1]

Definition at line 342 of file scsi_ch.h.

Referenced by `chgetparams()`.

6.97.2.3 [u_int8_t page_device_capabilities::pg_code](#)

Definition at line 319 of file scsi_ch.h.

6.97.2.4 [u_int8_t page_device_capabilities::pg_length](#)

Definition at line 320 of file scsi_ch.h.

6.97.2.5 [u_int8_t page_device_capabilities::reserved0](#)

Definition at line 333 of file scsi_ch.h.

6.97.2.6 [u_int8_t page_device_capabilities::reserved1](#)[4]

Definition at line 348 of file scsi_ch.h.

6.97.2.7 `u_int8_t page_device_capabilities::stor`

Definition at line 327 of file `scsi_ch.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_ch.h`

6.98 page_element_address_assignment Struct Reference

```
#include <scsi_ch.h>
```

Data Fields

- [u_int8_t pg_code](#)
- [u_int8_t pg_length](#)
- [u_int8_t mtea](#) [2]
- [u_int8_t nmte](#) [2]
- [u_int8_t fsea](#) [2]
- [u_int8_t nse](#) [2]
- [u_int8_t fieea](#) [2]
- [u_int8_t niee](#) [2]
- [u_int8_t fdtea](#) [2]
- [u_int8_t ndte](#) [2]
- [u_int8_t reserved](#) [2]

6.98.1 Detailed Description

Definition at line 388 of file scsi_ch.h.

6.98.2 Field Documentation

6.98.2.1 [u_int8_t page_element_address_assignment::fdtea](#)[2]

Definition at line 411 of file scsi_ch.h.

Referenced by `chdone()`, and `chgetparams()`.

6.98.2.2 [u_int8_t page_element_address_assignment::fieea](#)[2]

Definition at line 405 of file scsi_ch.h.

Referenced by `chdone()`, and `chgetparams()`.

6.98.2.3 [u_int8_t page_element_address_assignment::fsea](#)[2]

Definition at line 399 of file scsi_ch.h.

Referenced by `chdone()`, and `chgetparams()`.

6.98.2.4 [u_int8_t page_element_address_assignment::mtea](#)[2]

Definition at line 393 of file scsi_ch.h.

Referenced by `chdone()`, and `chgetparams()`.

6.98.2.5 `u_int8_t page_element_address_assignment::ndte[2]`

Definition at line 414 of file `scsi_ch.h`.

Referenced by `chdone()`, and `chgetparams()`.

6.98.2.6 `u_int8_t page_element_address_assignment::niee[2]`

Definition at line 408 of file `scsi_ch.h`.

Referenced by `chdone()`, and `chgetparams()`.

6.98.2.7 `u_int8_t page_element_address_assignment::nmte[2]`

Definition at line 396 of file `scsi_ch.h`.

Referenced by `chdone()`, and `chgetparams()`.

6.98.2.8 `u_int8_t page_element_address_assignment::nse[2]`

Definition at line 402 of file `scsi_ch.h`.

Referenced by `chdone()`, and `chgetparams()`.

6.98.2.9 `u_int8_t page_element_address_assignment::pg_code`

Definition at line 389 of file `scsi_ch.h`.

6.98.2.10 `u_int8_t page_element_address_assignment::pg_length`

Definition at line 390 of file `scsi_ch.h`.

6.98.2.11 `u_int8_t page_element_address_assignment::reserved[2]`

Definition at line 416 of file `scsi_ch.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_ch.h`

6.99 page_transport_geometry_parameters Struct Reference

```
#include <scsi_ch.h>
```

Data Fields

- [u_int8_t pg_code](#)
- [u_int8_t pg_length](#)
- [u_int8_t misc](#)
- [u_int8_t member](#)

6.99.1 Detailed Description

Definition at line 429 of file scsi_ch.h.

6.99.2 Field Documentation

6.99.2.1 [u_int8_t page_transport_geometry_parameters::member](#)

Definition at line 439 of file scsi_ch.h.

6.99.2.2 [u_int8_t page_transport_geometry_parameters::misc](#)

Definition at line 435 of file scsi_ch.h.

6.99.2.3 [u_int8_t page_transport_geometry_parameters::pg_code](#)

Definition at line 430 of file scsi_ch.h.

6.99.2.4 [u_int8_t page_transport_geometry_parameters::pg_length](#)

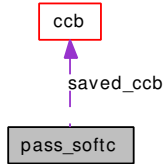
Definition at line 431 of file scsi_ch.h.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_ch.h](#)

6.100 pass_softc Struct Reference

Collaboration diagram for pass_softc:



Data Fields

- [pass_state](#) state
- [pass_flags](#) flags
- [u_int8_t pd_type](#)
- [ccb](#) saved_ccb
- devstat * [device_stats](#)
- cdev * [dev](#)

6.100.1 Detailed Description

Definition at line 71 of file scsi_pass.c.

6.100.2 Field Documentation

6.100.2.1 struct cdev* [pass_softc::dev](#)

Definition at line 77 of file scsi_pass.c.

Referenced by `passcleanup()`.

6.100.2.2 struct devstat* [pass_softc::device_stats](#)

Definition at line 76 of file scsi_pass.c.

Referenced by `passcleanup()`, and `passsendccb()`.

6.100.2.3 [pass_flags](#) [pass_softc::flags](#)

Definition at line 73 of file scsi_pass.c.

Referenced by `passclose()`, and `passopen()`.

6.100.2.4 [u_int8_t](#) [pass_softc::pd_type](#)

Definition at line 74 of file scsi_pass.c.

6.100.2.5 union ccb pass_softc::saved_ccb

Definition at line 75 of file scsi_pass.c.

Referenced by passerror().

6.100.2.6 pass_state pass_softc::state

Definition at line 72 of file scsi_pass.c.

Referenced by passstart().

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_pass.c](#)

6.101 `periph_driver` Struct Reference

```
#include <cam_periph.h>
```

Public Member Functions

- [TAILQ_HEAD](#) (`, cam_periph`) units

Data Fields

- [periph_init_func_t](#) `init`
- `char *` [driver_name](#)
- `u_int` [generation](#)

6.101.1 Detailed Description

Definition at line 76 of file `cam_periph.h`.

6.101.2 Member Function Documentation

6.101.2.1 `periph_driver::TAILQ_HEAD` (`cam_periph`)

6.101.3 Field Documentation

6.101.3.1 `char*` `periph_driver::driver_name`

Definition at line 78 of file `cam_periph.h`.

Referenced by `camperiphnextunit()`, and `camperiphunit()`.

6.101.3.2 `u_int` `periph_driver::generation`

Definition at line 80 of file `cam_periph.h`.

6.101.3.3 `periph_init_func_t` `periph_driver::init`

Definition at line 77 of file `cam_periph.h`.

Referenced by `xpt_finishconfig()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/cam_periph.h`

6.102 `periph_match_pattern` Struct Reference

```
#include <cam_ccb.h>
```

Data Fields

- `char periph_name` [DEV_IDLEN]
- `u_int32_t unit_number`
- `path_id_t path_id`
- `target_id_t target_id`
- `lun_id_t target_lun`
- `periph_pattern_flags flags`

6.102.1 Detailed Description

Definition at line 335 of file `cam_ccb.h`.

6.102.2 Field Documentation

6.102.2.1 `periph_pattern_flags periph_match_pattern::flags`

Definition at line 341 of file `cam_ccb.h`.

6.102.2.2 `path_id_t periph_match_pattern::path_id`

Definition at line 338 of file `cam_ccb.h`.

6.102.2.3 `char periph_match_pattern::periph_name`[DEV_IDLEN]

Definition at line 336 of file `cam_ccb.h`.

6.102.2.4 `target_id_t periph_match_pattern::target_id`

Definition at line 339 of file `cam_ccb.h`.

6.102.2.5 `lun_id_t periph_match_pattern::target_lun`

Definition at line 340 of file `cam_ccb.h`.

6.102.2.6 `u_int32_t periph_match_pattern::unit_number`

Definition at line 337 of file `cam_ccb.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/cam_ccb.h`

6.103 `periph_match_result` Struct Reference

```
#include <cam_ccb.h>
```

Data Fields

- char `periph_name` [DEV_IDLEN]
- u_int32_t `unit_number`
- `path_id_t` `path_id`
- `target_id_t` `target_id`
- `lun_id_t` `target_lun`

6.103.1 Detailed Description

Definition at line 395 of file `cam_ccb.h`.

6.103.2 Field Documentation

6.103.2.1 `path_id_t periph_match_result::path_id`

Definition at line 398 of file `cam_ccb.h`.

Referenced by `xptedtperiphfunc()`, and `xptplistperiphfunc()`.

6.103.2.2 char `periph_match_result::periph_name`[DEV_IDLEN]

Definition at line 396 of file `cam_ccb.h`.

Referenced by `xptedtperiphfunc()`, and `xptplistperiphfunc()`.

6.103.2.3 `target_id_t periph_match_result::target_id`

Definition at line 399 of file `cam_ccb.h`.

Referenced by `xptedtperiphfunc()`, and `xptplistperiphfunc()`.

6.103.2.4 `lun_id_t periph_match_result::target_lun`

Definition at line 400 of file `cam_ccb.h`.

Referenced by `xptedtperiphfunc()`, and `xptplistperiphfunc()`.

6.103.2.5 u_int32_t `periph_match_result::unit_number`

Definition at line 397 of file `cam_ccb.h`.

Referenced by `xptedtperiphfunc()`, and `xptplistperiphfunc()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/cam_ccb.h`

6.104 probe_softc Struct Reference

6.104.1 Detailed Description

Definition at line 5542 of file cam_xpt.c.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/cam_xpt.c](#)

6.105 pt_softc Struct Reference

Data Fields

- bio_queue_head [bio_queue](#)
- devstat * [device_stats](#)

6.105.1 Detailed Description

Definition at line 78 of file `scsi_pt.c`.

6.105.2 Field Documentation

6.105.2.1 struct bio_queue_head [pt_softc::bio_queue](#)

Definition at line 79 of file `scsi_pt.c`.

Referenced by `ptdone()`, `ptstart()`, and `ptstrategy()`.

6.105.2.2 struct devstat* [pt_softc::device_stats](#)

Definition at line 80 of file `scsi_pt.c`.

Referenced by `ptdone()`, `ptdtor()`, and `ptstart()`.

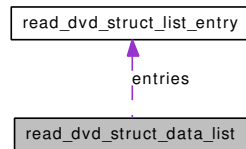
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_pt.c`

6.106 read_dvd_struct_data_list Struct Reference

```
#include <scsi_cd.h>
```

Collaboration diagram for read_dvd_struct_data_list:



Data Fields

- `u_int8_t data_len` [2]
- `u_int8_t reserved` [2]
- `read_dvd_struct_list_entry entries` [0]

6.106.1 Detailed Description

Definition at line 639 of file `scsi_cd.h`.

6.106.2 Field Documentation

6.106.2.1 `u_int8_t read_dvd_struct_data_list::data_len`[2]

Definition at line 641 of file `scsi_cd.h`.

6.106.2.2 `struct read_dvd_struct_list_entry read_dvd_struct_data_list::entries`[0]

Definition at line 643 of file `scsi_cd.h`.

6.106.2.3 `u_int8_t read_dvd_struct_data_list::reserved`[2]

Definition at line 642 of file `scsi_cd.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.h`

6.107 read_dvd_struct_list_entry Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- [u_int8_t format_code](#)
- [u_int8_t sds_rds](#)
- [u_int8_t struct_len](#) [2]

6.107.1 Detailed Description

Definition at line 626 of file [scsi_cd.h](#).

6.107.2 Field Documentation

6.107.2.1 [u_int8_t read_dvd_struct_list_entry::format_code](#)

Definition at line 628 of file [scsi_cd.h](#).

6.107.2.2 [u_int8_t read_dvd_struct_list_entry::sds_rds](#)

Definition at line 629 of file [scsi_cd.h](#).

6.107.2.3 [u_int8_t read_dvd_struct_list_entry::struct_len](#)[2]

Definition at line 636 of file [scsi_cd.h](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_cd.h](#)

6.108 read_dvd_struct_write_prot Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- [u_int8_t data_len](#) [2]
- [u_int8_t reserved0](#) [2]
- [u_int8_t write_prot_status](#)
- [u_int8_t reserved](#) [3]

6.108.1 Detailed Description

Definition at line 614 of file `scsi_cd.h`.

6.108.2 Field Documentation

6.108.2.1 [u_int8_t read_dvd_struct_write_prot::data_len](#)[2]

Definition at line 616 of file `scsi_cd.h`.

6.108.2.2 [u_int8_t read_dvd_struct_write_prot::reserved](#)[3]

Definition at line 623 of file `scsi_cd.h`.

6.108.2.3 [u_int8_t read_dvd_struct_write_prot::reserved0](#)[2]

Definition at line 617 of file `scsi_cd.h`.

6.108.2.4 [u_int8_t read_dvd_struct_write_prot::write_prot_status](#)

Definition at line 618 of file `scsi_cd.h`.

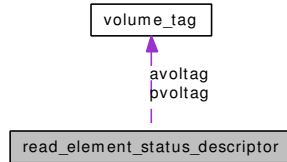
The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_cd.h](#)

6.109 read_element_status_descriptor Struct Reference

```
#include <scsi_ch.h>
```

Collaboration diagram for read_element_status_descriptor:



Data Fields

- [u_int8_t eaddr \[2\]](#)
- [u_int8_t flags1](#)
- [u_int8_t reserved0](#)
- [u_int8_t sense_code](#)
- [u_int8_t sense_qual](#)
- [u_int8_t dt_scsi_flags](#)
- [u_int8_t dt_scsi_addr](#)
- [u_int8_t reserved1](#)
- [u_int8_t flags2](#)
- [u_int8_t ssea \[2\]](#)
- [volume_tag pvoltag](#)
- [volume_tag avoltag](#)

6.109.1 Detailed Description

Definition at line 206 of file scsi_ch.h.

6.109.2 Field Documentation

6.109.2.1 struct [volume_tag](#) [read_element_status_descriptor::avoltag](#)

Definition at line 247 of file scsi_ch.h.

6.109.2.2 [u_int8_t](#) [read_element_status_descriptor::dt_scsi_addr](#)

Definition at line 237 of file scsi_ch.h.

6.109.2.3 [u_int8_t](#) [read_element_status_descriptor::dt_scsi_flags](#)

Definition at line 230 of file scsi_ch.h.

6.109.2.4 u_int8_t read_element_status_descriptor::eaddr[2]

Definition at line 207 of file scsi_ch.h.

Referenced by copy_element_status().

6.109.2.5 u_int8_t read_element_status_descriptor::flags1

Definition at line 208 of file scsi_ch.h.

6.109.2.6 u_int8_t read_element_status_descriptor::flags2

Definition at line 241 of file scsi_ch.h.

6.109.2.7 struct volume_tag read_element_status_descriptor::pvoltag

Definition at line 246 of file scsi_ch.h.

6.109.2.8 u_int8_t read_element_status_descriptor::reserved0

Definition at line 222 of file scsi_ch.h.

6.109.2.9 u_int8_t read_element_status_descriptor::reserved1

Definition at line 239 of file scsi_ch.h.

6.109.2.10 u_int8_t read_element_status_descriptor::sense_code

Definition at line 223 of file scsi_ch.h.

6.109.2.11 u_int8_t read_element_status_descriptor::sense_qual

Definition at line 224 of file scsi_ch.h.

6.109.2.12 u_int8_t read_element_status_descriptor::ssea[2]

Definition at line 244 of file scsi_ch.h.

The documentation for this struct was generated from the following file:

- /usr/src/sys/cam/scsi/scsi_ch.h

6.110 read_element_status_header Struct Reference

```
#include <scsi_ch.h>
```

Data Fields

- [u_int8_t fear](#) [2]
- [u_int8_t count](#) [2]
- [u_int8_t reserved](#)
- [u_int8_t nbytes](#) [3]

6.110.1 Detailed Description

Definition at line 179 of file `scsi_ch.h`.

6.110.2 Field Documentation

6.110.2.1 [u_int8_t read_element_status_header::count](#)[2]

Definition at line 181 of file `scsi_ch.h`.

Referenced by `chgetelemstatus()`.

6.110.2.2 [u_int8_t read_element_status_header::fear](#)[2]

Definition at line 180 of file `scsi_ch.h`.

6.110.2.3 [u_int8_t read_element_status_header::nbytes](#)[3]

Definition at line 183 of file `scsi_ch.h`.

6.110.2.4 [u_int8_t read_element_status_header::reserved](#)

Definition at line 182 of file `scsi_ch.h`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_ch.h](#)

6.111 read_element_status_page_header Struct Reference

```
#include <scsi_ch.h>
```

Data Fields

- [u_int8_t type](#)
- [u_int8_t flags](#)
- [u_int8_t edl \[2\]](#)
- [u_int8_t reserved](#)
- [u_int8_t nbytes \[3\]](#)

6.111.1 Detailed Description

Definition at line 186 of file `scsi_ch.h`.

6.111.2 Field Documentation

6.111.2.1 [u_int8_t read_element_status_page_header::edl\[2\]](#)

Definition at line 191 of file `scsi_ch.h`.

Referenced by `chgetelemstatus()`.

6.111.2.2 [u_int8_t read_element_status_page_header::flags](#)

Definition at line 188 of file `scsi_ch.h`.

Referenced by `chgetelemstatus()`.

6.111.2.3 [u_int8_t read_element_status_page_header::nbytes\[3\]](#)

Definition at line 193 of file `scsi_ch.h`.

6.111.2.4 [u_int8_t read_element_status_page_header::reserved](#)

Definition at line 192 of file `scsi_ch.h`.

6.111.2.5 [u_int8_t read_element_status_page_header::type](#)

Definition at line 187 of file `scsi_ch.h`.

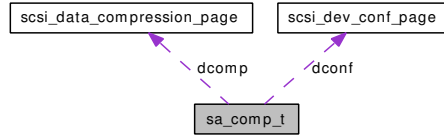
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_ch.h`

6.112 sa_comp_t Union Reference

```
#include <scsi_sa.h>
```

Collaboration diagram for sa_comp_t:



Data Fields

- struct {
 - u_int8_t [pagecode](#)
 - u_int8_t [pagelength](#)
 } [hdr](#)
- [scsi_dev_conf_page](#) [dconf](#)
- [scsi_data_compression_page](#) [dcomp](#)

6.112.1 Detailed Description

Definition at line 218 of file `scsi_sa.h`.

6.112.2 Field Documentation

6.112.2.1 struct [scsi_data_compression_page](#) [sa_comp_t::dcomp](#)

Definition at line 221 of file `scsi_sa.h`.

Referenced by `sagetparams()`, and `sasetparams()`.

6.112.2.2 struct [scsi_dev_conf_page](#) [sa_comp_t::dconf](#)

Definition at line 220 of file `scsi_sa.h`.

Referenced by `sagetparams()`, and `sasetparams()`.

6.112.2.3 struct { ... } [sa_comp_t::hdr](#)

Referenced by `sasetparams()`.

6.112.2.4 u_int8_t [sa_comp_t::pagecode](#)

Definition at line 219 of file `scsi_sa.h`.

Referenced by `sasetparams()`.

6.112.2.5 u_int8_t sa_comp_t::pagelength

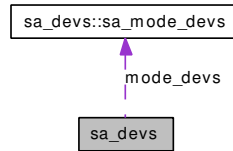
Definition at line 219 of file scsi_sa.h.

The documentation for this union was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_sa.h](#)

6.113 sa_devs Struct Reference

Collaboration diagram for sa_devs:



Data Fields

- `cdev * ctl_dev`
- `sa_devs::sa_mode_devs mode_devs [SA_NUM_MODES]`

Data Structures

- struct `sa_mode_devs`

6.113.1 Detailed Description

Definition at line 196 of file `scsi_sa.c`.

6.113.2 Field Documentation

6.113.2.1 struct `cdev* sa_devs::ctl_dev`

Definition at line 197 of file `scsi_sa.c`.

Referenced by `sacleanup()`.

6.113.2.2 struct `sa_devs::sa_mode_devs sa_devs::mode_devs[SA_NUM_MODES]`

Referenced by `sacleanup()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_sa.c`

6.114 sa_devs::sa_mode_devs Struct Reference

Data Fields

- cdev * [r_dev](#)
- cdev * [nr_dev](#)
- cdev * [er_dev](#)

6.114.1 Detailed Description

Definition at line 198 of file scsi_sa.c.

6.114.2 Field Documentation

6.114.2.1 struct cdev* [sa_devs::sa_mode_devs::er_dev](#)

Definition at line 201 of file scsi_sa.c.

Referenced by `sacleanup()`.

6.114.2.2 struct cdev* [sa_devs::sa_mode_devs::nr_dev](#)

Definition at line 200 of file scsi_sa.c.

Referenced by `sacleanup()`.

6.114.2.3 struct cdev* [sa_devs::sa_mode_devs::r_dev](#)

Definition at line 199 of file scsi_sa.c.

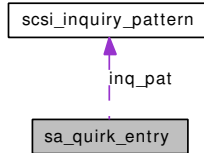
Referenced by `sacleanup()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_sa.c`

6.115 sa_quirk_entry Struct Reference

Collaboration diagram for sa_quirk_entry:



Data Fields

- [scsi_inquiry_pattern inq_pat](#)
- [sa_quirks quirks](#)
- [u_int32_t prefblk](#)

6.115.1 Detailed Description

Definition at line 265 of file scsi_sa.c.

6.115.2 Field Documentation

6.115.2.1 struct [scsi_inquiry_pattern](#) sa_quirk_entry::inq_pat

Definition at line 266 of file scsi_sa.c.

6.115.2.2 [u_int32_t](#) sa_quirk_entry::prefblk

Definition at line 268 of file scsi_sa.c.

6.115.2.3 [sa_quirks](#) sa_quirk_entry::quirks

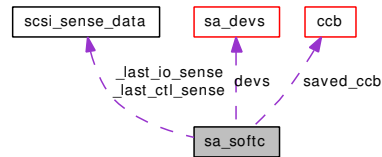
Definition at line 267 of file scsi_sa.c.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_sa.c](#)

6.116 sa_softc Struct Reference

Collaboration diagram for sa_softc:



Data Fields

- sa_state state
- sa_flags flags
- sa_quirks quirks
- bio_queue_head bio_queue
- int queue_count
- devstat * device_stats
- sa_devs devs
- int blk_gran
- int blk_mask
- int blk_shift
- u_int32_t max_blk
- u_int32_t min_blk
- u_int32_t comp_algorithm
- u_int32_t saved_comp_algorithm
- u_int32_t media_blksize
- u_int32_t last_media_blksize
- u_int32_t media_numblks
- u_int8_t media_density
- u_int8_t speed
- u_int8_t scsi_rev
- u_int8_t dsreg
- int buffer_mode
- int filemarks
- ccb saved_ccb
- int last_resid_was_io
- daddr_t fileno
- daddr_t blkno
- struct {
 - scsi_sense_data _last_io_sense
 - u_int32_t _last_io_resid
 - u_int8_t _last_io_cdb [CAM_MAX_CDBLEN]
 - scsi_sense_data _last_ctl_sense
 - u_int32_t _last_ctl_resid
 - u_int8_t _last_ctl_cdb [CAM_MAX_CDBLEN]
 } errinfo
- u_int32_t __pad0__: 29

- `u_int32_t open_ronly`: 1
- `u_int32_t open_pending_mount`: 1
- `u_int32_t ctrl_mode`: 1

6.116.1 Detailed Description

Definition at line 205 of file `scsi_sa.c`.

6.116.2 Field Documentation

6.116.2.1 `u_int32_t sa_softc::__pad0__`

Definition at line 259 of file `scsi_sa.c`.

6.116.2.2 `u_int8_t sa_softc::last_ctl_cdb`[CAM_MAX_CDBLEN]

Definition at line 247 of file `scsi_sa.c`.

6.116.2.3 `u_int32_t sa_softc::last_ctl_resid`

Definition at line 246 of file `scsi_sa.c`.

6.116.2.4 `struct scsi_sense_data sa_softc::last_ctl_sense`

Definition at line 245 of file `scsi_sa.c`.

6.116.2.5 `u_int8_t sa_softc::last_io_cdb`[CAM_MAX_CDBLEN]

Definition at line 244 of file `scsi_sa.c`.

6.116.2.6 `u_int32_t sa_softc::last_io_resid`

Definition at line 243 of file `scsi_sa.c`.

6.116.2.7 `struct scsi_sense_data sa_softc::last_io_sense`

Definition at line 242 of file `scsi_sa.c`.

6.116.2.8 `struct bio_queue_head sa_softc::bio_queue`

Definition at line 209 of file `scsi_sa.c`.

Referenced by `sadone()`, `saoninvalidate()`, `sastart()`, and `sastrategy()`.

6.116.2.9 int sa_softc::blk_gran

Definition at line 213 of file scsi_sa.c.

Referenced by saioctl(), samount(), and sastrategy().

6.116.2.10 int sa_softc::blk_mask

Definition at line 214 of file scsi_sa.c.

Referenced by saioctl(), and sastrategy().

6.116.2.11 int sa_softc::blk_shift

Definition at line 215 of file scsi_sa.c.

Referenced by sadone(), saioctl(), and sastart().

6.116.2.12 daddr_t sa_softc::blkno

Definition at line 236 of file scsi_sa.c.

Referenced by sadone(), saerror(), saioctl(), saloadunload(), saretension(), sarewind(), sasetpos(), saspace(), and sawritefilemarks().

6.116.2.13 int sa_softc::buffer_mode

Definition at line 227 of file scsi_sa.c.

Referenced by saioctl(), samount(), and sasetparams().

6.116.2.14 u_int32_t sa_softc::comp_algorithm

Definition at line 218 of file scsi_sa.c.

Referenced by saioctl(), samount(), and sasetparams().

6.116.2.15 u_int32_t sa_softc::ctrl_mode

Definition at line 259 of file scsi_sa.c.

Referenced by saclose(), and saopen().

6.116.2.16 struct devstat* sa_softc::device_stats

Definition at line 211 of file scsi_sa.c.

Referenced by sacleanup(), sadone(), saerase(), sasetparams(), saloadunload(), samount(), saprevent(), sardpos(), sareservereleaseunit(), saretension(), sarewind(), sasetparams(), sasetpos(), saspace(), sastart(), and sawritefilemarks().

6.116.2.17 struct [sa_devs sa_softc::devs](#)

Definition at line 212 of file `scsi_sa.c`.

Referenced by `sacleanup()`.

6.116.2.18 u_int8_t [sa_softc::dsreg](#)

Definition at line 226 of file `scsi_sa.c`.

Referenced by `sadone()`, `saerase()`, `saiocctl()`, `saloadunload()`, `sardpos()`, `sareservereleaseunit()`, `saretension()`, `sarewind()`, `sasetpos()`, `saspace()`, `sastart()`, and `sawritefilemarks()`.

6.116.2.19 struct { ... } [sa_softc::errinfo](#)

Referenced by `saiocctl()`.

6.116.2.20 int [sa_softc::filemarks](#)

Definition at line 228 of file `scsi_sa.c`.

Referenced by `saclose()`, `sadone()`, `saiocctl()`, `samarkswanted()`, `samount()`, and `sawritefilemarks()`.

6.116.2.21 daddr_t [sa_softc::fileno](#)

Definition at line 235 of file `scsi_sa.c`.

Referenced by `saerror()`, `saiocctl()`, `saloadunload()`, `saretension()`, `sarewind()`, `sasetpos()`, `saspace()`, and `sawritefilemarks()`.

6.116.2.22 sa_flags [sa_softc::flags](#)

Definition at line 207 of file `scsi_sa.c`.

Referenced by `saclose()`, `sadone()`, `saerror()`, `saiocctl()`, `samarkswanted()`, `samount()`, `saoninvalidate()`, `saopen()`, `saprevent()`, `sardpos()`, `sasetparams()`, `sastart()`, and `sastrategy()`.

6.116.2.23 u_int32_t [sa_softc::last_media_blksize](#)

Definition at line 221 of file `scsi_sa.c`.

Referenced by `saiocctl()`, and `samount()`.

6.116.2.24 int [sa_softc::last_resid_was_io](#)

Definition at line 230 of file `scsi_sa.c`.

Referenced by `saerror()`, and `saiocctl()`.

6.116.2.25 `u_int32_t sa_softc::max_blk`

Definition at line 216 of file `scsi_sa.c`.

Referenced by `saiocntl()`, `samount()`, and `sastrategy()`.

6.116.2.26 `u_int32_t sa_softc::media_blksize`

Definition at line 220 of file `scsi_sa.c`.

Referenced by `sadone()`, `saerror()`, `saiocntl()`, `samount()`, and `sastart()`.

6.116.2.27 `u_int8_t sa_softc::media_density`

Definition at line 223 of file `scsi_sa.c`.

Referenced by `saiocntl()`, `samount()`, and `sasetparams()`.

6.116.2.28 `u_int32_t sa_softc::media_numblks`

Definition at line 222 of file `scsi_sa.c`.

Referenced by `saiocntl()`, and `samount()`.

6.116.2.29 `u_int32_t sa_softc::min_blk`

Definition at line 217 of file `scsi_sa.c`.

Referenced by `samount()`, and `sastrategy()`.

6.116.2.30 `u_int32_t sa_softc::open_pending_mount`

Definition at line 259 of file `scsi_sa.c`.

Referenced by `saclose()`, `saiocntl()`, `saopen()`, and `sastrategy()`.

6.116.2.31 `u_int32_t sa_softc::open_ronly`

Definition at line 259 of file `scsi_sa.c`.

Referenced by `saclose()`, `saerase()`, `saopen()`, `sastrategy()`, and `sawritefilemarks()`.

6.116.2.32 `int sa_softc::queue_count`

Definition at line 210 of file `scsi_sa.c`.

Referenced by `saoninvalidate()`, `sastart()`, and `sastrategy()`.

6.116.2.33 `sa_quirks sa_softc::quirks`

Definition at line 208 of file `scsi_sa.c`.

Referenced by `saclose()`, `saerror()`, `sagetparams()`, `saiocntl()`, `samarkswanted()`, and `samount()`.

6.116.2.34 union [ccb sa_softc::saved_ccb](#)

Definition at line 229 of file `scsi_sa.c`.

Referenced by `saerror()`.

6.116.2.35 u_int32_t [sa_softc::saved_comp_algorithm](#)

Definition at line 219 of file `scsi_sa.c`.

Referenced by `saiioctl()`, and `sasetparams()`.

6.116.2.36 u_int8_t [sa_softc::scsi_rev](#)

Definition at line 225 of file `scsi_sa.c`.

Referenced by `samount()`, and `sasetparams()`.

6.116.2.37 u_int8_t [sa_softc::speed](#)

Definition at line 224 of file `scsi_sa.c`.

Referenced by `saiioctl()`, and `samount()`.

6.116.2.38 sa_state [sa_softc::state](#)

Definition at line 206 of file `scsi_sa.c`.

Referenced by `sastart()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_sa.c`

6.117 sc_p Struct Reference

```
#include <scsi_low.h>
```

Data Fields

- `u_int8_t * scp_data`
- `int scp_dataalen`
- `u_int8_t * scp_cmd`
- `int scp_cmdlen`
- `u_int8_t scp_direction`
- `u_int8_t scp_status`
- `u_int8_t scp_spare [2]`

6.117.1 Detailed Description

Definition at line 208 of file `scsi_low.h`.

6.117.2 Field Documentation

6.117.2.1 `u_int8_t* sc_p::scp_cmd`

Definition at line 212 of file `scsi_low.h`.

Referenced by `scsi_low_done()`, `scsi_low_sense_abort_start()`, `scsi_low_setup_start()`, `scsi_low_start()`, and `scsi_low_unit_ready_cmd()`.

6.117.2.2 `int sc_p::scp_cmdlen`

Definition at line 213 of file `scsi_low.h`.

Referenced by `scsi_low_done()`, `scsi_low_sense_abort_start()`, and `scsi_low_setup_start()`.

6.117.2.3 `u_int8_t* sc_p::scp_data`

Definition at line 209 of file `scsi_low.h`.

Referenced by `scsi_low_msginfunc_sdp()`, `scsi_low_print()`, `scsi_low_sense_abort_start()`, and `scsi_low_setup_start()`.

6.117.2.4 `int sc_p::scp_dataalen`

Definition at line 210 of file `scsi_low.h`.

Referenced by `scsi_low_done()`, `scsi_low_msginfunc_ext()`, `scsi_low_msginfunc_i_wide_residue()`, `scsi_low_msginfunc_sdp()`, `scsi_low_print()`, `scsi_low_sense_abort_start()`, and `scsi_low_setup_start()`.

6.117.2.5 u_int8_t sc_p::scp_direction

Definition at line 215 of file scsi_low.h.

Referenced by scsi_low_data(), scsi_low_sense_abort_start(), and scsi_low_setup_start().

6.117.2.6 u_int8_t sc_p::scp_spare[2]

Definition at line 220 of file scsi_low.h.

6.117.2.7 u_int8_t sc_p::scp_status

Definition at line 219 of file scsi_low.h.

Referenced by scsi_low_done(), scsi_low_msginfunc_lcc(), scsi_low_print(), and scsi_low_start().

The documentation for this struct was generated from the following file:

- /usr/src/sys/cam/scsi/scsi_low.h

6.118 scfg Struct Reference

Data Fields

- [uint8_t Nfans](#)
- [uint8_t Npwr](#)
- [uint8_t Nslots](#)
- [uint8_t DoorLock](#)
- [uint8_t Ntherm](#)
- [uint8_t Nspkrs](#)
- [uint8_t Nalarm](#)
- [uint8_t flag1](#)
- [uint8_t flag2](#)
- [uint8_t pwroff](#)
- [uint8_t slotoff](#)

6.118.1 Detailed Description

Definition at line 1592 of file scsi_ses.c.

6.118.2 Field Documentation

6.118.2.1 [uint8_t scfg::DoorLock](#)

Definition at line 1599 of file scsi_ses.c.

Referenced by [safte_getconfig\(\)](#), and [safte_softc_init\(\)](#).

6.118.2.2 [uint8_t scfg::flag1](#)

Definition at line 1606 of file scsi_ses.c.

Referenced by [safte_set_encstat\(\)](#), [safte_set_objstat\(\)](#), and [set_objstat_sel\(\)](#).

6.118.2.3 [uint8_t scfg::flag2](#)

Definition at line 1607 of file scsi_ses.c.

Referenced by [safte_set_encstat\(\)](#), [safte_set_objstat\(\)](#), and [set_objstat_sel\(\)](#).

6.118.2.4 [uint8_t scfg::Nalarm](#)

Definition at line 1602 of file scsi_ses.c.

Referenced by [safte_getconfig\(\)](#).

6.118.2.5 [uint8_t scfg::Nfans](#)

Definition at line 1596 of file scsi_ses.c.

Referenced by [safte_getconfig\(\)](#), [safte_rdstat\(\)](#), and [safte_softc_init\(\)](#).

6.118.2.6 uint8_t scfg::Npwr

Definition at line 1597 of file scsi_ses.c.

Referenced by safte_getconfig(), and safte_softc_init().

6.118.2.7 uint8_t scfg::Nslots

Definition at line 1598 of file scsi_ses.c.

Referenced by safte_getconfig(), safte_rdstat(), safte_softc_init(), and wrslot_stat().

6.118.2.8 uint8_t scfg::Nspkrs

Definition at line 1601 of file scsi_ses.c.

Referenced by safte_getconfig(), and safte_softc_init().

6.118.2.9 uint8_t scfg::Ntherm

Definition at line 1600 of file scsi_ses.c.

Referenced by safte_getconfig(), safte_rdstat(), and safte_softc_init().

6.118.2.10 uint8_t scfg::pwroff

Definition at line 1611 of file scsi_ses.c.

Referenced by safte_set_objstat(), safte_softc_init(), and set_objstat_sel().

6.118.2.11 uint8_t scfg::slotoff

Definition at line 1612 of file scsi_ses.c.

Referenced by safte_set_objstat(), safte_softc_init(), and wrslot_stat().

The documentation for this struct was generated from the following file:

- /usr/src/sys/cam/scsi/scsi_ses.c

6.119 scsi_changedef Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t byte2](#)
- [u_int8_t unused1](#)
- [u_int8_t how](#)
- [u_int8_t unused](#) [4]
- [u_int8_t datalen](#)
- [u_int8_t control](#)

6.119.1 Detailed Description

Definition at line 372 of file scsi_all.h.

6.119.2 Field Documentation

6.119.2.1 [u_int8_t scsi_changedef::byte2](#)

Definition at line 375 of file scsi_all.h.

6.119.2.2 [u_int8_t scsi_changedef::control](#)

Definition at line 380 of file scsi_all.h.

6.119.2.3 [u_int8_t scsi_changedef::datalen](#)

Definition at line 379 of file scsi_all.h.

6.119.2.4 [u_int8_t scsi_changedef::how](#)

Definition at line 377 of file scsi_all.h.

6.119.2.5 [u_int8_t scsi_changedef::opcode](#)

Definition at line 374 of file scsi_all.h.

6.119.2.6 [u_int8_t scsi_changedef::unused](#)[4]

Definition at line 378 of file scsi_all.h.

6.119.2.7 `u_int8_t scsi_changedef::unused1`

Definition at line 376 of file `scsi_all.h`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_all.h](#)

6.120 `scsi_control_page` Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- `u_int8_t page_code`
- `u_int8_t page_length`
- `u_int8_t rlec`
- `u_int8_t queue_flags`
- `u_int8_t eca_and_aen`
- `u_int8_t reserved`
- `u_int8_t aen_holdoff_period` [2]

6.120.1 Detailed Description

Definition at line 312 of file `scsi_all.h`.

6.120.2 Field Documentation

6.120.2.1 `u_int8_t scsi_control_page::aen_holdoff_period`[2]

Definition at line 329 of file `scsi_all.h`.

6.120.2.2 `u_int8_t scsi_control_page::eca_and_aen`

Definition at line 323 of file `scsi_all.h`.

6.120.2.3 `u_int8_t scsi_control_page::page_code`

Definition at line 313 of file `scsi_all.h`.

6.120.2.4 `u_int8_t scsi_control_page::page_length`

Definition at line 314 of file `scsi_all.h`.

6.120.2.5 `u_int8_t scsi_control_page::queue_flags`

Definition at line 317 of file `scsi_all.h`.

Referenced by `probedone()`.

6.120.2.6 `u_int8_t scsi_control_page::reserved`

Definition at line 328 of file `scsi_all.h`.

6.120.2.7 `u_int8_t scsi_control_page::rlec`

Definition at line 315 of file `scsi_all.h`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_all.h](#)

6.121 scsi_da_rw_recovery_page Struct Reference

```
#include <scsi_da.h>
```

Data Fields

- [u_int8_t page_code](#)
- [u_int8_t page_length](#)
- [u_int8_t byte3](#)
- [u_int8_t read_retry_count](#)
- [u_int8_t correction_span](#)
- [u_int8_t head_offset_count](#)
- [u_int8_t data_strobe_offset_cnt](#)
- [u_int8_t reserved](#)
- [u_int8_t write_retry_count](#)
- [u_int8_t reserved2](#)
- [u_int8_t recovery_time_limit](#) [2]

6.121.1 Detailed Description

Definition at line 424 of file `scsi_da.h`.

6.121.2 Field Documentation

6.121.2.1 [u_int8_t scsi_da_rw_recovery_page::byte3](#)

Definition at line 428 of file `scsi_da.h`.

6.121.2.2 [u_int8_t scsi_da_rw_recovery_page::correction_span](#)

Definition at line 438 of file `scsi_da.h`.

6.121.2.3 [u_int8_t scsi_da_rw_recovery_page::data_strobe_offset_cnt](#)

Definition at line 440 of file `scsi_da.h`.

6.121.2.4 [u_int8_t scsi_da_rw_recovery_page::head_offset_count](#)

Definition at line 439 of file `scsi_da.h`.

6.121.2.5 [u_int8_t scsi_da_rw_recovery_page::page_code](#)

Definition at line 425 of file `scsi_da.h`.

6.121.2.6 [u_int8_t scsi_da_rw_recovery_page::page_length](#)

Definition at line 427 of file `scsi_da.h`.

6.121.2.7 `u_int8_t scsi_da_rw_recovery_page::read_retry_count`

Definition at line 437 of file `scsi_da.h`.

6.121.2.8 `u_int8_t scsi_da_rw_recovery_page::recovery_time_limit[2]`

Definition at line 444 of file `scsi_da.h`.

6.121.2.9 `u_int8_t scsi_da_rw_recovery_page::reserved`

Definition at line 441 of file `scsi_da.h`.

6.121.2.10 `u_int8_t scsi_da_rw_recovery_page::reserved2`

Definition at line 443 of file `scsi_da.h`.

6.121.2.11 `u_int8_t scsi_da_rw_recovery_page::write_retry_count`

Definition at line 442 of file `scsi_da.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_da.h`

6.122 scsi_data_compression_page Struct Reference

```
#include <scsi_sa.h>
```

Data Fields

- [u_int8_t page_code](#)
- [u_int8_t page_length](#)
- [u_int8_t dce_and_dcc](#)
- [u_int8_t dde_and_red](#)
- [u_int8_t comp_algorithm](#) [4]
- [u_int8_t decomp_algorithm](#) [4]
- [u_int8_t reserved](#) [4]

6.122.1 Detailed Description

Definition at line 200 of file `scsi_sa.h`.

6.122.2 Field Documentation

6.122.2.1 [u_int8_t scsi_data_compression_page::comp_algorithm](#)[4]

Definition at line 213 of file `scsi_sa.h`.

Referenced by `sasetparams()`, and `sasetparams()`.

6.122.2.2 [u_int8_t scsi_data_compression_page::dce_and_dcc](#)

Definition at line 203 of file `scsi_sa.h`.

Referenced by `sasetparams()`, and `sasetparams()`.

6.122.2.3 [u_int8_t scsi_data_compression_page::dde_and_red](#)

Definition at line 206 of file `scsi_sa.h`.

Referenced by `sasetparams()`.

6.122.2.4 [u_int8_t scsi_data_compression_page::decomp_algorithm](#)[4]

Definition at line 214 of file `scsi_sa.h`.

Referenced by `sasetparams()`.

6.122.2.5 [u_int8_t scsi_data_compression_page::page_code](#)

Definition at line 201 of file `scsi_sa.h`.

6.122.2.6 `u_int8_t scsi_data_compression_page::page_length`

Definition at line 202 of file `scsi_sa.h`.

6.122.2.7 `u_int8_t scsi_data_compression_page::reserved[4]`

Definition at line 215 of file `scsi_sa.h`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_sa.h](#)

6.123 `scsi_defect_desc_block` Struct Reference

```
#include <scsi_da.h>
```

Data Fields

- `u_int8_t address` [4]

6.123.1 Detailed Description

Definition at line 292 of file `scsi_da.h`.

6.123.2 Field Documentation

6.123.2.1 `u_int8_t scsi_defect_desc_block::address`[4]

Definition at line 294 of file `scsi_da.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_da.h`

6.124 scsi_defect_desc_bytes_from_index Struct Reference

```
#include <scsi_da.h>
```

Data Fields

- `u_int8_t cylinder` [3]
- `u_int8_t head`
- `u_int8_t bytes_from_index` [4]

6.124.1 Detailed Description

Definition at line 297 of file `scsi_da.h`.

6.124.2 Field Documentation

6.124.2.1 `u_int8_t scsi_defect_desc_bytes_from_index::bytes_from_index`[4]

Definition at line 301 of file `scsi_da.h`.

6.124.2.2 `u_int8_t scsi_defect_desc_bytes_from_index::cylinder`[3]

Definition at line 299 of file `scsi_da.h`.

6.124.2.3 `u_int8_t scsi_defect_desc_bytes_from_index::head`

Definition at line 300 of file `scsi_da.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_da.h`

6.125 `scsi_defect_desc_phys_sector` Struct Reference

```
#include <scsi_da.h>
```

Data Fields

- `u_int8_t cylinder` [3]
- `u_int8_t head`
- `u_int8_t sector` [4]

6.125.1 Detailed Description

Definition at line 304 of file `scsi_da.h`.

6.125.2 Field Documentation

6.125.2.1 `u_int8_t scsi_defect_desc_phys_sector::cylinder`[3]

Definition at line 306 of file `scsi_da.h`.

6.125.2.2 `u_int8_t scsi_defect_desc_phys_sector::head`

Definition at line 307 of file `scsi_da.h`.

6.125.2.3 `u_int8_t scsi_defect_desc_phys_sector::sector`[4]

Definition at line 308 of file `scsi_da.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_da.h`

6.126 scsi_dev_conf_page Struct Reference

```
#include <scsi_sa.h>
```

Data Fields

- [u_int8_t pagecode](#)
- [u_int8_t pagelength](#)
- [u_int8_t byte2](#)
- [u_int8_t active_partition](#)
- [u_int8_t wb_full_ratio](#)
- [u_int8_t rb_empty_ratio](#)
- [u_int8_t wrdelay_time \[2\]](#)
- [u_int8_t byte8](#)
- [u_int8_t gap_size](#)
- [u_int8_t byte10](#)
- [u_int8_t ew_bufsize \[3\]](#)
- [u_int8_t sel_comp_alg](#)
- [u_int8_t extra_wp](#)

6.126.1 Detailed Description

Definition at line 170 of file `scsi_sa.h`.

6.126.2 Field Documentation

6.126.2.1 [u_int8_t scsi_dev_conf_page::active_partition](#)

Definition at line 174 of file `scsi_sa.h`.

6.126.2.2 [u_int8_t scsi_dev_conf_page::byte10](#)

Definition at line 187 of file `scsi_sa.h`.

6.126.2.3 [u_int8_t scsi_dev_conf_page::byte2](#)

Definition at line 173 of file `scsi_sa.h`.

6.126.2.4 [u_int8_t scsi_dev_conf_page::byte8](#)

Definition at line 178 of file `scsi_sa.h`.

6.126.2.5 [u_int8_t scsi_dev_conf_page::ew_bufsize\[3\]](#)

Definition at line 188 of file `scsi_sa.h`.

6.126.2.6 `u_int8_t scsi_dev_conf_page::extra_wp`

Definition at line 193 of file `scsi_sa.h`.

6.126.2.7 `u_int8_t scsi_dev_conf_page::gap_size`

Definition at line 186 of file `scsi_sa.h`.

6.126.2.8 `u_int8_t scsi_dev_conf_page::pagecode`

Definition at line 171 of file `scsi_sa.h`.

6.126.2.9 `u_int8_t scsi_dev_conf_page::pagelength`

Definition at line 172 of file `scsi_sa.h`.

6.126.2.10 `u_int8_t scsi_dev_conf_page::rb_empty_ratio`

Definition at line 176 of file `scsi_sa.h`.

6.126.2.11 `u_int8_t scsi_dev_conf_page::sel_comp_alg`

Definition at line 189 of file `scsi_sa.h`.

Referenced by `sagetparams()`, and `sasetparams()`.

6.126.2.12 `u_int8_t scsi_dev_conf_page::wb_full_ratio`

Definition at line 175 of file `scsi_sa.h`.

6.126.2.13 `u_int8_t scsi_dev_conf_page::wrdelay_time[2]`

Definition at line 177 of file `scsi_sa.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_sa.h`

6.127 scsi_erase Struct Reference

```
#include <scsi_sa.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t lun_imm_long](#)
- [u_int8_t reserved](#) [3]
- [u_int8_t control](#)

6.127.1 Detailed Description

Definition at line 133 of file `scsi_sa.h`.

6.127.2 Field Documentation

6.127.2.1 [u_int8_t scsi_erase::control](#)

Definition at line 141 of file `scsi_sa.h`.

6.127.2.2 [u_int8_t scsi_erase::lun_imm_long](#)

Definition at line 136 of file `scsi_sa.h`.

6.127.2.3 [u_int8_t scsi_erase::opcode](#)

Definition at line 135 of file `scsi_sa.h`.

6.127.2.4 [u_int8_t scsi_erase::reserved](#)[3]

Definition at line 140 of file `scsi_sa.h`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_sa.h](#)

6.128 `scsi_exchange_medium` Struct Reference

```
#include <scsi_ch.h>
```

Data Fields

- `u_int8_t opcode`
- `u_int8_t byte2`
- `u_int8_t tea` [2]
- `u_int8_t src` [2]
- `u_int8_t fdst` [2]
- `u_int8_t sdst` [2]
- `u_int8_t invert`
- `u_int8_t control`

6.128.1 Detailed Description

Definition at line 76 of file `scsi_ch.h`.

6.128.2 Field Documentation

6.128.2.1 `u_int8_t scsi_exchange_medium::byte2`

Definition at line 79 of file `scsi_ch.h`.

6.128.2.2 `u_int8_t scsi_exchange_medium::control`

Definition at line 87 of file `scsi_ch.h`.

6.128.2.3 `u_int8_t scsi_exchange_medium::fdst`[2]

Definition at line 82 of file `scsi_ch.h`.

6.128.2.4 `u_int8_t scsi_exchange_medium::invert`

Definition at line 84 of file `scsi_ch.h`.

6.128.2.5 `u_int8_t scsi_exchange_medium::opcode`

Definition at line 77 of file `scsi_ch.h`.

6.128.2.6 `u_int8_t scsi_exchange_medium::sdst`[2]

Definition at line 83 of file `scsi_ch.h`.

6.128.2.7 `u_int8_t scsi_exchange_medium::src[2]`

Definition at line 81 of file `scsi_ch.h`.

6.128.2.8 `u_int8_t scsi_exchange_medium::tea[2]`

Definition at line 80 of file `scsi_ch.h`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_ch.h](#)

6.129 scsi_format_unit Struct Reference

```
#include <scsi_da.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t byte2](#)
- [u_int8_t vendor_specific](#)
- [u_int8_t interleave \[2\]](#)
- [u_int8_t control](#)

6.129.1 Detailed Description

Definition at line 70 of file `scsi_da.h`.

6.129.2 Field Documentation

6.129.2.1 [u_int8_t scsi_format_unit::byte2](#)

Definition at line 73 of file `scsi_da.h`.

6.129.2.2 [u_int8_t scsi_format_unit::control](#)

Definition at line 82 of file `scsi_da.h`.

6.129.2.3 [u_int8_t scsi_format_unit::interleave\[2\]](#)

Definition at line 81 of file `scsi_da.h`.

6.129.2.4 [u_int8_t scsi_format_unit::opcode](#)

Definition at line 72 of file `scsi_da.h`.

6.129.2.5 [u_int8_t scsi_format_unit::vendor_specific](#)

Definition at line 80 of file `scsi_da.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_da.h`

6.130 scsi_generic Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t bytes \[11\]](#)

6.130.1 Detailed Description

Definition at line 108 of file [scsi_all.h](#).

6.130.2 Field Documentation

6.130.2.1 [u_int8_t scsi_generic::bytes\[11\]](#)

Definition at line 111 of file [scsi_all.h](#).

6.130.2.2 [u_int8_t scsi_generic::opcode](#)

Definition at line 110 of file [scsi_all.h](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_all.h](#)

6.131 `scsi_initialize_element_status` Struct Reference

```
#include <scsi_ch.h>
```

Data Fields

- `u_int8_t opcode`
- `u_int8_t byte2`
- `u_int8_t reserved` [3]
- `u_int8_t control`

6.131.1 Detailed Description

Definition at line 94 of file `scsi_ch.h`.

6.131.2 Field Documentation

6.131.2.1 `u_int8_t scsi_initialize_element_status::byte2`

Definition at line 97 of file `scsi_ch.h`.

6.131.2.2 `u_int8_t scsi_initialize_element_status::control`

Definition at line 99 of file `scsi_ch.h`.

6.131.2.3 `u_int8_t scsi_initialize_element_status::opcode`

Definition at line 95 of file `scsi_ch.h`.

6.131.2.4 `u_int8_t scsi_initialize_element_status::reserved`[3]

Definition at line 98 of file `scsi_ch.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_ch.h`

6.132 scsi_inquiry Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t byte2](#)
- [u_int8_t page_code](#)
- [u_int8_t reserved](#)
- [u_int8_t length](#)
- [u_int8_t control](#)

6.132.1 Detailed Description

Definition at line 153 of file `scsi_all.h`.

6.132.2 Field Documentation

6.132.2.1 [u_int8_t scsi_inquiry::byte2](#)

Definition at line 156 of file `scsi_all.h`.

Referenced by `targbhdone()`.

6.132.2.2 [u_int8_t scsi_inquiry::control](#)

Definition at line 161 of file `scsi_all.h`.

6.132.2.3 [u_int8_t scsi_inquiry::length](#)

Definition at line 160 of file `scsi_all.h`.

Referenced by `targbhdone()`.

6.132.2.4 [u_int8_t scsi_inquiry::opcode](#)

Definition at line 155 of file `scsi_all.h`.

6.132.2.5 [u_int8_t scsi_inquiry::page_code](#)

Definition at line 158 of file `scsi_all.h`.

Referenced by `targbhdone()`.

6.132.2.6 u_int8_t scsi_inquiry::reserved

Definition at line 159 of file scsi_all.h.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_all.h](#)

6.133 scsi_inquiry_data Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- `u_int8_t device`
- `u_int8_t dev_qual2`
- `u_int8_t version`
- `u_int8_t response_format`
- `u_int8_t additional_length`
- `u_int8_t reserved`
- `u_int8_t spc2_flags`
- `u_int8_t flags`
- `char vendor [SID_VENDOR_SIZE]`
- `char product [SID_PRODUCT_SIZE]`
- `char revision [SID_REVISION_SIZE]`
- `u_int8_t vendor_specific0 [SID_VENDOR_SPECIFIC_0_SIZE]`
- `u_int8_t spi3data`
- `u_int8_t reserved2`
- `u_int8_t version1 [2]`
- `u_int8_t version2 [2]`
- `u_int8_t version3 [2]`
- `u_int8_t version4 [2]`
- `u_int8_t version5 [2]`
- `u_int8_t version6 [2]`
- `u_int8_t version7 [2]`
- `u_int8_t version8 [2]`
- `u_int8_t reserved3 [22]`
- `u_int8_t vendor_specific1 [SID_VENDOR_SPECIFIC_1_SIZE]`

6.133.1 Detailed Description

Definition at line 547 of file `scsi_all.h`.

6.133.2 Field Documentation

6.133.2.1 `u_int8_t scsi_inquiry_data::additional_length`

Definition at line 601 of file `scsi_all.h`.

Referenced by `probedone()`, and `sesasync()`.

6.133.2.2 `u_int8_t scsi_inquiry_data::dev_qual2`

Definition at line 585 of file `scsi_all.h`.

6.133.2.3 `u_int8_t scsi_inquiry_data::device`

Definition at line 549 of file `scsi_all.h`.

Referenced by `xpt_set_transfer_settings()`.

6.133.2.4 `u_int8_t scsi_inquiry_data::flags`

Definition at line 617 of file `scsi_all.h`.

Referenced by `cdregister()`, `daregister()`, `passregister()`, and `xpt_set_transfer_settings()`.

6.133.2.5 `char scsi_inquiry_data::product[SID_PRODUCT_SIZE]`

Definition at line 628 of file `scsi_all.h`.

Referenced by `scsi_print_inquiry()`.

6.133.2.6 `u_int8_t scsi_inquiry_data::reserved`

Definition at line 605 of file `scsi_all.h`.

6.133.2.7 `u_int8_t scsi_inquiry_data::reserved2`

Definition at line 647 of file `scsi_all.h`.

6.133.2.8 `u_int8_t scsi_inquiry_data::reserved3[22]`

Definition at line 660 of file `scsi_all.h`.

6.133.2.9 `u_int8_t scsi_inquiry_data::response_format`

Definition at line 598 of file `scsi_all.h`.

6.133.2.10 `char scsi_inquiry_data::revision[SID_REVISION_SIZE]`

Definition at line 630 of file `scsi_all.h`.

Referenced by `scsi_print_inquiry()`.

6.133.2.11 `u_int8_t scsi_inquiry_data::spc2_flags`

Definition at line 606 of file `scsi_all.h`.

6.133.2.12 `u_int8_t scsi_inquiry_data::spi3data`

Definition at line 646 of file `scsi_all.h`.

Referenced by `xpt_devisе_transport()`, and `xpt_set_transfer_settings()`.

6.133.2.13 `char scsi_inquiry_data::vendor[SID_VENDOR_SIZE]`

Definition at line 626 of file `scsi_all.h`.

Referenced by `scsi_print_inquiry()`.

6.133.2.14 `u_int8_t scsi_inquiry_data::vendor_specific0[SID_VENDOR_SPECIFIC_0_SIZE]`

Definition at line 636 of file `scsi_all.h`.

6.133.2.15 `u_int8_t scsi_inquiry_data::vendor_specific1[SID_VENDOR_SPECIFIC_1_SIZE]`

Definition at line 663 of file `scsi_all.h`.

6.133.2.16 `u_int8_t scsi_inquiry_data::version`

Definition at line 588 of file `scsi_all.h`.

6.133.2.17 `u_int8_t scsi_inquiry_data::version1[2]`

Definition at line 651 of file `scsi_all.h`.

6.133.2.18 `u_int8_t scsi_inquiry_data::version2[2]`

Definition at line 652 of file `scsi_all.h`.

6.133.2.19 `u_int8_t scsi_inquiry_data::version3[2]`

Definition at line 653 of file `scsi_all.h`.

6.133.2.20 `u_int8_t scsi_inquiry_data::version4[2]`

Definition at line 654 of file `scsi_all.h`.

6.133.2.21 `u_int8_t scsi_inquiry_data::version5[2]`

Definition at line 655 of file `scsi_all.h`.

6.133.2.22 `u_int8_t scsi_inquiry_data::version6[2]`

Definition at line 656 of file `scsi_all.h`.

6.133.2.23 `u_int8_t scsi_inquiry_data::version7[2]`

Definition at line 657 of file `scsi_all.h`.

6.133.2.24 `u_int8_t scsi_inquiry_data::version8[2]`

Definition at line 658 of file `scsi_all.h`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_all.h](#)

6.134 scsi_inquiry_pattern Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- [u_int8_t type](#)
- [u_int8_t media_type](#)
- [const char * vendor](#)
- [const char * product](#)
- [const char * revision](#)

6.134.1 Detailed Description

Definition at line 846 of file `scsi_all.h`.

6.134.2 Field Documentation

6.134.2.1 [u_int8_t scsi_inquiry_pattern::media_type](#)

Definition at line 848 of file `scsi_all.h`.

Referenced by `scsi_inquiry_match()`.

6.134.2.2 [const char* scsi_inquiry_pattern::product](#)

Definition at line 852 of file `scsi_all.h`.

Referenced by `scsi_inquiry_match()`.

6.134.2.3 [const char* scsi_inquiry_pattern::revision](#)

Definition at line 853 of file `scsi_all.h`.

Referenced by `scsi_inquiry_match()`.

6.134.2.4 [u_int8_t scsi_inquiry_pattern::type](#)

Definition at line 847 of file `scsi_all.h`.

Referenced by `scsi_inquiry_match()`.

6.134.2.5 [const char* scsi_inquiry_pattern::vendor](#)

Definition at line 851 of file `scsi_all.h`.

Referenced by `scsi_inquiry_match()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.135 scsi_load_unload Struct Reference

```
#include <scsi_sa.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t immediate](#)
- [u_int8_t reserved](#) [2]
- [u_int8_t eot_reten_load](#)
- [u_int8_t control](#)

6.135.1 Detailed Description

Definition at line 64 of file `scsi_sa.h`.

6.135.2 Field Documentation

6.135.2.1 [u_int8_t scsi_load_unload::control](#)

Definition at line 74 of file `scsi_sa.h`.

6.135.2.2 [u_int8_t scsi_load_unload::eot_reten_load](#)

Definition at line 70 of file `scsi_sa.h`.

6.135.2.3 [u_int8_t scsi_load_unload::immediate](#)

Definition at line 67 of file `scsi_sa.h`.

6.135.2.4 [u_int8_t scsi_load_unload::opcode](#)

Definition at line 66 of file `scsi_sa.h`.

6.135.2.5 [u_int8_t scsi_load_unload::reserved](#)[2]

Definition at line 69 of file `scsi_sa.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_sa.h`

6.136 scsi_log_header Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- [u_int8_t page](#)
- [u_int8_t reserved](#)
- [u_int8_t datalen](#) [2]

6.136.1 Detailed Description

Definition at line 288 of file `scsi_all.h`.

6.136.2 Field Documentation

6.136.2.1 [u_int8_t scsi_log_header::datalen](#)[2]

Definition at line 292 of file `scsi_all.h`.

6.136.2.2 [u_int8_t scsi_log_header::page](#)

Definition at line 290 of file `scsi_all.h`.

6.136.2.3 [u_int8_t scsi_log_header::reserved](#)

Definition at line 291 of file `scsi_all.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.137 scsi_log_param_header Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- `u_int8_t param_code` [2]
- `u_int8_t param_control`
- `u_int8_t param_len`

6.137.1 Detailed Description

Definition at line 295 of file `scsi_all.h`.

6.137.2 Field Documentation

6.137.2.1 `u_int8_t scsi_log_param_header::param_code`[2]

Definition at line 296 of file `scsi_all.h`.

6.137.2.2 `u_int8_t scsi_log_param_header::param_control`

Definition at line 297 of file `scsi_all.h`.

6.137.2.3 `u_int8_t scsi_log_param_header::param_len`

Definition at line 309 of file `scsi_all.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.138 scsi_log_select Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t byte2](#)
- [u_int8_t page](#)
- [u_int8_t reserved](#) [4]
- [u_int8_t length](#) [2]
- [u_int8_t control](#)

6.138.1 Detailed Description

Definition at line 271 of file [scsi_all.h](#).

6.138.2 Field Documentation

6.138.2.1 [u_int8_t scsi_log_select::byte2](#)

Definition at line 274 of file [scsi_all.h](#).

6.138.2.2 [u_int8_t scsi_log_select::control](#)

Definition at line 285 of file [scsi_all.h](#).

6.138.2.3 [u_int8_t scsi_log_select::length](#)[2]

Definition at line 284 of file [scsi_all.h](#).

6.138.2.4 [u_int8_t scsi_log_select::opcode](#)

Definition at line 273 of file [scsi_all.h](#).

6.138.2.5 [u_int8_t scsi_log_select::page](#)

Definition at line 277 of file [scsi_all.h](#).

6.138.2.6 [u_int8_t scsi_log_select::reserved](#)[4]

Definition at line 283 of file [scsi_all.h](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_all.h](#)

6.139 `scsi_log_sense` Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- `u_int8_t opcode`
- `u_int8_t byte2`
- `u_int8_t page`
- `u_int8_t reserved` [2]
- `u_int8_t paramptr` [2]
- `u_int8_t length` [2]
- `u_int8_t control`

6.139.1 Detailed Description

Definition at line 244 of file `scsi_all.h`.

6.139.2 Field Documentation

6.139.2.1 `u_int8_t scsi_log_sense::byte2`

Definition at line 247 of file `scsi_all.h`.

6.139.2.2 `u_int8_t scsi_log_sense::control`

Definition at line 268 of file `scsi_all.h`.

6.139.2.3 `u_int8_t scsi_log_sense::length`[2]

Definition at line 267 of file `scsi_all.h`.

6.139.2.4 `u_int8_t scsi_log_sense::opcode`

Definition at line 246 of file `scsi_all.h`.

6.139.2.5 `u_int8_t scsi_log_sense::page`

Definition at line 250 of file `scsi_all.h`.

6.139.2.6 `u_int8_t scsi_log_sense::paramptr`[2]

Definition at line 266 of file `scsi_all.h`.

6.139.2.7 `u_int8_t scsi_log_sense::reserved[2]`

Definition at line 265 of file `scsi_all.h`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_all.h](#)

6.140 `scsi_low_error_code` Struct Reference

Data Fields

- int [error_bits](#)
- int [error_code](#)

6.140.1 Detailed Description

Definition at line 339 of file `scsi_low.c`.

6.140.2 Field Documentation

6.140.2.1 int `scsi_low_error_code::error_bits`

Definition at line 340 of file `scsi_low.c`.

Referenced by `scsi_low_translate_error_code()`.

6.140.2.2 int `scsi_low_error_code::error_code`

Definition at line 341 of file `scsi_low.c`.

Referenced by `scsi_low_translate_error_code()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_low.c`

6.141 scsi_low_funcs Struct Reference

```
#include <scsi_low.h>
```

Data Fields

- [int\(* scsi_low_init\)\(sc_low_t, int\)](#)
- [void\(* scsi_low_bus_reset\)\(sc_low_t\)](#)
- [int\(* scsi_low_targ_init\)\(sc_low_t, struct targ_info *, int\)](#)
- [int\(* scsi_low_lun_init\)\(sc_low_t, struct targ_info *, struct lun_info *, int\)](#)
- [int\(* scsi_low_start_bus\)\(sc_low_t, struct slccb *\)](#)
- [int\(* scsi_low_establish_lun_nexus\)\(sc_low_t\)](#)
- [int\(* scsi_low_establish_ccb_nexus\)\(sc_low_t\)](#)
- [void\(* scsi_low_attention\)\(sc_low_t\)](#)
- [int\(* scsi_low_msg\)\(sc_low_t, struct targ_info *, u_int\)](#)
- [int\(* scsi_low_timeout\)\(sc_low_t\)](#)
- [int\(* scsi_low_poll\)\(void *\)](#)
- [int\(* scsi_low_power\)\(sc_low_t, u_int\)](#)
- [int\(* scsi_low_ioctl\)\(sc_low_t, u_long, caddr_t, int, struct proc *\)](#)

6.141.1 Detailed Description

Definition at line 529 of file scsi_low.h.

6.141.2 Field Documentation

6.141.2.1 [void\(* scsi_low_funcs::scsi_low_attention\)\(sc_low_t\)](#)

6.141.2.2 [void\(* scsi_low_funcs::scsi_low_bus_reset\)\(sc_low_t\)](#)

6.141.2.3 [int\(* scsi_low_funcs::scsi_low_establish_ccb_nexus\)\(sc_low_t\)](#)

Referenced by [scsi_low_establish_ccb\(\)](#).

6.141.2.4 [int\(* scsi_low_funcs::scsi_low_establish_lun_nexus\)\(sc_low_t\)](#)

6.141.2.5 [int\(* scsi_low_funcs::scsi_low_init\)\(sc_low_t, int\)](#)

6.141.2.6 [int\(* scsi_low_funcs::scsi_low_ioctl\)\(sc_low_t, u_long, caddr_t, int, struct proc *\)](#)

6.141.2.7 [int\(* scsi_low_funcs::scsi_low_lun_init\)\(sc_low_t, struct targ_info *, struct lun_info *, int\)](#)

Referenced by [scsi_low_alloc_li\(\)](#).

6.141.2.8 `int(* scsi_low_funcs::scsi_low_msg)(sc_low_t, struct targ_info *, u_int)`

6.141.2.9 `int(* scsi_low_funcs::scsi_low_poll)(void *)`

6.141.2.10 `int(* scsi_low_funcs::scsi_low_power)(sc_low_t, u_int)`

Referenced by `scsi_low_engage()`.

6.141.2.11 `int(* scsi_low_funcs::scsi_low_start_bus)(sc_low_t, struct slccb *)`

6.141.2.12 `int(* scsi_low_funcs::scsi_low_targ_init)(sc_low_t, struct targ_info *, int)`

6.141.2.13 `int(* scsi_low_funcs::scsi_low_timeout)(sc_low_t)`

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_low.h`

6.142 scsi_low_msg_log Struct Reference

```
#include <scsi_low.h>
```

Data Fields

- int [slml_ptr](#)
- struct {
 - u_int8_t [msg](#) [2]
 - } [slml_msg](#) [SCSI_LOW_MSG_LOG_DATALEN]

6.142.1 Detailed Description

Definition at line 382 of file [scsi_low.h](#).

6.142.2 Field Documentation

6.142.2.1 u_int8_t [scsi_low_msg_log::msg](#)[2]

Definition at line 385 of file [scsi_low.h](#).

6.142.2.2 struct { ... } [scsi_low_msg_log::slml_msg](#)[SCSI_LOW_MSG_LOG_DATALEN]

6.142.2.3 int [scsi_low_msg_log::slml_ptr](#)

Definition at line 383 of file [scsi_low.h](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_low.h](#)

6.143 scsi_low_msgin_data Struct Reference

Data Fields

- [u_int md_len](#)
- [int\(* md_msgfunc\)\(struct scsi_low_softc *\)](#)

6.143.1 Detailed Description

Definition at line 3229 of file `scsi_low.c`.

6.143.2 Field Documentation

6.143.2.1 [u_int scsi_low_msgin_data::md_len](#)

Definition at line 3230 of file `scsi_low.c`.

Referenced by `scsi_low_msgin()`.

6.143.2.2 [int\(* scsi_low_msgin_data::md_msgfunc\)\(struct scsi_low_softc *\)](#)

Referenced by `scsi_low_msgin()`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_low.c](#)

6.144 scsi_low_msgout_data Struct Reference

Data Fields

- `u_int md_flags`
- `u_int8_t md_msg`
- `int(* md_msgfunc)(struct scsi_low_softc *)`
- `int(* md_errfunc)(struct scsi_low_softc *, u_int)`
- `u_int md_condition`

6.144.1 Detailed Description

Definition at line 3186 of file `scsi_low.c`.

6.144.2 Field Documentation

6.144.2.1 `u_int scsi_low_msgout_data::md_condition`

Definition at line 3192 of file `scsi_low.c`.

Referenced by `scsi_low_msgout()`.

6.144.2.2 `int(* scsi_low_msgout_data::md_errfunc)(struct scsi_low_softc *, u_int)`

Referenced by `scsi_low_msginfunc_msg_reject()`.

6.144.2.3 `u_int scsi_low_msgout_data::md_flags`

Definition at line 3187 of file `scsi_low.c`.

Referenced by `scsi_low_msginfunc_msg_reject()`, and `scsi_low_msgout()`.

6.144.2.4 `u_int8_t scsi_low_msgout_data::md_msg`

Definition at line 3188 of file `scsi_low.c`.

Referenced by `scsi_low_msgout()`.

6.144.2.5 `int(* scsi_low_msgout_data::md_msgfunc)(struct scsi_low_softc *)`

Referenced by `scsi_low_msgout()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_low.c`

6.145 scsi_low_osdep_funcs Struct Reference

```
#include <scsi_low.h>
```

Data Fields

- [int\(* scsi_low_osdep_attach\)](#)(struct [scsi_low_softc](#) *)
- [int\(* scsi_low_osdep_world_start\)](#)(struct [scsi_low_softc](#) *)
- [int\(* scsi_low_osdep_dettach\)](#)(struct [scsi_low_softc](#) *)
- [int\(* scsi_low_osdep_ccb_setup\)](#)(struct [scsi_low_softc](#) *, struct [slccb](#) *)
- [int\(* scsi_low_osdep_done\)](#)(struct [scsi_low_softc](#) *, struct [slccb](#) *)
- [void\(* scsi_low_osdep_timeout\)](#)(struct [scsi_low_softc](#) *, int, int)

6.145.1 Detailed Description

Definition at line 148 of file [scsi_low.h](#).

6.145.2 Field Documentation

6.145.2.1 [int\(* scsi_low_osdep_funcs::scsi_low_osdep_attach\)](#)(struct [scsi_low_softc](#) *)

6.145.2.2 [int\(* scsi_low_osdep_funcs::scsi_low_osdep_ccb_setup\)](#)(struct [scsi_low_softc](#) *, struct [slccb](#) *)

6.145.2.3 [int\(* scsi_low_osdep_funcs::scsi_low_osdep_dettach\)](#)(struct [scsi_low_softc](#) *)

6.145.2.4 [int\(* scsi_low_osdep_funcs::scsi_low_osdep_done\)](#)(struct [scsi_low_softc](#) *, struct [slccb](#) *)

6.145.2.5 [void\(* scsi_low_osdep_funcs::scsi_low_osdep_timeout\)](#)(struct [scsi_low_softc](#) *, int, int)

Referenced by [scsi_low_engage\(\)](#), and [scsi_low_timeout\(\)](#).

6.145.2.6 [int\(* scsi_low_osdep_funcs::scsi_low_osdep_world_start\)](#)(struct [scsi_low_softc](#) *)

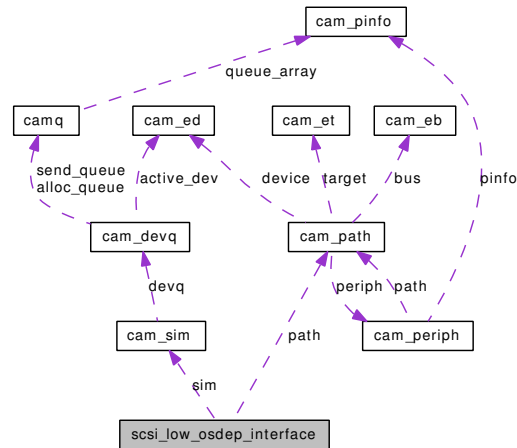
The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_low.h](#)

6.146 scsi_low_osdep_interface Struct Reference

```
#include <scsi_low.h>
```

Collaboration diagram for scsi_low_osdep_interface:



Data Fields

- `DEVPOR_DEVICE` [si_dev](#)
- `cam_sim` * [sim](#)
- `cam_path` * [path](#)
- `int` [si_poll_count](#)
- `callout_handle` [engage_ch](#)
- `callout_handle` [timeout_ch](#)

6.146.1 Detailed Description

Definition at line 117 of file `scsi_low.h`.

6.146.2 Field Documentation

6.146.2.1 struct `callout_handle` [scsi_low_osdep_interface::engage_ch](#)

Definition at line 125 of file `scsi_low.h`.

6.146.2.2 struct `cam_path*` [scsi_low_osdep_interface::path](#)

Definition at line 121 of file `scsi_low.h`.

6.146.2.3 `DEVPOR_DEVICE` [scsi_low_osdep_interface::si_dev](#)

Definition at line 118 of file `scsi_low.h`.

6.146.2.4 int [scsi_low_osdep_interface::si_poll_count](#)

Definition at line 123 of file [scsi_low.h](#).

6.146.2.5 struct [cam_sim*](#) [scsi_low_osdep_interface::sim](#)

Definition at line 120 of file [scsi_low.h](#).

6.146.2.6 struct [callout_handle](#) [scsi_low_osdep_interface::timeout_ch](#)

Definition at line 126 of file [scsi_low.h](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_low.h](#)

6.147 scsi_low_osdep_lun_interface Struct Reference

```
#include <scsi_low.h>
```

6.147.1 Detailed Description

Definition at line 135 of file `scsi_low.h`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_low.h](#)

6.148 `scsi_low_osdep_targ_interface` Struct Reference

```
#include <scsi_low.h>
```

6.148.1 Detailed Description

Definition at line 132 of file `scsi_low.h`.

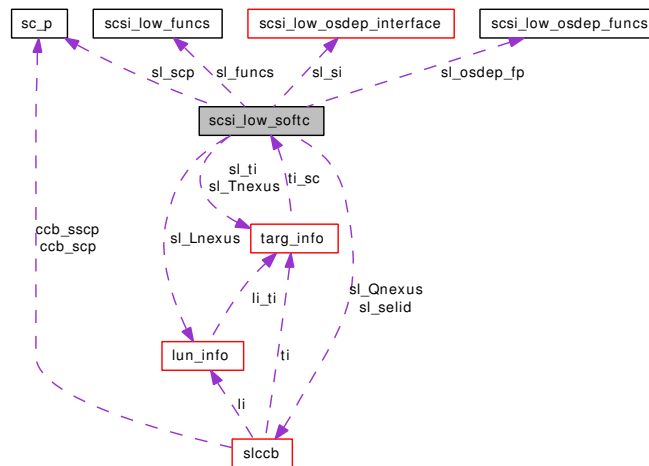
The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_low.h](#)

6.149 scsi_low_softc Struct Reference

```
#include <scsi_low.h>
```

Collaboration diagram for scsi_low_softc:



Public Member Functions

- `LIST_ENTRY (scsi_low_softc) sl_chain`

Data Fields

- `scsi_low_osdep_interface sl_si`
- `scsi_low_osdep_funcs * sl_osdep_fp`
- `u_char sl_xname [16]`
- `targ_info * sl_ti [SCSI_LOW_NTARGETS]`
- `targ_info_tab sl_titab`
- `targ_info * sl_Tnexus`
- `lun_info * sl_Lnexus`
- `slccb * sl_Qnexus`
- `int sl_nexus_call`
- `slccbtabs sl_start`
- `int sl_max_retry`
- `int sl_ph_count`
- `int sl_timeout_count`
- `int sl_nio`
- `int sl_disc`
- `int sl_retry_sel`
- `slccb * sl_selid`
- `int sl_atten`
- `int sl_clear_atten`
- `u_int sl_msgphase`
- `u_int sl_error`
- `sc_p sl_scp`

- `u_int sl_active`
- `int sl_powc`
- `u_int sl_rstep`
- `u_int sl_flags`
- `u_int sl_cfgflags`
- `int sl_show_result`
- `u_int sl_hostid`
- `int sl_nluns`
- `int sl_ntargs`
- `int sl_openings`
- `scsi_low_funcs * sl_funcs`
- `int sl_targsize`

6.149.1 Detailed Description

Definition at line 545 of file `scsi_low.h`.

6.149.2 Member Function Documentation

6.149.2.1 `scsi_low_softc::LIST_ENTRY` (`scsi_low_softc`)

6.149.3 Field Documentation

6.149.3.1 `u_int scsi_low_softc::sl_active`

Definition at line 615 of file `scsi_low.h`.

6.149.3.2 `int scsi_low_softc::sl_atten`

Definition at line 580 of file `scsi_low.h`.

6.149.3.3 `u_int scsi_low_softc::sl_cfgflags`

Definition at line 630 of file `scsi_low.h`.

Referenced by `scsi_low_activate_pisa()`, `scsi_low_calcf_lun()`, and `scsi_low_calcf_target()`.

6.149.3.4 `int scsi_low_softc::sl_clear_atten`

Definition at line 581 of file `scsi_low.h`.

6.149.3.5 `int scsi_low_softc::sl_disc`

Definition at line 575 of file `scsi_low.h`.

Referenced by `scsi_low_establish_ccb()`.

6.149.3.6 `u_int scsi_low_softc::sl_error`

Definition at line 594 of file `scsi_low.h`.

Referenced by `scsi_low_establish_ccb()`.

6.149.3.7 `u_int scsi_low_softc::sl_flags`

Definition at line 620 of file `scsi_low.h`.

Referenced by `scsi_low_engage()`.

6.149.3.8 `struct scsi_low_funcs* scsi_low_softc::sl_funcs`

Definition at line 651 of file `scsi_low.h`.

Referenced by `scsi_low_alloc_li()`, `scsi_low_engage()`, and `scsi_low_establish_ccb()`.

6.149.3.9 `u_int scsi_low_softc::sl_hostid`

Definition at line 645 of file `scsi_low.h`.

6.149.3.10 `struct lun_info* scsi_low_softc::sl_Lnexus`

Definition at line 561 of file `scsi_low.h`.

6.149.3.11 `int scsi_low_softc::sl_max_retry`

Definition at line 569 of file `scsi_low.h`.

6.149.3.12 `u_int scsi_low_softc::sl_msgphase`

Definition at line 584 of file `scsi_low.h`.

6.149.3.13 `int scsi_low_softc::sl_nexus_call`

Definition at line 563 of file `scsi_low.h`.

6.149.3.14 `int scsi_low_softc::sl_nio`

Definition at line 574 of file `scsi_low.h`.

6.149.3.15 `int scsi_low_softc::sl_nluns`

Definition at line 646 of file `scsi_low.h`.

6.149.3.16 `int scsi_low_softc::sl_ntargs`

Definition at line 647 of file `scsi_low.h`.

6.149.3.17 `int scsi_low_softc::sl_openings`

Definition at line 648 of file `scsi_low.h`.

6.149.3.18 `struct scsi_low_osdep_funcs* scsi_low_softc::sl_osdep_fp`

Definition at line 549 of file `scsi_low.h`.

Referenced by `scsi_low_engage()`, and `scsi_low_timeout()`.

6.149.3.19 `int scsi_low_softc::sl_ph_count`

Definition at line 570 of file `scsi_low.h`.

6.149.3.20 `int scsi_low_softc::sl_powc`

Definition at line 616 of file `scsi_low.h`.

6.149.3.21 `struct slccb* scsi_low_softc::sl_Qnexus`

Definition at line 562 of file `scsi_low.h`.

Referenced by `scsi_low_establish_ccb()`.

6.149.3.22 `int scsi_low_softc::sl_retry_sel`

Definition at line 576 of file `scsi_low.h`.

6.149.3.23 `u_int scsi_low_softc::sl_rstep`

Definition at line 617 of file `scsi_low.h`.

Referenced by `scsi_low_engage()`.

6.149.3.24 `struct sc_p scsi_low_softc::sl_scp`

Definition at line 612 of file `scsi_low.h`.

Referenced by `scsi_low_establish_ccb()`.

6.149.3.25 `struct slccb* scsi_low_softc::sl_selid`

Definition at line 577 of file `scsi_low.h`.

6.149.3.26 `int scsi_low_softc::sl_show_result`

Definition at line 637 of file `scsi_low.h`.

6.149.3.27 `struct scsi_low_osdep_interface scsi_low_softc::sl_si`

Definition at line 547 of file `scsi_low.h`.

6.149.3.28 `struct slccbtab scsi_low_softc::sl_start`

Definition at line 566 of file `scsi_low.h`.

6.149.3.29 `int scsi_low_softc::sl_targsize`

Definition at line 654 of file `scsi_low.h`.

6.149.3.30 `struct targ_info* scsi_low_softc::sl_ti[SCSI_LOW_NTARGETS]`

Definition at line 556 of file `scsi_low.h`.

Referenced by `scsi_low_alloc_ti()`.

6.149.3.31 `int scsi_low_softc::sl_timeout_count`

Definition at line 571 of file `scsi_low.h`.

6.149.3.32 `struct targ_info_tab scsi_low_softc::sl_titab`

Definition at line 557 of file `scsi_low.h`.

6.149.3.33 `struct targ_info* scsi_low_softc::sl_Tnexus`

Definition at line 560 of file `scsi_low.h`.

6.149.3.34 `u_char scsi_low_softc::sl_xname[16]`

Definition at line 550 of file `scsi_low.h`.

Referenced by `scsi_low_calcf_show()`, `scsi_low_calcf_target()`, and `scsi_low_establish_ccb()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_low.h`

6.150 scsi_low_statics Struct Reference

Data Fields

- int [nexus_win](#)
- int [nexus_fail](#)
- int [nexus_disconnected](#)
- int [nexus_reselected](#)
- int [nexus_conflict](#)

6.150.1 Detailed Description

Definition at line 200 of file [scsi_low.c](#).

6.150.2 Field Documentation

6.150.2.1 int [scsi_low_statics::nexus_conflict](#)

Definition at line 205 of file [scsi_low.c](#).

6.150.2.2 int [scsi_low_statics::nexus_disconnected](#)

Definition at line 203 of file [scsi_low.c](#).

6.150.2.3 int [scsi_low_statics::nexus_fail](#)

Definition at line 202 of file [scsi_low.c](#).

6.150.2.4 int [scsi_low_statics::nexus_reselected](#)

Definition at line 204 of file [scsi_low.c](#).

6.150.2.5 int [scsi_low_statics::nexus_win](#)

Definition at line 201 of file [scsi_low.c](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_low.c](#)

6.151 scsi_mode_blk_desc Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- [u_int8_t density](#)
- [u_int8_t nblocks](#) [3]
- [u_int8_t reserved](#)
- [u_int8_t blklen](#) [3]

6.151.1 Detailed Description

Definition at line 819 of file `scsi_all.h`.

6.151.2 Field Documentation

6.151.2.1 [u_int8_t scsi_mode_blk_desc::blklen](#)[3]

Definition at line 824 of file `scsi_all.h`.

6.151.2.2 [u_int8_t scsi_mode_blk_desc::density](#)

Definition at line 821 of file `scsi_all.h`.

6.151.2.3 [u_int8_t scsi_mode_blk_desc::nblocks](#)[3]

Definition at line 822 of file `scsi_all.h`.

6.151.2.4 [u_int8_t scsi_mode_blk_desc::reserved](#)

Definition at line 823 of file `scsi_all.h`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_all.h](#)

6.152 `scsi_mode_block_descr` Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- `u_int8_t density_code`
- `u_int8_t num_blocks` [3]
- `u_int8_t reserved`
- `u_int8_t block_len` [3]

6.152.1 Detailed Description

Definition at line 236 of file `scsi_all.h`.

6.152.2 Field Documentation

6.152.2.1 `u_int8_t scsi_mode_block_descr::block_len`[3]

Definition at line 241 of file `scsi_all.h`.

6.152.2.2 `u_int8_t scsi_mode_block_descr::density_code`

Definition at line 238 of file `scsi_all.h`.

6.152.2.3 `u_int8_t scsi_mode_block_descr::num_blocks`[3]

Definition at line 239 of file `scsi_all.h`.

6.152.2.4 `u_int8_t scsi_mode_block_descr::reserved`

Definition at line 240 of file `scsi_all.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.153 scsi_mode_hdr_10 Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- [u_int8_t data_len](#) [2]
- [u_int8_t medium_type](#)
- [u_int8_t dev_specific](#)
- [u_int8_t reserved](#) [2]
- [u_int8_t block_descr_len](#) [2]

6.153.1 Detailed Description

Definition at line 227 of file `scsi_all.h`.

6.153.2 Field Documentation

6.153.2.1 [u_int8_t scsi_mode_hdr_10::block_descr_len](#)[2]

Definition at line 233 of file `scsi_all.h`.

6.153.2.2 [u_int8_t scsi_mode_hdr_10::data_len](#)[2]

Definition at line 229 of file `scsi_all.h`.

6.153.2.3 [u_int8_t scsi_mode_hdr_10::dev_specific](#)

Definition at line 231 of file `scsi_all.h`.

6.153.2.4 [u_int8_t scsi_mode_hdr_10::medium_type](#)

Definition at line 230 of file `scsi_all.h`.

6.153.2.5 [u_int8_t scsi_mode_hdr_10::reserved](#)[2]

Definition at line 232 of file `scsi_all.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.154 `scsi_mode_hdr_6` Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- `u_int8_t data_len`
- `u_int8_t medium_type`
- `u_int8_t dev_specific`
- `u_int8_t block_descr_len`

6.154.1 Detailed Description

Definition at line 219 of file `scsi_all.h`.

6.154.2 Field Documentation

6.154.2.1 `u_int8_t scsi_mode_hdr_6::block_descr_len`

Definition at line 224 of file `scsi_all.h`.

6.154.2.2 `u_int8_t scsi_mode_hdr_6::data_len`

Definition at line 221 of file `scsi_all.h`.

6.154.2.3 `u_int8_t scsi_mode_hdr_6::dev_specific`

Definition at line 223 of file `scsi_all.h`.

6.154.2.4 `u_int8_t scsi_mode_hdr_6::medium_type`

Definition at line 222 of file `scsi_all.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.155 scsi_mode_header_10 Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- [u_int8_t data_length](#) [2]
- [u_int8_t medium_type](#)
- [u_int8_t dev_spec](#)
- [u_int8_t unused](#) [2]
- [u_int8_t blk_desc_len](#) [2]

6.155.1 Detailed Description

Definition at line 804 of file `scsi_all.h`.

6.155.2 Field Documentation

6.155.2.1 [u_int8_t scsi_mode_header_10::blk_desc_len](#)[2]

Definition at line 810 of file `scsi_all.h`.

Referenced by `find_mode_page_10()`.

6.155.2.2 [u_int8_t scsi_mode_header_10::data_length](#)[2]

Definition at line 806 of file `scsi_all.h`.

Referenced by `cdgetmode()`, and `cdsetmode()`.

6.155.2.3 [u_int8_t scsi_mode_header_10::dev_spec](#)

Definition at line 808 of file `scsi_all.h`.

6.155.2.4 [u_int8_t scsi_mode_header_10::medium_type](#)

Definition at line 807 of file `scsi_all.h`.

Referenced by `cdsetmode()`.

6.155.2.5 [u_int8_t scsi_mode_header_10::unused](#)[2]

Definition at line 809 of file `scsi_all.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.156 `scsi_mode_header_6` Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- `u_int8_t data_length`
- `u_int8_t medium_type`
- `u_int8_t dev_spec`
- `u_int8_t blk_desc_len`

6.156.1 Detailed Description

Definition at line 796 of file `scsi_all.h`.

6.156.2 Field Documentation

6.156.2.1 `u_int8_t scsi_mode_header_6::blk_desc_len`

Definition at line 801 of file `scsi_all.h`.

Referenced by `cd6byteworkaround()`, `find_mode_page_6()`, `probedone()`, `sagetparams()`, and `sasetparams()`.

6.156.2.2 `u_int8_t scsi_mode_header_6::data_length`

Definition at line 798 of file `scsi_all.h`.

Referenced by `cdgetmode()`, `cdsetmode()`, and `sasetparams()`.

6.156.2.3 `u_int8_t scsi_mode_header_6::dev_spec`

Definition at line 800 of file `scsi_all.h`.

Referenced by `sagetparams()`, and `sasetparams()`.

6.156.2.4 `u_int8_t scsi_mode_header_6::medium_type`

Definition at line 799 of file `scsi_all.h`.

Referenced by `cdsetmode()`, and `sasetparams()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.157 scsi_mode_page_header Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- [u_int8_t page_code](#)
- [u_int8_t page_length](#)

6.157.1 Detailed Description

Definition at line 813 of file [scsi_all.h](#).

6.157.2 Field Documentation

6.157.2.1 [u_int8_t scsi_mode_page_header::page_code](#)

Definition at line 815 of file [scsi_all.h](#).

Referenced by [cd6byteworkaround\(\)](#).

6.157.2.2 [u_int8_t scsi_mode_page_header::page_length](#)

Definition at line 816 of file [scsi_all.h](#).

Referenced by [cd6byteworkaround\(\)](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_all.h](#)

6.158 `scsi_mode_select_10` Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- `u_int8_t opcode`
- `u_int8_t byte2`
- `u_int8_t unused` [5]
- `u_int8_t length` [2]
- `u_int8_t control`

6.158.1 Detailed Description

Definition at line 207 of file `scsi_all.h`.

6.158.2 Field Documentation

6.158.2.1 `u_int8_t scsi_mode_select_10::byte2`

Definition at line 210 of file `scsi_all.h`.

6.158.2.2 `u_int8_t scsi_mode_select_10::control`

Definition at line 213 of file `scsi_all.h`.

6.158.2.3 `u_int8_t scsi_mode_select_10::length`[2]

Definition at line 212 of file `scsi_all.h`.

6.158.2.4 `u_int8_t scsi_mode_select_10::opcode`

Definition at line 209 of file `scsi_all.h`.

6.158.2.5 `u_int8_t scsi_mode_select_10::unused`[5]

Definition at line 211 of file `scsi_all.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.159 scsi_mode_select_6 Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t byte2](#)
- [u_int8_t unused](#) [2]
- [u_int8_t length](#)
- [u_int8_t control](#)

6.159.1 Detailed Description

Definition at line 196 of file `scsi_all.h`.

6.159.2 Field Documentation

6.159.2.1 [u_int8_t scsi_mode_select_6::byte2](#)

Definition at line 199 of file `scsi_all.h`.

Referenced by `cd6byteworkaround()`.

6.159.2.2 [u_int8_t scsi_mode_select_6::control](#)

Definition at line 204 of file `scsi_all.h`.

Referenced by `cd6byteworkaround()`.

6.159.2.3 [u_int8_t scsi_mode_select_6::length](#)

Definition at line 203 of file `scsi_all.h`.

6.159.2.4 [u_int8_t scsi_mode_select_6::opcode](#)

Definition at line 198 of file `scsi_all.h`.

6.159.2.5 [u_int8_t scsi_mode_select_6::unused](#)[2]

Definition at line 202 of file `scsi_all.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.160 `scsi_mode_sense_10` Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- `u_int8_t opcode`
- `u_int8_t byte2`
- `u_int8_t page`
- `u_int8_t unused` [4]
- `u_int8_t length` [2]
- `u_int8_t control`

6.160.1 Detailed Description

Definition at line 186 of file `scsi_all.h`.

6.160.2 Field Documentation

6.160.2.1 `u_int8_t scsi_mode_sense_10::byte2`

Definition at line 189 of file `scsi_all.h`.

6.160.2.2 `u_int8_t scsi_mode_sense_10::control`

Definition at line 193 of file `scsi_all.h`.

6.160.2.3 `u_int8_t scsi_mode_sense_10::length`[2]

Definition at line 192 of file `scsi_all.h`.

6.160.2.4 `u_int8_t scsi_mode_sense_10::opcode`

Definition at line 188 of file `scsi_all.h`.

6.160.2.5 `u_int8_t scsi_mode_sense_10::page`

Definition at line 190 of file `scsi_all.h`.

6.160.2.6 `u_int8_t scsi_mode_sense_10::unused`[4]

Definition at line 191 of file `scsi_all.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.161 scsi_mode_sense_6 Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t byte2](#)
- [u_int8_t page](#)
- [u_int8_t unused](#)
- [u_int8_t length](#)
- [u_int8_t control](#)

6.161.1 Detailed Description

Definition at line 164 of file `scsi_all.h`.

6.161.2 Field Documentation

6.161.2.1 [u_int8_t scsi_mode_sense_6::byte2](#)

Definition at line 167 of file `scsi_all.h`.

Referenced by `cd6byteworkaround()`, `chdone()`, and `chgetparams()`.

6.161.2.2 [u_int8_t scsi_mode_sense_6::control](#)

Definition at line 183 of file `scsi_all.h`.

Referenced by `cd6byteworkaround()`.

6.161.2.3 [u_int8_t scsi_mode_sense_6::length](#)

Definition at line 182 of file `scsi_all.h`.

6.161.2.4 [u_int8_t scsi_mode_sense_6::opcode](#)

Definition at line 166 of file `scsi_all.h`.

6.161.2.5 [u_int8_t scsi_mode_sense_6::page](#)

Definition at line 169 of file `scsi_all.h`.

Referenced by `cd6byteworkaround()`.

6.161.2.6 `u_int8_t scsi_mode_sense_6::unused`

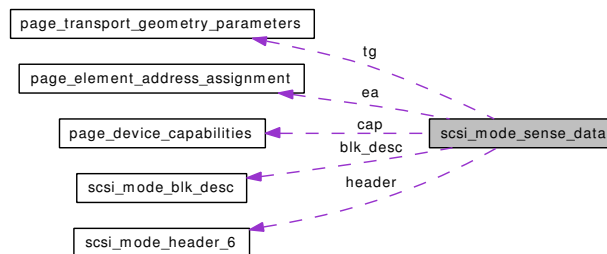
Definition at line 181 of file `scsi_all.h`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_all.h](#)

6.162 scsi_mode_sense_data Struct Reference

Collaboration diagram for `scsi_mode_sense_data`:



Data Fields

- [scsi_mode_header_6](#) header
- [scsi_mode_blk_desc](#) blk_desc
- union {
 - [page_element_address_assignment](#) ea
 - [page_transport_geometry_parameters](#) tg
 - [page_device_capabilities](#) cap
 } pages

6.162.1 Detailed Description

Definition at line 132 of file `scsi_ch.c`.

6.162.2 Field Documentation

6.162.2.1 struct [scsi_mode_blk_desc](#) `scsi_mode_sense_data::blk_desc`

Definition at line 134 of file `scsi_ch.c`.

6.162.2.2 struct [page_device_capabilities](#) `scsi_mode_sense_data::cap`

Definition at line 138 of file `scsi_ch.c`.

6.162.2.3 struct [page_element_address_assignment](#) `scsi_mode_sense_data::ea`

Definition at line 136 of file `scsi_ch.c`.

6.162.2.4 struct [scsi_mode_header_6](#) `scsi_mode_sense_data::header`

Definition at line 133 of file `scsi_ch.c`.

6.162.2.5 `union { ... } scsi_mode_sense_data::pages`

6.162.2.6 `struct page_transport_geometry_parameters scsi_mode_sense_data::tg`

Definition at line 137 of file `scsi_ch.c`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_ch.c`

6.163 scsi_move_medium Struct Reference

```
#include <scsi_ch.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t byte2](#)
- [u_int8_t tea](#) [2]
- [u_int8_t src](#) [2]
- [u_int8_t dst](#) [2]
- [u_int8_t reserved](#) [2]
- [u_int8_t invert](#)
- [u_int8_t control](#)

6.163.1 Detailed Description

Definition at line 106 of file `scsi_ch.h`.

6.163.2 Field Documentation

6.163.2.1 [u_int8_t scsi_move_medium::byte2](#)

Definition at line 109 of file `scsi_ch.h`.

6.163.2.2 [u_int8_t scsi_move_medium::control](#)

Definition at line 116 of file `scsi_ch.h`.

6.163.2.3 [u_int8_t scsi_move_medium::dst](#)[2]

Definition at line 112 of file `scsi_ch.h`.

6.163.2.4 [u_int8_t scsi_move_medium::invert](#)

Definition at line 114 of file `scsi_ch.h`.

6.163.2.5 [u_int8_t scsi_move_medium::opcode](#)

Definition at line 107 of file `scsi_ch.h`.

6.163.2.6 [u_int8_t scsi_move_medium::reserved](#)[2]

Definition at line 113 of file `scsi_ch.h`.

6.163.2.7 `u_int8_t scsi_move_medium::src`[2]

Definition at line 111 of file `scsi_ch.h`.

6.163.2.8 `u_int8_t scsi_move_medium::tea`[2]

Definition at line 110 of file `scsi_ch.h`.

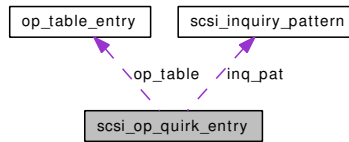
The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_ch.h](#)

6.164 scsi_op_quirk_entry Struct Reference

```
#include <scsi_all.h>
```

Collaboration diagram for `scsi_op_quirk_entry`:



Data Fields

- `scsi_inquiry_pattern inq_pat`
- `int num_ops`
- `op_table_entry * op_table`

6.164.1 Detailed Description

Definition at line 891 of file `scsi_all.h`.

6.164.2 Field Documentation

6.164.2.1 struct `scsi_inquiry_pattern scsi_op_quirk_entry::inq_pat`

Definition at line 892 of file `scsi_all.h`.

6.164.2.2 int `scsi_op_quirk_entry::num_ops`

Definition at line 893 of file `scsi_all.h`.

6.164.2.3 struct `op_table_entry* scsi_op_quirk_entry::op_table`

Definition at line 894 of file `scsi_all.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.165 `scsi_pause` Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- `u_int8_t op_code`
- `u_int8_t byte2`
- `u_int8_t unused` [6]
- `u_int8_t resume`
- `u_int8_t control`

6.165.1 Detailed Description

Definition at line 59 of file `scsi_cd.h`.

6.165.2 Field Documentation

6.165.2.1 `u_int8_t scsi_pause::byte2`

Definition at line 62 of file `scsi_cd.h`.

6.165.2.2 `u_int8_t scsi_pause::control`

Definition at line 65 of file `scsi_cd.h`.

6.165.2.3 `u_int8_t scsi_pause::op_code`

Definition at line 61 of file `scsi_cd.h`.

6.165.2.4 `u_int8_t scsi_pause::resume`

Definition at line 64 of file `scsi_cd.h`.

6.165.2.5 `u_int8_t scsi_pause::unused[6]`

Definition at line 63 of file `scsi_cd.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.h`

6.166 scsi_play_10 Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- [u_int8_t op_code](#)
- [u_int8_t byte2](#)
- [u_int8_t blk_addr](#) [4]
- [u_int8_t unused](#)
- [u_int8_t xfer_len](#) [2]
- [u_int8_t control](#)

6.166.1 Detailed Description

Definition at line 97 of file `scsi_cd.h`.

6.166.2 Field Documentation

6.166.2.1 [u_int8_t scsi_play_10::blk_addr](#)[4]

Definition at line 101 of file `scsi_cd.h`.

6.166.2.2 [u_int8_t scsi_play_10::byte2](#)

Definition at line 100 of file `scsi_cd.h`.

6.166.2.3 [u_int8_t scsi_play_10::control](#)

Definition at line 104 of file `scsi_cd.h`.

6.166.2.4 [u_int8_t scsi_play_10::op_code](#)

Definition at line 99 of file `scsi_cd.h`.

6.166.2.5 [u_int8_t scsi_play_10::unused](#)

Definition at line 102 of file `scsi_cd.h`.

6.166.2.6 [u_int8_t scsi_play_10::xfer_len](#)[2]

Definition at line 103 of file `scsi_cd.h`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_cd.h](#)

6.167 scsi_play_12 Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- [u_int8_t op_code](#)
- [u_int8_t byte2](#)
- [u_int8_t blk_addr](#) [4]
- [u_int8_t xfer_len](#) [4]
- [u_int8_t unused](#)
- [u_int8_t control](#)

6.167.1 Detailed Description

Definition at line 107 of file [scsi_cd.h](#).

6.167.2 Field Documentation

6.167.2.1 [u_int8_t scsi_play_12::blk_addr](#)[4]

Definition at line 111 of file [scsi_cd.h](#).

6.167.2.2 [u_int8_t scsi_play_12::byte2](#)

Definition at line 110 of file [scsi_cd.h](#).

6.167.2.3 [u_int8_t scsi_play_12::control](#)

Definition at line 114 of file [scsi_cd.h](#).

6.167.2.4 [u_int8_t scsi_play_12::op_code](#)

Definition at line 109 of file [scsi_cd.h](#).

6.167.2.5 [u_int8_t scsi_play_12::unused](#)

Definition at line 113 of file [scsi_cd.h](#).

6.167.2.6 [u_int8_t scsi_play_12::xfer_len](#)[4]

Definition at line 112 of file [scsi_cd.h](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_cd.h](#)

6.168 scsi_play_msf Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- [u_int8_t op_code](#)
- [u_int8_t byte2](#)
- [u_int8_t unused](#)
- [u_int8_t start_m](#)
- [u_int8_t start_s](#)
- [u_int8_t start_f](#)
- [u_int8_t end_m](#)
- [u_int8_t end_s](#)
- [u_int8_t end_f](#)
- [u_int8_t control](#)

6.168.1 Detailed Description

Definition at line 70 of file `scsi_cd.h`.

6.168.2 Field Documentation

6.168.2.1 [u_int8_t scsi_play_msf::byte2](#)

Definition at line 73 of file `scsi_cd.h`.

6.168.2.2 [u_int8_t scsi_play_msf::control](#)

Definition at line 81 of file `scsi_cd.h`.

6.168.2.3 [u_int8_t scsi_play_msf::end_f](#)

Definition at line 80 of file `scsi_cd.h`.

6.168.2.4 [u_int8_t scsi_play_msf::end_m](#)

Definition at line 78 of file `scsi_cd.h`.

6.168.2.5 [u_int8_t scsi_play_msf::end_s](#)

Definition at line 79 of file `scsi_cd.h`.

6.168.2.6 [u_int8_t scsi_play_msf::op_code](#)

Definition at line 72 of file `scsi_cd.h`.

6.168.2.7 `u_int8_t scsi_play_msf::start_f`

Definition at line 77 of file `scsi_cd.h`.

6.168.2.8 `u_int8_t scsi_play_msf::start_m`

Definition at line 75 of file `scsi_cd.h`.

6.168.2.9 `u_int8_t scsi_play_msf::start_s`

Definition at line 76 of file `scsi_cd.h`.

6.168.2.10 `u_int8_t scsi_play_msf::unused`

Definition at line 74 of file `scsi_cd.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.h`

6.169 scsi_play_rel_12 Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- [u_int8_t op_code](#)
- [u_int8_t byte2](#)
- [u_int8_t blk_addr](#) [4]
- [u_int8_t xfer_len](#) [4]
- [u_int8_t track](#)
- [u_int8_t control](#)

6.169.1 Detailed Description

Definition at line 117 of file [scsi_cd.h](#).

6.169.2 Field Documentation

6.169.2.1 [u_int8_t scsi_play_rel_12::blk_addr](#)[4]

Definition at line 121 of file [scsi_cd.h](#).

6.169.2.2 [u_int8_t scsi_play_rel_12::byte2](#)

Definition at line 120 of file [scsi_cd.h](#).

6.169.2.3 [u_int8_t scsi_play_rel_12::control](#)

Definition at line 124 of file [scsi_cd.h](#).

6.169.2.4 [u_int8_t scsi_play_rel_12::op_code](#)

Definition at line 119 of file [scsi_cd.h](#).

6.169.2.5 [u_int8_t scsi_play_rel_12::track](#)

Definition at line 123 of file [scsi_cd.h](#).

6.169.2.6 [u_int8_t scsi_play_rel_12::xfer_len](#)[4]

Definition at line 122 of file [scsi_cd.h](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_cd.h](#)

6.170 `scsi_play_track` Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- `u_int8_t op_code`
- `u_int8_t byte2`
- `u_int8_t unused` [2]
- `u_int8_t start_track`
- `u_int8_t start_index`
- `u_int8_t unused1`
- `u_int8_t end_track`
- `u_int8_t end_index`
- `u_int8_t control`

6.170.1 Detailed Description

Definition at line 84 of file `scsi_cd.h`.

6.170.2 Field Documentation

6.170.2.1 `u_int8_t scsi_play_track::byte2`

Definition at line 87 of file `scsi_cd.h`.

6.170.2.2 `u_int8_t scsi_play_track::control`

Definition at line 94 of file `scsi_cd.h`.

6.170.2.3 `u_int8_t scsi_play_track::end_index`

Definition at line 93 of file `scsi_cd.h`.

6.170.2.4 `u_int8_t scsi_play_track::end_track`

Definition at line 92 of file `scsi_cd.h`.

6.170.2.5 `u_int8_t scsi_play_track::op_code`

Definition at line 86 of file `scsi_cd.h`.

6.170.2.6 `u_int8_t scsi_play_track::start_index`

Definition at line 90 of file `scsi_cd.h`.

6.170.2.7 `u_int8_t scsi_play_track::start_track`

Definition at line 89 of file `scsi_cd.h`.

6.170.2.8 `u_int8_t scsi_play_track::unused[2]`

Definition at line 88 of file `scsi_cd.h`.

6.170.2.9 `u_int8_t scsi_play_track::unused1`

Definition at line 91 of file `scsi_cd.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.h`

6.171 `scsi_position_to_element` Struct Reference

```
#include <scsi_ch.h>
```

Data Fields

- `u_int8_t opcode`
- `u_int8_t byte2`
- `u_int8_t tea` [2]
- `u_int8_t dst` [2]
- `u_int8_t reserved` [2]
- `u_int8_t invert`
- `u_int8_t control`

6.171.1 Detailed Description

Definition at line 123 of file `scsi_ch.h`.

6.171.2 Field Documentation

6.171.2.1 `u_int8_t scsi_position_to_element::byte2`

Definition at line 126 of file `scsi_ch.h`.

6.171.2.2 `u_int8_t scsi_position_to_element::control`

Definition at line 132 of file `scsi_ch.h`.

6.171.2.3 `u_int8_t scsi_position_to_element::dst`[2]

Definition at line 128 of file `scsi_ch.h`.

6.171.2.4 `u_int8_t scsi_position_to_element::invert`

Definition at line 130 of file `scsi_ch.h`.

6.171.2.5 `u_int8_t scsi_position_to_element::opcode`

Definition at line 124 of file `scsi_ch.h`.

6.171.2.6 `u_int8_t scsi_position_to_element::reserved`[2]

Definition at line 129 of file `scsi_ch.h`.

6.171.2.7 `u_int8_t scsi_position_to_element::tea[2]`

Definition at line 127 of file `scsi_ch.h`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_ch.h](#)

6.172 `scsi_prevent` Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- `u_int8_t opcode`
- `u_int8_t byte2`
- `u_int8_t unused` [2]
- `u_int8_t how`
- `u_int8_t control`

6.172.1 Detailed Description

Definition at line 350 of file `scsi_all.h`.

6.172.2 Field Documentation

6.172.2.1 `u_int8_t scsi_prevent::byte2`

Definition at line 353 of file `scsi_all.h`.

6.172.2.2 `u_int8_t scsi_prevent::control`

Definition at line 356 of file `scsi_all.h`.

6.172.2.3 `u_int8_t scsi_prevent::how`

Definition at line 355 of file `scsi_all.h`.

6.172.2.4 `u_int8_t scsi_prevent::opcode`

Definition at line 352 of file `scsi_all.h`.

6.172.2.5 `u_int8_t scsi_prevent::unused`[2]

Definition at line 354 of file `scsi_all.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.173 scsi_read_block_limits Struct Reference

```
#include <scsi_sa.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t byte2](#)
- [u_int8_t unused](#) [3]
- [u_int8_t control](#)

6.173.1 Detailed Description

Definition at line 37 of file `scsi_sa.h`.

6.173.2 Field Documentation

6.173.2.1 [u_int8_t scsi_read_block_limits::byte2](#)

Definition at line 40 of file `scsi_sa.h`.

6.173.2.2 [u_int8_t scsi_read_block_limits::control](#)

Definition at line 42 of file `scsi_sa.h`.

6.173.2.3 [u_int8_t scsi_read_block_limits::opcode](#)

Definition at line 39 of file `scsi_sa.h`.

6.173.2.4 [u_int8_t scsi_read_block_limits::unused](#)[3]

Definition at line 41 of file `scsi_sa.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_sa.h`

6.174 `scsi_read_block_limits_data` Struct Reference

```
#include <scsi_sa.h>
```

Data Fields

- `u_int8_t gran`
- `u_int8_t maximum` [3]
- `u_int8_t minimum` [2]

6.174.1 Detailed Description

Definition at line 45 of file `scsi_sa.h`.

6.174.2 Field Documentation

6.174.2.1 `u_int8_t scsi_read_block_limits_data::gran`

Definition at line 47 of file `scsi_sa.h`.

6.174.2.2 `u_int8_t scsi_read_block_limits_data::maximum`[3]

Definition at line 50 of file `scsi_sa.h`.

Referenced by `samount()`.

6.174.2.3 `u_int8_t scsi_read_block_limits_data::minimum`[2]

Definition at line 51 of file `scsi_sa.h`.

Referenced by `samount()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_sa.h`

6.175 scsi_read_buffer Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t byte2](#)
- [u_int8_t buffer_id](#)
- [u_int8_t offset](#) [3]
- [u_int8_t length](#) [3]
- [u_int8_t control](#)

6.175.1 Detailed Description

Definition at line 383 of file [scsi_all.h](#).

6.175.2 Field Documentation

6.175.2.1 [u_int8_t scsi_read_buffer::buffer_id](#)

Definition at line 392 of file [scsi_all.h](#).

6.175.2.2 [u_int8_t scsi_read_buffer::byte2](#)

Definition at line 386 of file [scsi_all.h](#).

6.175.2.3 [u_int8_t scsi_read_buffer::control](#)

Definition at line 395 of file [scsi_all.h](#).

6.175.2.4 [u_int8_t scsi_read_buffer::length](#)[3]

Definition at line 394 of file [scsi_all.h](#).

6.175.2.5 [u_int8_t scsi_read_buffer::offset](#)[3]

Definition at line 393 of file [scsi_all.h](#).

6.175.2.6 [u_int8_t scsi_read_buffer::opcode](#)

Definition at line 385 of file [scsi_all.h](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_all.h](#)

6.176 `scsi_read_capacity` Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- `u_int8_t opcode`
- `u_int8_t byte2`
- `u_int8_t addr` [4]
- `u_int8_t unused` [3]
- `u_int8_t control`

6.176.1 Detailed Description

Definition at line 677 of file `scsi_all.h`.

6.176.2 Field Documentation

6.176.2.1 `u_int8_t scsi_read_capacity::addr`[4]

Definition at line 681 of file `scsi_all.h`.

6.176.2.2 `u_int8_t scsi_read_capacity::byte2`

Definition at line 680 of file `scsi_all.h`.

6.176.2.3 `u_int8_t scsi_read_capacity::control`

Definition at line 683 of file `scsi_all.h`.

6.176.2.4 `u_int8_t scsi_read_capacity::opcode`

Definition at line 679 of file `scsi_all.h`.

6.176.2.5 `u_int8_t scsi_read_capacity::unused`[3]

Definition at line 682 of file `scsi_all.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.177 scsi_read_capacity_16 Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- [uint8_t opcode](#)
- [uint8_t service_action](#)
- [uint8_t addr](#) [8]
- [uint8_t alloc_len](#) [4]
- [uint8_t reladr](#)
- [uint8_t control](#)

6.177.1 Detailed Description

Definition at line 686 of file `scsi_all.h`.

6.177.2 Field Documentation

6.177.2.1 [uint8_t scsi_read_capacity_16::addr](#)[8]

Definition at line 691 of file `scsi_all.h`.

6.177.2.2 [uint8_t scsi_read_capacity_16::alloc_len](#)[4]

Definition at line 692 of file `scsi_all.h`.

6.177.2.3 [uint8_t scsi_read_capacity_16::control](#)

Definition at line 696 of file `scsi_all.h`.

6.177.2.4 [uint8_t scsi_read_capacity_16::opcode](#)

Definition at line 688 of file `scsi_all.h`.

6.177.2.5 [uint8_t scsi_read_capacity_16::reladr](#)

Definition at line 695 of file `scsi_all.h`.

6.177.2.6 [uint8_t scsi_read_capacity_16::service_action](#)

Definition at line 690 of file `scsi_all.h`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_all.h](#)

6.178 `scsi_read_capacity_data` Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- `u_int8_t addr` [4]
- `u_int8_t length` [4]

6.178.1 Detailed Description

Definition at line 699 of file `scsi_all.h`.

6.178.2 Field Documentation

6.178.2.1 `u_int8_t scsi_read_capacity_data::addr`[4]

Definition at line 701 of file `scsi_all.h`.

Referenced by `cddone()`, `cdsize()`, `dadone()`, and `dagetcapacity()`.

6.178.2.2 `u_int8_t scsi_read_capacity_data::length`[4]

Definition at line 702 of file `scsi_all.h`.

Referenced by `cddone()`, `cdsize()`, `dadone()`, and `dagetcapacity()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.179 scsi_read_capacity_data_long Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- `uint8_t addr` [8]
- `uint8_t length` [4]

6.179.1 Detailed Description

Definition at line 705 of file `scsi_all.h`.

6.179.2 Field Documentation

6.179.2.1 `uint8_t scsi_read_capacity_data_long::addr`[8]

Definition at line 707 of file `scsi_all.h`.

Referenced by `dadone()`.

6.179.2.2 `uint8_t scsi_read_capacity_data_long::length`[4]

Definition at line 708 of file `scsi_all.h`.

Referenced by `dadone()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.180 scsi_read_cd_cap_data Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- [u_int8_t addr_3](#)
- [u_int8_t addr_2](#)
- [u_int8_t addr_1](#)
- [u_int8_t addr_0](#)
- [u_int8_t length_3](#)
- [u_int8_t length_2](#)
- [u_int8_t length_1](#)
- [u_int8_t length_0](#)

6.180.1 Detailed Description

Definition at line 646 of file scsi_cd.h.

6.180.2 Field Documentation

6.180.2.1 [u_int8_t scsi_read_cd_cap_data::addr_0](#)

Definition at line 651 of file scsi_cd.h.

6.180.2.2 [u_int8_t scsi_read_cd_cap_data::addr_1](#)

Definition at line 650 of file scsi_cd.h.

6.180.2.3 [u_int8_t scsi_read_cd_cap_data::addr_2](#)

Definition at line 649 of file scsi_cd.h.

6.180.2.4 [u_int8_t scsi_read_cd_cap_data::addr_3](#)

Definition at line 648 of file scsi_cd.h.

6.180.2.5 [u_int8_t scsi_read_cd_cap_data::length_0](#)

Definition at line 655 of file scsi_cd.h.

6.180.2.6 [u_int8_t scsi_read_cd_cap_data::length_1](#)

Definition at line 654 of file scsi_cd.h.

6.180.2.7 `u_int8_t scsi_read_cd_cap_data::length_2`

Definition at line 653 of file `scsi_cd.h`.

6.180.2.8 `u_int8_t scsi_read_cd_cap_data::length_3`

Definition at line 652 of file `scsi_cd.h`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_cd.h](#)

6.181 scsi_read_cd_capacity Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- [u_int8_t op_code](#)
- [u_int8_t byte2](#)
- [u_int8_t addr_3](#)
- [u_int8_t addr_2](#)
- [u_int8_t addr_1](#)
- [u_int8_t addr_0](#)
- [u_int8_t unused](#) [3]
- [u_int8_t control](#)

6.181.1 Detailed Description

Definition at line 160 of file `scsi_cd.h`.

6.181.2 Field Documentation

6.181.2.1 [u_int8_t scsi_read_cd_capacity::addr_0](#)

Definition at line 167 of file `scsi_cd.h`.

6.181.2.2 [u_int8_t scsi_read_cd_capacity::addr_1](#)

Definition at line 166 of file `scsi_cd.h`.

6.181.2.3 [u_int8_t scsi_read_cd_capacity::addr_2](#)

Definition at line 165 of file `scsi_cd.h`.

6.181.2.4 [u_int8_t scsi_read_cd_capacity::addr_3](#)

Definition at line 164 of file `scsi_cd.h`.

6.181.2.5 [u_int8_t scsi_read_cd_capacity::byte2](#)

Definition at line 163 of file `scsi_cd.h`.

6.181.2.6 [u_int8_t scsi_read_cd_capacity::control](#)

Definition at line 169 of file `scsi_cd.h`.

6.181.2.7 `u_int8_t scsi_read_cd_capacity::op_code`

Definition at line 162 of file `scsi_cd.h`.

6.181.2.8 `u_int8_t scsi_read_cd_capacity::unused[3]`

Definition at line 168 of file `scsi_cd.h`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_cd.h](#)

6.182 scsi_read_defect_data_10 Struct Reference

```
#include <scsi_da.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t byte2](#)
- [u_int8_t format](#)
- [u_int8_t reserved](#) [4]
- [u_int8_t alloc_length](#) [2]
- [u_int8_t control](#)

6.182.1 Detailed Description

Definition at line 93 of file `scsi_da.h`.

6.182.2 Field Documentation

6.182.2.1 [u_int8_t scsi_read_defect_data_10::alloc_length](#)[2]

Definition at line 113 of file `scsi_da.h`.

6.182.2.2 [u_int8_t scsi_read_defect_data_10::byte2](#)

Definition at line 102 of file `scsi_da.h`.

6.182.2.3 [u_int8_t scsi_read_defect_data_10::control](#)

Definition at line 115 of file `scsi_da.h`.

6.182.2.4 [u_int8_t scsi_read_defect_data_10::format](#)

Definition at line 109 of file `scsi_da.h`.

6.182.2.5 [u_int8_t scsi_read_defect_data_10::opcode](#)

Definition at line 95 of file `scsi_da.h`.

6.182.2.6 [u_int8_t scsi_read_defect_data_10::reserved](#)[4]

Definition at line 111 of file `scsi_da.h`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_da.h](#)

6.183 scsi_read_defect_data_12 Struct Reference

```
#include <scsi_da.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t byte2](#)
- [u_int8_t format](#)
- [u_int8_t reserved](#) [4]
- [u_int8_t alloc_length](#) [4]
- [u_int8_t control](#)

6.183.1 Detailed Description

Definition at line 118 of file `scsi_da.h`.

6.183.2 Field Documentation

6.183.2.1 [u_int8_t scsi_read_defect_data_12::alloc_length](#)[4]

Definition at line 139 of file `scsi_da.h`.

6.183.2.2 [u_int8_t scsi_read_defect_data_12::byte2](#)

Definition at line 127 of file `scsi_da.h`.

6.183.2.3 [u_int8_t scsi_read_defect_data_12::control](#)

Definition at line 141 of file `scsi_da.h`.

6.183.2.4 [u_int8_t scsi_read_defect_data_12::format](#)

Definition at line 135 of file `scsi_da.h`.

6.183.2.5 [u_int8_t scsi_read_defect_data_12::opcode](#)

Definition at line 120 of file `scsi_da.h`.

6.183.2.6 [u_int8_t scsi_read_defect_data_12::reserved](#)[4]

Definition at line 137 of file `scsi_da.h`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_da.h](#)

6.184 `scsi_read_defect_data_hdr_10` Struct Reference

```
#include <scsi_da.h>
```

Data Fields

- `u_int8_t reserved`
- `u_int8_t format`
- `u_int8_t length [2]`

6.184.1 Detailed Description

Definition at line 279 of file `scsi_da.h`.

6.184.2 Field Documentation

6.184.2.1 `u_int8_t scsi_read_defect_data_hdr_10::format`

Definition at line 288 of file `scsi_da.h`.

6.184.2.2 `u_int8_t scsi_read_defect_data_hdr_10::length[2]`

Definition at line 289 of file `scsi_da.h`.

6.184.2.3 `u_int8_t scsi_read_defect_data_hdr_10::reserved`

Definition at line 281 of file `scsi_da.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_da.h`

6.185 scsi_read_defect_data_hdr_12 Struct Reference

```
#include <scsi_da.h>
```

Data Fields

- `u_int8_t reserved`
- `u_int8_t format`
- `u_int8_t length [4]`

6.185.1 Detailed Description

Definition at line 311 of file `scsi_da.h`.

6.185.2 Field Documentation

6.185.2.1 `u_int8_t scsi_read_defect_data_hdr_12::format`

Definition at line 320 of file `scsi_da.h`.

6.185.2.2 `u_int8_t scsi_read_defect_data_hdr_12::length[4]`

Definition at line 321 of file `scsi_da.h`.

6.185.2.3 `u_int8_t scsi_read_defect_data_hdr_12::reserved`

Definition at line 313 of file `scsi_da.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_da.h`

6.186 `scsi_read_dvd_struct_data_bca` Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- `u_int8_t data_len` [2]
- `u_int8_t reserved` [2]
- `u_int8_t bca_info` [188]

6.186.1 Detailed Description

Definition at line 455 of file `scsi_cd.h`.

6.186.2 Field Documentation

6.186.2.1 `u_int8_t scsi_read_dvd_struct_data_bca::bca_info`[188]

Definition at line 459 of file `scsi_cd.h`.

6.186.2.2 `u_int8_t scsi_read_dvd_struct_data_bca::data_len`[2]

Definition at line 457 of file `scsi_cd.h`.

6.186.2.3 `u_int8_t scsi_read_dvd_struct_data_bca::reserved`[2]

Definition at line 458 of file `scsi_cd.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.h`

6.187 scsi_read_dvd_struct_data_copy_manage Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- [u_int8_t data_len](#) [2]
- [u_int8_t reserved0](#) [2]
- [u_int8_t byte4](#)
- [u_int8_t reserved1](#) [3]

6.187.1 Detailed Description

Definition at line 469 of file `scsi_cd.h`.

6.187.2 Field Documentation

6.187.2.1 [u_int8_t scsi_read_dvd_struct_data_copy_manage::byte4](#)

Definition at line 473 of file `scsi_cd.h`.

6.187.2.2 [u_int8_t scsi_read_dvd_struct_data_copy_manage::data_len](#)[2]

Definition at line 471 of file `scsi_cd.h`.

6.187.2.3 [u_int8_t scsi_read_dvd_struct_data_copy_manage::reserved0](#)[2]

Definition at line 472 of file `scsi_cd.h`.

6.187.2.4 [u_int8_t scsi_read_dvd_struct_data_copy_manage::reserved1](#)[3]

Definition at line 481 of file `scsi_cd.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.h`

6.188 `scsi_read_dvd_struct_data_copyright` Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- `u_int8_t data_len` [2]
- `u_int8_t reserved0` [2]
- `u_int8_t cps_type`
- `u_int8_t region_info`
- `u_int8_t reserved1` [2]

6.188.1 Detailed Description

Definition at line 437 of file `scsi_cd.h`.

6.188.2 Field Documentation

6.188.2.1 `u_int8_t scsi_read_dvd_struct_data_copyright::cps_type`

Definition at line 441 of file `scsi_cd.h`.

Referenced by `cdreaddvdstructure()`.

6.188.2.2 `u_int8_t scsi_read_dvd_struct_data_copyright::data_len`[2]

Definition at line 439 of file `scsi_cd.h`.

6.188.2.3 `u_int8_t scsi_read_dvd_struct_data_copyright::region_info`

Definition at line 444 of file `scsi_cd.h`.

Referenced by `cdreaddvdstructure()`.

6.188.2.4 `u_int8_t scsi_read_dvd_struct_data_copyright::reserved0`[2]

Definition at line 440 of file `scsi_cd.h`.

6.188.2.5 `u_int8_t scsi_read_dvd_struct_data_copyright::reserved1`[2]

Definition at line 445 of file `scsi_cd.h`.

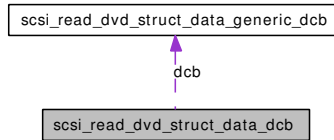
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.h`

6.189 scsi_read_dvd_struct_data_dcb Struct Reference

```
#include <scsi_cd.h>
```

Collaboration diagram for `scsi_read_dvd_struct_data_dcb`:



Data Fields

- `u_int8_t data_len` [2]
- `u_int8_t reserved` [2]
- `scsi_read_dvd_struct_data_generic_dcb dcb`

6.189.1 Detailed Description

Definition at line 607 of file `scsi_cd.h`.

6.189.2 Field Documentation

6.189.2.1 `u_int8_t scsi_read_dvd_struct_data_dcb::data_len`[2]

Definition at line 609 of file `scsi_cd.h`.

6.189.2.2 `struct scsi_read_dvd_struct_data_generic_dcb scsi_read_dvd_struct_data_dcb::dcb`

Definition at line 611 of file `scsi_cd.h`.

6.189.2.3 `u_int8_t scsi_read_dvd_struct_data_dcb::reserved`[2]

Definition at line 610 of file `scsi_cd.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.h`

6.190 `scsi_read_dvd_struct_data_dds` Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- `u_int8_t data_len` [2]
- `u_int8_t reserved` [2]
- `u_int8_t dds_info` [2048]

6.190.1 Detailed Description

Definition at line 501 of file `scsi_cd.h`.

6.190.2 Field Documentation

6.190.2.1 `u_int8_t scsi_read_dvd_struct_data_dds::data_len`[2]

Definition at line 503 of file `scsi_cd.h`.

6.190.2.2 `u_int8_t scsi_read_dvd_struct_data_dds::dds_info`[2048]

Definition at line 505 of file `scsi_cd.h`.

6.190.2.3 `u_int8_t scsi_read_dvd_struct_data_dds::reserved`[2]

Definition at line 504 of file `scsi_cd.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.h`

6.191 `scsi_read_dvd_struct_data_disc_id` Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- `u_int8_t data_len` [2]
- `u_int8_t reserved` [4]
- `u_int8_t random_num` [2]
- `u_int8_t year` [4]
- `u_int8_t month` [2]
- `u_int8_t day` [2]
- `u_int8_t hour` [2]
- `u_int8_t minute` [2]
- `u_int8_t second` [2]

6.191.1 Detailed Description

Definition at line 581 of file `scsi_cd.h`.

6.191.2 Field Documentation

6.191.2.1 `u_int8_t scsi_read_dvd_struct_data_disc_id::data_len`[2]

Definition at line 583 of file `scsi_cd.h`.

6.191.2.2 `u_int8_t scsi_read_dvd_struct_data_disc_id::day`[2]

Definition at line 588 of file `scsi_cd.h`.

6.191.2.3 `u_int8_t scsi_read_dvd_struct_data_disc_id::hour`[2]

Definition at line 589 of file `scsi_cd.h`.

6.191.2.4 `u_int8_t scsi_read_dvd_struct_data_disc_id::minute`[2]

Definition at line 590 of file `scsi_cd.h`.

6.191.2.5 `u_int8_t scsi_read_dvd_struct_data_disc_id::month`[2]

Definition at line 587 of file `scsi_cd.h`.

6.191.2.6 `u_int8_t scsi_read_dvd_struct_data_disc_id::random_num`[2]

Definition at line 585 of file `scsi_cd.h`.

6.191.2.7 `u_int8_t scsi_read_dvd_struct_data_disc_id::reserved[4]`

Definition at line 584 of file `scsi_cd.h`.

6.191.2.8 `u_int8_t scsi_read_dvd_struct_data_disc_id::second[2]`

Definition at line 591 of file `scsi_cd.h`.

6.191.2.9 `u_int8_t scsi_read_dvd_struct_data_disc_id::year[4]`

Definition at line 586 of file `scsi_cd.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.h`

6.192 scsi_read_dvd_struct_data_disc_key Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- [u_int8_t data_len](#) [2]
- [u_int8_t reserved](#) [2]
- [u_int8_t disc_key](#) [2048]

6.192.1 Detailed Description

Definition at line 448 of file [scsi_cd.h](#).

6.192.2 Field Documentation

6.192.2.1 [u_int8_t scsi_read_dvd_struct_data_disc_key::data_len](#)[2]

Definition at line 450 of file [scsi_cd.h](#).

6.192.2.2 [u_int8_t scsi_read_dvd_struct_data_disc_key::disc_key](#)[2048]

Definition at line 452 of file [scsi_cd.h](#).

6.192.2.3 [u_int8_t scsi_read_dvd_struct_data_disc_key::reserved](#)[2]

Definition at line 451 of file [scsi_cd.h](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_cd.h](#)

6.193 `scsi_read_dvd_struct_data_disc_key_blk` Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- `u_int8_t data_len` [2]
- `u_int8_t reserved`
- `u_int8_t total_packs`
- `u_int8_t disc_key_pack_data` [28668]

6.193.1 Detailed Description

Definition at line 491 of file `scsi_cd.h`.

6.193.2 Field Documentation

6.193.2.1 `u_int8_t scsi_read_dvd_struct_data_disc_key_blk::data_len`[2]

Definition at line 496 of file `scsi_cd.h`.

6.193.2.2 `u_int8_t scsi_read_dvd_struct_data_disc_key_blk::disc_key_pack_data`[28668]

Definition at line 499 of file `scsi_cd.h`.

6.193.2.3 `u_int8_t scsi_read_dvd_struct_data_disc_key_blk::reserved`

Definition at line 497 of file `scsi_cd.h`.

6.193.2.4 `u_int8_t scsi_read_dvd_struct_data_disc_key_blk::total_packs`

Definition at line 498 of file `scsi_cd.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.h`

6.194 scsi_read_dvd_struct_data_generic_dcb Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- [u_int8_t content_desc](#) [4]
- [u_int8_t unknown_desc_actions](#) [4]
- [u_int8_t vendor_id](#) [32]
- [u_int8_t dcb_data](#) [32728]

6.194.1 Detailed Description

Definition at line 594 of file `scsi_cd.h`.

6.194.2 Field Documentation

6.194.2.1 [u_int8_t scsi_read_dvd_struct_data_generic_dcb::content_desc](#)[4]

Definition at line 596 of file `scsi_cd.h`.

6.194.2.2 [u_int8_t scsi_read_dvd_struct_data_generic_dcb::dcb_data](#)[32728]

Definition at line 604 of file `scsi_cd.h`.

6.194.2.3 [u_int8_t scsi_read_dvd_struct_data_generic_dcb::unknown_desc_actions](#)[4]

Definition at line 598 of file `scsi_cd.h`.

6.194.2.4 [u_int8_t scsi_read_dvd_struct_data_generic_dcb::vendor_id](#)[32]

Definition at line 603 of file `scsi_cd.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.h`

6.195 `scsi_read_dvd_struct_data_header` Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- `u_int8_t data_len` [2]
- `u_int8_t reserved` [2]

6.195.1 Detailed Description

Definition at line 358 of file `scsi_cd.h`.

6.195.2 Field Documentation

6.195.2.1 `u_int8_t scsi_read_dvd_struct_data_header::data_len`[2]

Definition at line 360 of file `scsi_cd.h`.

6.195.2.2 `u_int8_t scsi_read_dvd_struct_data_header::reserved`[2]

Definition at line 361 of file `scsi_cd.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.h`

6.196 scsi_read_dvd_struct_data_layer_desc Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- [u_int8_t book_type_version](#)
- [u_int8_t disc_size_max_rate](#)
- [u_int8_t layer_info](#)
- [u_int8_t density](#)
- [u_int8_t zeros0](#)
- [u_int8_t main_data_start](#) [3]
- [u_int8_t zeros1](#)
- [u_int8_t main_data_end](#) [3]
- [u_int8_t zeros2](#)
- [u_int8_t end_sector_layer0](#) [3]
- [u_int8_t bca](#)
- [u_int8_t media_specific](#) [2031]

6.196.1 Detailed Description

Definition at line 364 of file `scsi_cd.h`.

6.196.2 Field Documentation

6.196.2.1 [u_int8_t scsi_read_dvd_struct_data_layer_desc::bca](#)

Definition at line 423 of file `scsi_cd.h`.

6.196.2.2 [u_int8_t scsi_read_dvd_struct_data_layer_desc::book_type_version](#)

Definition at line 366 of file `scsi_cd.h`.

6.196.2.3 [u_int8_t scsi_read_dvd_struct_data_layer_desc::density](#)

Definition at line 402 of file `scsi_cd.h`.

6.196.2.4 [u_int8_t scsi_read_dvd_struct_data_layer_desc::disc_size_max_rate](#)

Definition at line 379 of file `scsi_cd.h`.

6.196.2.5 [u_int8_t scsi_read_dvd_struct_data_layer_desc::end_sector_layer0](#)[3]

Definition at line 422 of file `scsi_cd.h`.

6.196.2.6 `u_int8_t scsi_read_dvd_struct_data_layer_desc::layer_info`

Definition at line 389 of file `scsi_cd.h`.

6.196.2.7 `u_int8_t scsi_read_dvd_struct_data_layer_desc::main_data_end[3]`

Definition at line 420 of file `scsi_cd.h`.

6.196.2.8 `u_int8_t scsi_read_dvd_struct_data_layer_desc::main_data_start[3]`

Definition at line 416 of file `scsi_cd.h`.

6.196.2.9 `u_int8_t scsi_read_dvd_struct_data_layer_desc::media_specific[2031]`

Definition at line 427 of file `scsi_cd.h`.

6.196.2.10 `u_int8_t scsi_read_dvd_struct_data_layer_desc::zeros0`

Definition at line 415 of file `scsi_cd.h`.

6.196.2.11 `u_int8_t scsi_read_dvd_struct_data_layer_desc::zeros1`

Definition at line 419 of file `scsi_cd.h`.

6.196.2.12 `u_int8_t scsi_read_dvd_struct_data_layer_desc::zeros2`

Definition at line 421 of file `scsi_cd.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.h`

6.197 scsi_read_dvd_struct_data_leadin Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- [u_int8_t data_len](#) [2]
- [u_int8_t reserved0](#) [2]
- [u_int8_t field_id_1](#)
- [u_int8_t app_code](#)
- [u_int8_t disc_physical_data](#)
- [u_int8_t last_addr](#) [3]
- [u_int8_t reserved1](#) [2]
- [u_int8_t field_id_2](#)
- [u_int8_t rwp](#)
- [u_int8_t rwp_wavelength](#)
- [u_int8_t optimum_write_strategy](#)
- [u_int8_t reserved2](#) [4]
- [u_int8_t field_id_3](#)
- [u_int8_t manuf_id_17_12](#) [6]
- [u_int8_t reserved3](#)
- [u_int8_t field_id_4](#)
- [u_int8_t manuf_id_11_6](#) [6]
- [u_int8_t reserved4](#)
- [u_int8_t field_id_5](#)
- [u_int8_t manuf_id_5_0](#) [6]
- [u_int8_t reserved5](#) [25]

6.197.1 Detailed Description

Definition at line 556 of file `scsi_cd.h`.

6.197.2 Field Documentation

6.197.2.1 [u_int8_t scsi_read_dvd_struct_data_leadin::app_code](#)

Definition at line 561 of file `scsi_cd.h`.

6.197.2.2 [u_int8_t scsi_read_dvd_struct_data_leadin::data_len](#)[2]

Definition at line 558 of file `scsi_cd.h`.

6.197.2.3 [u_int8_t scsi_read_dvd_struct_data_leadin::disc_physical_data](#)

Definition at line 562 of file `scsi_cd.h`.

6.197.2.4 `u_int8_t scsi_read_dvd_struct_data_lead_in::field_id_1`

Definition at line 560 of file `scsi_cd.h`.

6.197.2.5 `u_int8_t scsi_read_dvd_struct_data_lead_in::field_id_2`

Definition at line 565 of file `scsi_cd.h`.

6.197.2.6 `u_int8_t scsi_read_dvd_struct_data_lead_in::field_id_3`

Definition at line 570 of file `scsi_cd.h`.

6.197.2.7 `u_int8_t scsi_read_dvd_struct_data_lead_in::field_id_4`

Definition at line 573 of file `scsi_cd.h`.

6.197.2.8 `u_int8_t scsi_read_dvd_struct_data_lead_in::field_id_5`

Definition at line 576 of file `scsi_cd.h`.

6.197.2.9 `u_int8_t scsi_read_dvd_struct_data_lead_in::last_addr[3]`

Definition at line 563 of file `scsi_cd.h`.

6.197.2.10 `u_int8_t scsi_read_dvd_struct_data_lead_in::manuf_id_11_6[6]`

Definition at line 574 of file `scsi_cd.h`.

6.197.2.11 `u_int8_t scsi_read_dvd_struct_data_lead_in::manuf_id_17_12[6]`

Definition at line 571 of file `scsi_cd.h`.

6.197.2.12 `u_int8_t scsi_read_dvd_struct_data_lead_in::manuf_id_5_0[6]`

Definition at line 577 of file `scsi_cd.h`.

6.197.2.13 `u_int8_t scsi_read_dvd_struct_data_lead_in::optimum_write_strategy`

Definition at line 568 of file `scsi_cd.h`.

6.197.2.14 `u_int8_t scsi_read_dvd_struct_data_lead_in::reserved0[2]`

Definition at line 559 of file `scsi_cd.h`.

6.197.2.15 `u_int8_t scsi_read_dvd_struct_data_leadin::reserved1[2]`

Definition at line 564 of file `scsi_cd.h`.

6.197.2.16 `u_int8_t scsi_read_dvd_struct_data_leadin::reserved2[4]`

Definition at line 569 of file `scsi_cd.h`.

6.197.2.17 `u_int8_t scsi_read_dvd_struct_data_leadin::reserved3`

Definition at line 572 of file `scsi_cd.h`.

6.197.2.18 `u_int8_t scsi_read_dvd_struct_data_leadin::reserved4`

Definition at line 575 of file `scsi_cd.h`.

6.197.2.19 `u_int8_t scsi_read_dvd_struct_data_leadin::reserved5[25]`

Definition at line 578 of file `scsi_cd.h`.

6.197.2.20 `u_int8_t scsi_read_dvd_struct_data_leadin::rwp`

Definition at line 566 of file `scsi_cd.h`.

6.197.2.21 `u_int8_t scsi_read_dvd_struct_data_leadin::rwp_wavelength`

Definition at line 567 of file `scsi_cd.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.h`

6.198 `scsi_read_dvd_struct_data_manufacturer` Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- `u_int8_t data_len` [2]
- `u_int8_t reserved` [2]
- `u_int8_t manuf_info` [2048]

6.198.1 Detailed Description

Definition at line 462 of file `scsi_cd.h`.

6.198.2 Field Documentation

6.198.2.1 `u_int8_t scsi_read_dvd_struct_data_manufacturer::data_len`[2]

Definition at line 464 of file `scsi_cd.h`.

6.198.2.2 `u_int8_t scsi_read_dvd_struct_data_manufacturer::manuf_info`[2048]

Definition at line 466 of file `scsi_cd.h`.

6.198.2.3 `u_int8_t scsi_read_dvd_struct_data_manufacturer::reserved`[2]

Definition at line 465 of file `scsi_cd.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.h`

6.199 scsi_read_dvd_struct_data_medium_status Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- [u_int8_t data_len](#) [2]
- [u_int8_t reserved0](#) [2]
- [u_int8_t byte4](#)
- [u_int8_t disc_type_id](#)
- [u_int8_t reserved1](#)
- [u_int8_t ram_swi_info](#)

6.199.1 Detailed Description

Definition at line 508 of file `scsi_cd.h`.

6.199.2 Field Documentation

6.199.2.1 [u_int8_t scsi_read_dvd_struct_data_medium_status::byte4](#)

Definition at line 512 of file `scsi_cd.h`.

6.199.2.2 [u_int8_t scsi_read_dvd_struct_data_medium_status::data_len](#)[2]

Definition at line 510 of file `scsi_cd.h`.

6.199.2.3 [u_int8_t scsi_read_dvd_struct_data_medium_status::disc_type_id](#)

Definition at line 518 of file `scsi_cd.h`.

6.199.2.4 [u_int8_t scsi_read_dvd_struct_data_medium_status::ram_swi_info](#)

Definition at line 522 of file `scsi_cd.h`.

6.199.2.5 [u_int8_t scsi_read_dvd_struct_data_medium_status::reserved0](#)[2]

Definition at line 511 of file `scsi_cd.h`.

6.199.2.6 [u_int8_t scsi_read_dvd_struct_data_medium_status::reserved1](#)

Definition at line 521 of file `scsi_cd.h`.

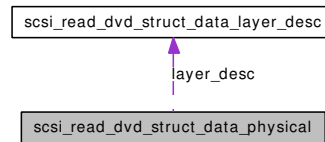
The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_cd.h](#)

6.200 scsi_read_dvd_struct_data_physical Struct Reference

```
#include <scsi_cd.h>
```

Collaboration diagram for `scsi_read_dvd_struct_data_physical`:



Data Fields

- `u_int8_t data_len` [2]
- `u_int8_t reserved` [2]
- `scsi_read_dvd_struct_data_layer_desc layer_desc`

6.200.1 Detailed Description

Definition at line 430 of file `scsi_cd.h`.

6.200.2 Field Documentation

6.200.2.1 `u_int8_t scsi_read_dvd_struct_data_physical::data_len`[2]

Definition at line 432 of file `scsi_cd.h`.

6.200.2.2 `struct scsi_read_dvd_struct_data_layer_desc scsi_read_dvd_struct_data_physical::layer_desc`

Definition at line 434 of file `scsi_cd.h`.

Referenced by `cdreadvdstructure()`.

6.200.2.3 `u_int8_t scsi_read_dvd_struct_data_physical::reserved`[2]

Definition at line 433 of file `scsi_cd.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.h`

6.201 scsi_read_dvd_struct_data_prot_discid Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- `u_int8_t data_len` [2]
- `u_int8_t reserved` [2]
- `u_int8_t prot_discid_data` [16]

6.201.1 Detailed Description

Definition at line 484 of file `scsi_cd.h`.

6.201.2 Field Documentation

6.201.2.1 `u_int8_t scsi_read_dvd_struct_data_prot_discid::data_len`[2]

Definition at line 486 of file `scsi_cd.h`.

6.201.2.2 `u_int8_t scsi_read_dvd_struct_data_prot_discid::prot_discid_data`[16]

Definition at line 488 of file `scsi_cd.h`.

6.201.2.3 `u_int8_t scsi_read_dvd_struct_data_prot_discid::reserved`[2]

Definition at line 487 of file `scsi_cd.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.h`

6.202 `scsi_read_dvd_struct_data_rmd` Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- `u_int8_t data_len` [2]
- `u_int8_t reserved` [2]
- `u_int8_t last_sector_num` [4]
- `u_int8_t rmd_bytes` [32768]

6.202.1 Detailed Description

Definition at line 543 of file `scsi_cd.h`.

6.202.2 Field Documentation

6.202.2.1 `u_int8_t scsi_read_dvd_struct_data_rmd::data_len`[2]

Definition at line 545 of file `scsi_cd.h`.

6.202.2.2 `u_int8_t scsi_read_dvd_struct_data_rmd::last_sector_num`[4]

Definition at line 547 of file `scsi_cd.h`.

6.202.2.3 `u_int8_t scsi_read_dvd_struct_data_rmd::reserved`[2]

Definition at line 546 of file `scsi_cd.h`.

6.202.2.4 `u_int8_t scsi_read_dvd_struct_data_rmd::rmd_bytes`[32768]

Definition at line 548 of file `scsi_cd.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.h`

6.203 scsi_read_dvd_struct_data_rmd_borderout Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- [u_int8_t data_len](#) [2]
- [u_int8_t reserved](#) [2]
- [u_int8_t rmd](#) [30720]

6.203.1 Detailed Description

Definition at line 536 of file `scsi_cd.h`.

6.203.2 Field Documentation

6.203.2.1 [u_int8_t scsi_read_dvd_struct_data_rmd_borderout::data_len](#)[2]

Definition at line 538 of file `scsi_cd.h`.

6.203.2.2 [u_int8_t scsi_read_dvd_struct_data_rmd_borderout::reserved](#)[2]

Definition at line 539 of file `scsi_cd.h`.

6.203.2.3 [u_int8_t scsi_read_dvd_struct_data_rmd_borderout::rmd](#)[30720]

Definition at line 540 of file `scsi_cd.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.h`

6.204 `scsi_read_dvd_struct_data_spare_area` Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- `u_int8_t data_len` [2]
- `u_int8_t reserved` [2]
- `u_int8_t unused_primary` [4]
- `u_int8_t unused_supl` [4]
- `u_int8_t allocated_supl` [4]

6.204.1 Detailed Description

Definition at line 527 of file `scsi_cd.h`.

6.204.2 Field Documentation

6.204.2.1 `u_int8_t scsi_read_dvd_struct_data_spare_area::allocated_supl`[4]

Definition at line 533 of file `scsi_cd.h`.

6.204.2.2 `u_int8_t scsi_read_dvd_struct_data_spare_area::data_len`[2]

Definition at line 529 of file `scsi_cd.h`.

6.204.2.3 `u_int8_t scsi_read_dvd_struct_data_spare_area::reserved`[2]

Definition at line 530 of file `scsi_cd.h`.

6.204.2.4 `u_int8_t scsi_read_dvd_struct_data_spare_area::unused_primary`[4]

Definition at line 531 of file `scsi_cd.h`.

6.204.2.5 `u_int8_t scsi_read_dvd_struct_data_spare_area::unused_supl`[4]

Definition at line 532 of file `scsi_cd.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.h`

6.205 scsi_read_dvd_structure Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t reserved](#)
- [u_int8_t address](#) [4]
- [u_int8_t layer_number](#)
- [u_int8_t format](#)
- [u_int8_t alloc_len](#) [2]
- [u_int8_t agid](#)
- [u_int8_t control](#)

6.205.1 Detailed Description

Definition at line 217 of file `scsi_cd.h`.

6.205.2 Field Documentation

6.205.2.1 [u_int8_t scsi_read_dvd_structure::address](#)[4]

Definition at line 221 of file `scsi_cd.h`.

6.205.2.2 [u_int8_t scsi_read_dvd_structure::agid](#)

Definition at line 243 of file `scsi_cd.h`.

6.205.2.3 [u_int8_t scsi_read_dvd_structure::alloc_len](#)[2]

Definition at line 242 of file `scsi_cd.h`.

6.205.2.4 [u_int8_t scsi_read_dvd_structure::control](#)

Definition at line 244 of file `scsi_cd.h`.

6.205.2.5 [u_int8_t scsi_read_dvd_structure::format](#)

Definition at line 223 of file `scsi_cd.h`.

6.205.2.6 [u_int8_t scsi_read_dvd_structure::layer_number](#)

Definition at line 222 of file `scsi_cd.h`.

6.205.2.7 u_int8_t scsi_read_dvd_structure::opcode

Definition at line 219 of file scsi_cd.h.

6.205.2.8 u_int8_t scsi_read_dvd_structure::reserved

Definition at line 220 of file scsi_cd.h.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_cd.h](#)

6.206 scsi_read_element_status Struct Reference

```
#include <scsi_ch.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t byte2](#)
- [u_int8_t sea](#) [2]
- [u_int8_t count](#) [2]
- [u_int8_t reserved0](#)
- [u_int8_t len](#) [3]
- [u_int8_t reserved1](#)
- [u_int8_t control](#)

6.206.1 Detailed Description

Definition at line 138 of file `scsi_ch.h`.

6.206.2 Field Documentation

6.206.2.1 [u_int8_t scsi_read_element_status::byte2](#)

Definition at line 141 of file `scsi_ch.h`.

6.206.2.2 [u_int8_t scsi_read_element_status::control](#)

Definition at line 149 of file `scsi_ch.h`.

6.206.2.3 [u_int8_t scsi_read_element_status::count](#)[2]

Definition at line 145 of file `scsi_ch.h`.

6.206.2.4 [u_int8_t scsi_read_element_status::len](#)[3]

Definition at line 147 of file `scsi_ch.h`.

6.206.2.5 [u_int8_t scsi_read_element_status::opcode](#)

Definition at line 139 of file `scsi_ch.h`.

6.206.2.6 [u_int8_t scsi_read_element_status::reserved0](#)

Definition at line 146 of file `scsi_ch.h`.

6.206.2.7 `u_int8_t scsi_read_element_status::reserved1`

Definition at line 148 of file `scsi_ch.h`.

6.206.2.8 `u_int8_t scsi_read_element_status::sea[2]`

Definition at line 144 of file `scsi_ch.h`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_ch.h](#)

6.207 scsi_read_format_capacities Struct Reference

```
#include <scsi_da.h>
```

Data Fields

- [uint8_t opcode](#)
- [uint8_t byte2](#)
- [uint8_t reserved0](#) [5]
- [uint8_t alloc_length](#) [2]
- [uint8_t reserved1](#) [3]

6.207.1 Detailed Description

Definition at line 188 of file `scsi_da.h`.

6.207.2 Field Documentation

6.207.2.1 [uint8_t scsi_read_format_capacities::alloc_length](#)[2]

Definition at line 194 of file `scsi_da.h`.

6.207.2.2 [uint8_t scsi_read_format_capacities::byte2](#)

Definition at line 191 of file `scsi_da.h`.

6.207.2.3 [uint8_t scsi_read_format_capacities::opcode](#)

Definition at line 190 of file `scsi_da.h`.

6.207.2.4 [uint8_t scsi_read_format_capacities::reserved0](#)[5]

Definition at line 193 of file `scsi_da.h`.

6.207.2.5 [uint8_t scsi_read_format_capacities::reserved1](#)[3]

Definition at line 195 of file `scsi_da.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_da.h`

6.208 scsi_read_header Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- [u_int8_t op_code](#)
- [u_int8_t byte2](#)
- [u_int8_t blk_addr](#) [4]
- [u_int8_t unused](#)
- [u_int8_t data_len](#) [2]
- [u_int8_t control](#)

6.208.1 Detailed Description

Definition at line 127 of file [scsi_cd.h](#).

6.208.2 Field Documentation

6.208.2.1 [u_int8_t scsi_read_header::blk_addr](#)[4]

Definition at line 131 of file [scsi_cd.h](#).

6.208.2.2 [u_int8_t scsi_read_header::byte2](#)

Definition at line 130 of file [scsi_cd.h](#).

6.208.2.3 [u_int8_t scsi_read_header::control](#)

Definition at line 134 of file [scsi_cd.h](#).

6.208.2.4 [u_int8_t scsi_read_header::data_len](#)[2]

Definition at line 133 of file [scsi_cd.h](#).

6.208.2.5 [u_int8_t scsi_read_header::op_code](#)

Definition at line 129 of file [scsi_cd.h](#).

6.208.2.6 [u_int8_t scsi_read_header::unused](#)

Definition at line 132 of file [scsi_cd.h](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_cd.h](#)

6.209 scsi_read_subchannel Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- [u_int8_t op_code](#)
- [u_int8_t byte1](#)
- [u_int8_t byte2](#)
- [u_int8_t subchan_format](#)
- [u_int8_t unused](#) [2]
- [u_int8_t track](#)
- [u_int8_t data_len](#) [2]
- [u_int8_t control](#)

6.209.1 Detailed Description

Definition at line 137 of file `scsi_cd.h`.

6.209.2 Field Documentation

6.209.2.1 [u_int8_t scsi_read_subchannel::byte1](#)

Definition at line 140 of file `scsi_cd.h`.

6.209.2.2 [u_int8_t scsi_read_subchannel::byte2](#)

Definition at line 141 of file `scsi_cd.h`.

6.209.2.3 [u_int8_t scsi_read_subchannel::control](#)

Definition at line 147 of file `scsi_cd.h`.

6.209.2.4 [u_int8_t scsi_read_subchannel::data_len](#)[2]

Definition at line 146 of file `scsi_cd.h`.

6.209.2.5 [u_int8_t scsi_read_subchannel::op_code](#)

Definition at line 139 of file `scsi_cd.h`.

6.209.2.6 [u_int8_t scsi_read_subchannel::subchan_format](#)

Definition at line 143 of file `scsi_cd.h`.

6.209.2.7 `u_int8_t scsi_read_subchannel::track`

Definition at line 145 of file `scsi_cd.h`.

6.209.2.8 `u_int8_t scsi_read_subchannel::unused[2]`

Definition at line 144 of file `scsi_cd.h`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_cd.h](#)

6.210 scsi_read_toc Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- [u_int8_t op_code](#)
- [u_int8_t byte2](#)
- [u_int8_t unused](#) [4]
- [u_int8_t from_track](#)
- [u_int8_t data_len](#) [2]
- [u_int8_t control](#)

6.210.1 Detailed Description

Definition at line 150 of file [scsi_cd.h](#).

6.210.2 Field Documentation

6.210.2.1 [u_int8_t scsi_read_toc::byte2](#)

Definition at line 153 of file [scsi_cd.h](#).

6.210.2.2 [u_int8_t scsi_read_toc::control](#)

Definition at line 157 of file [scsi_cd.h](#).

6.210.2.3 [u_int8_t scsi_read_toc::data_len](#)[2]

Definition at line 156 of file [scsi_cd.h](#).

6.210.2.4 [u_int8_t scsi_read_toc::from_track](#)

Definition at line 155 of file [scsi_cd.h](#).

6.210.2.5 [u_int8_t scsi_read_toc::op_code](#)

Definition at line 152 of file [scsi_cd.h](#).

6.210.2.6 [u_int8_t scsi_read_toc::unused](#)[4]

Definition at line 154 of file [scsi_cd.h](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_cd.h](#)

6.211 scsi_reassign_blocks Struct Reference

```
#include <scsi_da.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t byte2](#)
- [u_int8_t unused](#) [3]
- [u_int8_t control](#)

6.211.1 Detailed Description

Definition at line 85 of file `scsi_da.h`.

6.211.2 Field Documentation

6.211.2.1 [u_int8_t scsi_reassign_blocks::byte2](#)

Definition at line 88 of file `scsi_da.h`.

6.211.2.2 [u_int8_t scsi_reassign_blocks::control](#)

Definition at line 90 of file `scsi_da.h`.

6.211.2.3 [u_int8_t scsi_reassign_blocks::opcode](#)

Definition at line 87 of file `scsi_da.h`.

6.211.2.4 [u_int8_t scsi_reassign_blocks::unused](#)[3]

Definition at line 89 of file `scsi_da.h`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_da.h](#)

6.212 scsi_reassign_blocks_data Struct Reference

```
#include <scsi_da.h>
```

Data Fields

- `u_int8_t reserved` [2]
- `u_int8_t length` [2]
- struct {
 - `u_int8_t dlbaddr` [4]
 - `defect_descriptor` [1]

6.212.1 Detailed Description

Definition at line 264 of file `scsi_da.h`.

6.212.2 Field Documentation

6.212.2.1 `struct { ... } scsi_reassign_blocks_data::defect_descriptor`[1]

6.212.2.2 `u_int8_t scsi_reassign_blocks_data::dlbaddr`[4]

Definition at line 269 of file `scsi_da.h`.

6.212.2.3 `u_int8_t scsi_reassign_blocks_data::length`[2]

Definition at line 267 of file `scsi_da.h`.

6.212.2.4 `u_int8_t scsi_reassign_blocks_data::reserved`[2]

Definition at line 266 of file `scsi_da.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_da.h`

6.213 scsi_release Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t byte2](#)
- [u_int8_t unused](#) [2]
- [u_int8_t length](#)
- [u_int8_t control](#)

6.213.1 Detailed Description

Definition at line 341 of file `scsi_all.h`.

6.213.2 Field Documentation

6.213.2.1 [u_int8_t scsi_release::byte2](#)

Definition at line 344 of file `scsi_all.h`.

6.213.2.2 [u_int8_t scsi_release::control](#)

Definition at line 347 of file `scsi_all.h`.

6.213.2.3 [u_int8_t scsi_release::length](#)

Definition at line 346 of file `scsi_all.h`.

6.213.2.4 [u_int8_t scsi_release::opcode](#)

Definition at line 343 of file `scsi_all.h`.

6.213.2.5 [u_int8_t scsi_release::unused](#)[2]

Definition at line 345 of file `scsi_all.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.214 scsi_report_key Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t reserved0](#)
- [u_int8_t lba](#) [4]
- [u_int8_t reserved1](#) [2]
- [u_int8_t alloc_len](#) [2]
- [u_int8_t agid_keyformat](#)
- [u_int8_t control](#)

6.214.1 Detailed Description

Definition at line 182 of file `scsi_cd.h`.

6.214.2 Field Documentation

6.214.2.1 [u_int8_t scsi_report_key::agid_keyformat](#)

Definition at line 189 of file `scsi_cd.h`.

6.214.2.2 [u_int8_t scsi_report_key::alloc_len](#)[2]

Definition at line 188 of file `scsi_cd.h`.

6.214.2.3 [u_int8_t scsi_report_key::control](#)

Definition at line 202 of file `scsi_cd.h`.

6.214.2.4 [u_int8_t scsi_report_key::lba](#)[4]

Definition at line 186 of file `scsi_cd.h`.

6.214.2.5 [u_int8_t scsi_report_key::opcode](#)

Definition at line 184 of file `scsi_cd.h`.

6.214.2.6 [u_int8_t scsi_report_key::reserved0](#)

Definition at line 185 of file `scsi_cd.h`.

6.214.2.7 `u_int8_t scsi_report_key::reserved1[2]`

Definition at line 187 of file `scsi_cd.h`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_cd.h](#)

6.215 scsi_report_key_data_agid Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- [u_int8_t data_len](#) [2]
- [u_int8_t reserved](#) [5]
- [u_int8_t agid](#)

6.215.1 Detailed Description

Definition at line 272 of file [scsi_cd.h](#).

6.215.2 Field Documentation

6.215.2.1 [u_int8_t scsi_report_key_data_agid::agid](#)

Definition at line 276 of file [scsi_cd.h](#).

Referenced by [cdreportkey\(\)](#).

6.215.2.2 [u_int8_t scsi_report_key_data_agid::data_len](#)[2]

Definition at line 274 of file [scsi_cd.h](#).

6.215.2.3 [u_int8_t scsi_report_key_data_agid::reserved](#)[5]

Definition at line 275 of file [scsi_cd.h](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_cd.h](#)

6.216 scsi_report_key_data_asf Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- [u_int8_t data_len](#) [2]
- [u_int8_t reserved](#) [5]
- [u_int8_t success](#)

6.216.1 Detailed Description

Definition at line 316 of file [scsi_cd.h](#).

6.216.2 Field Documentation

6.216.2.1 [u_int8_t scsi_report_key_data_asf::data_len](#)[2]

Definition at line 318 of file [scsi_cd.h](#).

6.216.2.2 [u_int8_t scsi_report_key_data_asf::reserved](#)[5]

Definition at line 319 of file [scsi_cd.h](#).

6.216.2.3 [u_int8_t scsi_report_key_data_asf::success](#)

Definition at line 320 of file [scsi_cd.h](#).

Referenced by [cdreportkey\(\)](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_cd.h](#)

6.217 scsi_report_key_data_challenge Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- [u_int8_t data_len](#) [2]
- [u_int8_t reserved0](#) [2]
- [u_int8_t challenge_key](#) [10]
- [u_int8_t reserved1](#) [2]

6.217.1 Detailed Description

Definition at line 281 of file `scsi_cd.h`.

6.217.2 Field Documentation

6.217.2.1 [u_int8_t scsi_report_key_data_challenge::challenge_key](#)[10]

Definition at line 285 of file `scsi_cd.h`.

Referenced by `cdreportkey()`, and `cdsendkey()`.

6.217.2.2 [u_int8_t scsi_report_key_data_challenge::data_len](#)[2]

Definition at line 283 of file `scsi_cd.h`.

Referenced by `cdsendkey()`.

6.217.2.3 [u_int8_t scsi_report_key_data_challenge::reserved0](#)[2]

Definition at line 284 of file `scsi_cd.h`.

6.217.2.4 [u_int8_t scsi_report_key_data_challenge::reserved1](#)[2]

Definition at line 286 of file `scsi_cd.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.h`

6.218 scsi_report_key_data_header Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- [u_int8_t data_len](#) [2]
- [u_int8_t reserved](#) [2]

6.218.1 Detailed Description

Definition at line 266 of file [scsi_cd.h](#).

6.218.2 Field Documentation

6.218.2.1 [u_int8_t scsi_report_key_data_header::data_len](#)[2]

Definition at line 268 of file [scsi_cd.h](#).

6.218.2.2 [u_int8_t scsi_report_key_data_header::reserved](#)[2]

Definition at line 269 of file [scsi_cd.h](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_cd.h](#)

6.219 scsi_report_key_data_key1_key2 Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- [u_int8_t data_len](#) [2]
- [u_int8_t reserved0](#) [2]
- [u_int8_t key1](#) [5]
- [u_int8_t reserved1](#) [3]

6.219.1 Detailed Description

Definition at line 289 of file `scsi_cd.h`.

6.219.2 Field Documentation

6.219.2.1 [u_int8_t scsi_report_key_data_key1_key2::data_len](#)[2]

Definition at line 291 of file `scsi_cd.h`.

Referenced by `cdsendkey()`.

6.219.2.2 [u_int8_t scsi_report_key_data_key1_key2::key1](#)[5]

Definition at line 293 of file `scsi_cd.h`.

Referenced by `cdreportkey()`, and `cdsendkey()`.

6.219.2.3 [u_int8_t scsi_report_key_data_key1_key2::reserved0](#)[2]

Definition at line 292 of file `scsi_cd.h`.

6.219.2.4 [u_int8_t scsi_report_key_data_key1_key2::reserved1](#)[3]

Definition at line 294 of file `scsi_cd.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.h`

6.220 `scsi_report_key_data_rpc` Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- `u_int8_t data_len` [2]
- `u_int8_t rpc_scheme0`
- `u_int8_t reserved0`
- `u_int8_t byte4`
- `u_int8_t region_mask`
- `u_int8_t rpc_scheme1`
- `u_int8_t reserved1`

6.220.1 Detailed Description

Definition at line 324 of file `scsi_cd.h`.

6.220.2 Field Documentation

6.220.2.1 `u_int8_t scsi_report_key_data_rpc::byte4`

Definition at line 331 of file `scsi_cd.h`.

Referenced by `cdreportkey()`.

6.220.2.2 `u_int8_t scsi_report_key_data_rpc::data_len`[2]

Definition at line 326 of file `scsi_cd.h`.

6.220.2.3 `u_int8_t scsi_report_key_data_rpc::region_mask`

Definition at line 342 of file `scsi_cd.h`.

Referenced by `cdreportkey()`.

6.220.2.4 `u_int8_t scsi_report_key_data_rpc::reserved0`

Definition at line 330 of file `scsi_cd.h`.

6.220.2.5 `u_int8_t scsi_report_key_data_rpc::reserved1`

Definition at line 344 of file `scsi_cd.h`.

6.220.2.6 `u_int8_t scsi_report_key_data_rpc::rpc_scheme0`

Definition at line 327 of file `scsi_cd.h`.

6.220.2.7 `u_int8_t scsi_report_key_data_rpc::rpc_scheme1`

Definition at line 343 of file `scsi_cd.h`.

Referenced by `cdreportkey()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.h`

6.221 `scsi_report_key_data_title` Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- `u_int8_t data_len` [2]
- `u_int8_t reserved0` [2]
- `u_int8_t byte0`
- `u_int8_t title_key` [5]
- `u_int8_t reserved1` [2]

6.221.1 Detailed Description

Definition at line 297 of file `scsi_cd.h`.

6.221.2 Field Documentation

6.221.2.1 `u_int8_t scsi_report_key_data_title::byte0`

Definition at line 301 of file `scsi_cd.h`.

Referenced by `cdreportkey()`.

6.221.2.2 `u_int8_t scsi_report_key_data_title::data_len`[2]

Definition at line 299 of file `scsi_cd.h`.

6.221.2.3 `u_int8_t scsi_report_key_data_title::reserved0`[2]

Definition at line 300 of file `scsi_cd.h`.

6.221.2.4 `u_int8_t scsi_report_key_data_title::reserved1`[2]

Definition at line 313 of file `scsi_cd.h`.

6.221.2.5 `u_int8_t scsi_report_key_data_title::title_key`[5]

Definition at line 312 of file `scsi_cd.h`.

Referenced by `cdreportkey()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.h`

6.222 scsi_report_luns Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- [uint8_t opcode](#)
- [uint8_t reserved1](#)
- [uint8_t select_report](#)
- [uint8_t reserved2](#) [3]
- [uint8_t length](#) [4]
- [uint8_t reserved3](#)
- [uint8_t control](#)

6.222.1 Detailed Description

Definition at line 711 of file `scsi_all.h`.

6.222.2 Field Documentation

6.222.2.1 [uint8_t scsi_report_luns::control](#)

Definition at line 722 of file `scsi_all.h`.

6.222.2.2 [uint8_t scsi_report_luns::length](#)[4]

Definition at line 720 of file `scsi_all.h`.

6.222.2.3 [uint8_t scsi_report_luns::opcode](#)

Definition at line 713 of file `scsi_all.h`.

6.222.2.4 [uint8_t scsi_report_luns::reserved1](#)

Definition at line 714 of file `scsi_all.h`.

6.222.2.5 [uint8_t scsi_report_luns::reserved2](#)[3]

Definition at line 719 of file `scsi_all.h`.

6.222.2.6 [uint8_t scsi_report_luns::reserved3](#)

Definition at line 721 of file `scsi_all.h`.

6.222.2.7 uint8_t scsi_report_luns::select_report

Definition at line 718 of file scsi_all.h.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_all.h](#)

6.223 `scsi_report_luns_data` Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- `u_int8_t length` [4]
- `u_int8_t reserved` [4]
- struct {
 - `u_int8_t lundata` [8]
- `luns` [0]

6.223.1 Detailed Description

Definition at line 725 of file `scsi_all.h`.

6.223.2 Field Documentation

6.223.2.1 `u_int8_t scsi_report_luns_data::length`[4]

Definition at line 726 of file `scsi_all.h`.

6.223.2.2 `u_int8_t scsi_report_luns_data::lundata`[8]

Definition at line 732 of file `scsi_all.h`.

6.223.2.3 struct { ... } `scsi_report_luns_data::luns`[0]

6.223.2.4 `u_int8_t scsi_report_luns_data::reserved`[4]

Definition at line 727 of file `scsi_all.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.224 `scsi_request_sense` Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- `u_int8_t opcode`
- `u_int8_t byte2`
- `u_int8_t unused` [2]
- `u_int8_t length`
- `u_int8_t control`

6.224.1 Detailed Description

Definition at line 114 of file `scsi_all.h`.

6.224.2 Field Documentation

6.224.2.1 `u_int8_t scsi_request_sense::byte2`

Definition at line 117 of file `scsi_all.h`.

6.224.2.2 `u_int8_t scsi_request_sense::control`

Definition at line 120 of file `scsi_all.h`.

6.224.2.3 `u_int8_t scsi_request_sense::length`

Definition at line 119 of file `scsi_all.h`.

Referenced by `targbhdone()`.

6.224.2.4 `u_int8_t scsi_request_sense::opcode`

Definition at line 116 of file `scsi_all.h`.

6.224.2.5 `u_int8_t scsi_request_sense::unused`[2]

Definition at line 118 of file `scsi_all.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.225 `scsi_request_volume_element_address` Struct Reference

```
#include <scsi_ch.h>
```

Data Fields

- `u_int8_t opcode`
- `u_int8_t byte2`
- `u_int8_t eaddr` [2]
- `u_int8_t count` [2]
- `u_int8_t reserved0`
- `u_int8_t len` [3]
- `u_int8_t reserved1`
- `u_int8_t control`

6.225.1 Detailed Description

Definition at line 152 of file `scsi_ch.h`.

6.225.2 Field Documentation

6.225.2.1 `u_int8_t scsi_request_volume_element_address::byte2`

Definition at line 155 of file `scsi_ch.h`.

6.225.2.2 `u_int8_t scsi_request_volume_element_address::control`

Definition at line 163 of file `scsi_ch.h`.

6.225.2.3 `u_int8_t scsi_request_volume_element_address::count`[2]

Definition at line 159 of file `scsi_ch.h`.

6.225.2.4 `u_int8_t scsi_request_volume_element_address::eaddr`[2]

Definition at line 158 of file `scsi_ch.h`.

6.225.2.5 `u_int8_t scsi_request_volume_element_address::len`[3]

Definition at line 161 of file `scsi_ch.h`.

6.225.2.6 `u_int8_t scsi_request_volume_element_address::opcode`

Definition at line 153 of file `scsi_ch.h`.

6.225.2.7 `u_int8_t scsi_request_volume_element_address::reserved0`

Definition at line 160 of file `scsi_ch.h`.

6.225.2.8 `u_int8_t scsi_request_volume_element_address::reserved1`

Definition at line 162 of file `scsi_ch.h`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_ch.h](#)

6.226 scsi_reserve Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t byte2](#)
- [u_int8_t unused](#) [2]
- [u_int8_t length](#)
- [u_int8_t control](#)

6.226.1 Detailed Description

Definition at line 332 of file `scsi_all.h`.

6.226.2 Field Documentation

6.226.2.1 [u_int8_t scsi_reserve::byte2](#)

Definition at line 335 of file `scsi_all.h`.

6.226.2.2 [u_int8_t scsi_reserve::control](#)

Definition at line 338 of file `scsi_all.h`.

6.226.2.3 [u_int8_t scsi_reserve::length](#)

Definition at line 337 of file `scsi_all.h`.

6.226.2.4 [u_int8_t scsi_reserve::opcode](#)

Definition at line 334 of file `scsi_all.h`.

6.226.2.5 [u_int8_t scsi_reserve::unused](#)[2]

Definition at line 336 of file `scsi_all.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.227 scsi_reserve_release_unit Struct Reference

```
#include <scsi_sa.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t lun_thirdparty](#)
- [u_int8_t reserved](#) [3]
- [u_int8_t control](#)

6.227.1 Detailed Description

Definition at line 118 of file `scsi_sa.h`.

6.227.2 Field Documentation

6.227.2.1 [u_int8_t scsi_reserve_release_unit::control](#)

Definition at line 127 of file `scsi_sa.h`.

6.227.2.2 [u_int8_t scsi_reserve_release_unit::lun_thirdparty](#)

Definition at line 121 of file `scsi_sa.h`.

6.227.2.3 [u_int8_t scsi_reserve_release_unit::opcode](#)

Definition at line 120 of file `scsi_sa.h`.

6.227.2.4 [u_int8_t scsi_reserve_release_unit::reserved](#)[3]

Definition at line 126 of file `scsi_sa.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_sa.h`

6.228 scsi_rewind Struct Reference

```
#include <scsi_sa.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t immediate](#)
- [u_int8_t reserved](#) [3]
- [u_int8_t control](#)

6.228.1 Detailed Description

Definition at line 77 of file `scsi_sa.h`.

6.228.2 Field Documentation

6.228.2.1 [u_int8_t scsi_rewind::control](#)

Definition at line 83 of file `scsi_sa.h`.

6.228.2.2 [u_int8_t scsi_rewind::immediate](#)

Definition at line 80 of file `scsi_sa.h`.

6.228.2.3 [u_int8_t scsi_rewind::opcode](#)

Definition at line 79 of file `scsi_sa.h`.

6.228.2.4 [u_int8_t scsi_rewind::reserved](#)[3]

Definition at line 82 of file `scsi_sa.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_sa.h`

6.229 scsi_rezero_unit Struct Reference

```
#include <scsi_da.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t byte2](#)
- [u_int8_t reserved](#) [3]
- [u_int8_t control](#)

6.229.1 Detailed Description

Definition at line 57 of file `scsi_da.h`.

6.229.2 Field Documentation

6.229.2.1 [u_int8_t scsi_rezero_unit::byte2](#)

Definition at line 61 of file `scsi_da.h`.

6.229.2.2 [u_int8_t scsi_rezero_unit::control](#)

Definition at line 63 of file `scsi_da.h`.

6.229.2.3 [u_int8_t scsi_rezero_unit::opcode](#)

Definition at line 59 of file `scsi_da.h`.

6.229.2.4 [u_int8_t scsi_rezero_unit::reserved](#)[3]

Definition at line 62 of file `scsi_da.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_da.h`

6.230 scsi_rw_10 Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t byte2](#)
- [u_int8_t addr](#) [4]
- [u_int8_t reserved](#)
- [u_int8_t length](#) [2]
- [u_int8_t control](#)

6.230.1 Detailed Description

Definition at line 418 of file `scsi_all.h`.

6.230.2 Field Documentation

6.230.2.1 [u_int8_t scsi_rw_10::addr](#)[4]

Definition at line 427 of file `scsi_all.h`.

Referenced by `cmd6workaround()`, and `scsi_read_write()`.

6.230.2.2 [u_int8_t scsi_rw_10::byte2](#)

Definition at line 426 of file `scsi_all.h`.

Referenced by `cmd6workaround()`, and `scsi_read_write()`.

6.230.2.3 [u_int8_t scsi_rw_10::control](#)

Definition at line 430 of file `scsi_all.h`.

Referenced by `cmd6workaround()`, and `scsi_read_write()`.

6.230.2.4 [u_int8_t scsi_rw_10::length](#)[2]

Definition at line 429 of file `scsi_all.h`.

Referenced by `cmd6workaround()`, and `scsi_read_write()`.

6.230.2.5 [u_int8_t scsi_rw_10::opcode](#)

Definition at line 420 of file `scsi_all.h`.

Referenced by `cmd6workaround()`, and `scsi_read_write()`.

6.230.2.6 u_int8_t scsi_rw_10::reserved

Definition at line 428 of file `scsi_all.h`.

Referenced by `cmd6workaround()`, and `scsi_read_write()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.231 scsi_rw_12 Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t byte2](#)
- [u_int8_t addr](#) [4]
- [u_int8_t length](#) [4]
- [u_int8_t reserved](#)
- [u_int8_t control](#)

6.231.1 Detailed Description

Definition at line 433 of file `scsi_all.h`.

6.231.2 Field Documentation

6.231.2.1 [u_int8_t scsi_rw_12::addr](#)[4]

Definition at line 440 of file `scsi_all.h`.

Referenced by `scsi_read_write()`.

6.231.2.2 [u_int8_t scsi_rw_12::byte2](#)

Definition at line 439 of file `scsi_all.h`.

Referenced by `scsi_read_write()`.

6.231.2.3 [u_int8_t scsi_rw_12::control](#)

Definition at line 443 of file `scsi_all.h`.

Referenced by `scsi_read_write()`.

6.231.2.4 [u_int8_t scsi_rw_12::length](#)[4]

Definition at line 441 of file `scsi_all.h`.

Referenced by `scsi_read_write()`.

6.231.2.5 [u_int8_t scsi_rw_12::opcode](#)

Definition at line 435 of file `scsi_all.h`.

Referenced by `scsi_read_write()`.

6.231.2.6 u_int8_t scsi_rw_12::reserved

Definition at line 442 of file `scsi_all.h`.

Referenced by `scsi_read_write()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.232 scsi_rw_16 Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t byte2](#)
- [u_int8_t addr](#) [8]
- [u_int8_t length](#) [4]
- [u_int8_t reserved](#)
- [u_int8_t control](#)

6.232.1 Detailed Description

Definition at line 446 of file `scsi_all.h`.

6.232.2 Field Documentation

6.232.2.1 [u_int8_t scsi_rw_16::addr](#)[8]

Definition at line 453 of file `scsi_all.h`.

Referenced by `scsi_read_write()`.

6.232.2.2 [u_int8_t scsi_rw_16::byte2](#)

Definition at line 452 of file `scsi_all.h`.

Referenced by `scsi_read_write()`.

6.232.2.3 [u_int8_t scsi_rw_16::control](#)

Definition at line 456 of file `scsi_all.h`.

Referenced by `scsi_read_write()`.

6.232.2.4 [u_int8_t scsi_rw_16::length](#)[4]

Definition at line 454 of file `scsi_all.h`.

Referenced by `scsi_read_write()`.

6.232.2.5 [u_int8_t scsi_rw_16::opcode](#)

Definition at line 448 of file `scsi_all.h`.

Referenced by `scsi_read_write()`.

6.232.2.6 u_int8_t scsi_rw_16::reserved

Definition at line 455 of file `scsi_all.h`.

Referenced by `scsi_read_write()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.233 scsi_rw_6 Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t addr \[3\]](#)
- [u_int8_t length](#)
- [u_int8_t control](#)

6.233.1 Detailed Description

Definition at line 408 of file `scsi_all.h`.

6.233.2 Field Documentation

6.233.2.1 [u_int8_t scsi_rw_6::addr\[3\]](#)

Definition at line 411 of file `scsi_all.h`.

Referenced by `cmd6workaround()`, and `scsi_read_write()`.

6.233.2.2 [u_int8_t scsi_rw_6::control](#)

Definition at line 415 of file `scsi_all.h`.

Referenced by `cmd6workaround()`, and `scsi_read_write()`.

6.233.2.3 [u_int8_t scsi_rw_6::length](#)

Definition at line 414 of file `scsi_all.h`.

Referenced by `cmd6workaround()`, and `scsi_read_write()`.

6.233.2.4 [u_int8_t scsi_rw_6::opcode](#)

Definition at line 410 of file `scsi_all.h`.

Referenced by `cmd6workaround()`, and `scsi_read_write()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.234 scsi_sa_rw Struct Reference

```
#include <scsi_sa.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t sli_fixed](#)
- [u_int8_t length \[3\]](#)
- [u_int8_t control](#)

6.234.1 Detailed Description

Definition at line 54 of file `scsi_sa.h`.

6.234.2 Field Documentation

6.234.2.1 [u_int8_t scsi_sa_rw::control](#)

Definition at line 61 of file `scsi_sa.h`.

Referenced by `scsi_sa_read_write()`.

6.234.2.2 [u_int8_t scsi_sa_rw::length\[3\]](#)

Definition at line 60 of file `scsi_sa.h`.

Referenced by `scsi_sa_read_write()`.

6.234.2.3 [u_int8_t scsi_sa_rw::opcode](#)

Definition at line 56 of file `scsi_sa.h`.

Referenced by `scsi_sa_read_write()`.

6.234.2.4 [u_int8_t scsi_sa_rw::sli_fixed](#)

Definition at line 57 of file `scsi_sa.h`.

Referenced by `scsi_sa_read_write()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_sa.h`

6.235 scsi_send_diag Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t byte2](#)
- [u_int8_t unused](#) [1]
- [u_int8_t paramlen](#) [2]
- [u_int8_t control](#)

6.235.1 Detailed Description

Definition at line 131 of file [scsi_all.h](#).

6.235.2 Field Documentation

6.235.2.1 [u_int8_t scsi_send_diag::byte2](#)

Definition at line 134 of file [scsi_all.h](#).

6.235.2.2 [u_int8_t scsi_send_diag::control](#)

Definition at line 141 of file [scsi_all.h](#).

6.235.2.3 [u_int8_t scsi_send_diag::opcode](#)

Definition at line 133 of file [scsi_all.h](#).

6.235.2.4 [u_int8_t scsi_send_diag::paramlen](#)[2]

Definition at line 140 of file [scsi_all.h](#).

6.235.2.5 [u_int8_t scsi_send_diag::unused](#)[1]

Definition at line 139 of file [scsi_all.h](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_all.h](#)

6.236 scsi_send_key Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t reserved](#) [7]
- [u_int8_t param_len](#) [2]
- [u_int8_t agid_keyformat](#)
- [u_int8_t control](#)

6.236.1 Detailed Description

Definition at line 208 of file `scsi_cd.h`.

6.236.2 Field Documentation

6.236.2.1 [u_int8_t scsi_send_key::agid_keyformat](#)

Definition at line 213 of file `scsi_cd.h`.

6.236.2.2 [u_int8_t scsi_send_key::control](#)

Definition at line 214 of file `scsi_cd.h`.

6.236.2.3 [u_int8_t scsi_send_key::opcode](#)

Definition at line 210 of file `scsi_cd.h`.

6.236.2.4 [u_int8_t scsi_send_key::param_len](#)[2]

Definition at line 212 of file `scsi_cd.h`.

6.236.2.5 [u_int8_t scsi_send_key::reserved](#)[7]

Definition at line 211 of file `scsi_cd.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.h`

6.237 scsi_send_key_data_rpc Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- [u_int8_t data_len](#) [2]
- [u_int8_t reserved0](#) [2]
- [u_int8_t region_code](#)
- [u_int8_t reserved1](#) [3]

6.237.1 Detailed Description

Definition at line 347 of file `scsi_cd.h`.

6.237.2 Field Documentation

6.237.2.1 [u_int8_t scsi_send_key_data_rpc::data_len](#)[2]

Definition at line 349 of file `scsi_cd.h`.

Referenced by `cdsendkey()`.

6.237.2.2 [u_int8_t scsi_send_key_data_rpc::region_code](#)

Definition at line 351 of file `scsi_cd.h`.

Referenced by `cdsendkey()`.

6.237.2.3 [u_int8_t scsi_send_key_data_rpc::reserved0](#)[2]

Definition at line 350 of file `scsi_cd.h`.

6.237.2.4 [u_int8_t scsi_send_key_data_rpc::reserved1](#)[3]

Definition at line 352 of file `scsi_cd.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_cd.h`

6.238 scsi_send_receive Struct Reference

```
#include <scsi_pt.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t byte2](#)
- [u_int8_t xfer_len](#) [3]
- [u_int8_t control](#)

6.238.1 Detailed Description

Definition at line 34 of file `scsi_pt.h`.

6.238.2 Field Documentation

6.238.2.1 [u_int8_t scsi_send_receive::byte2](#)

Definition at line 37 of file `scsi_pt.h`.

Referenced by `scsi_send_receive()`.

6.238.2.2 [u_int8_t scsi_send_receive::control](#)

Definition at line 39 of file `scsi_pt.h`.

Referenced by `scsi_send_receive()`.

6.238.2.3 [u_int8_t scsi_send_receive::opcode](#)

Definition at line 36 of file `scsi_pt.h`.

Referenced by `scsi_send_receive()`.

6.238.2.4 [u_int8_t scsi_send_receive::xfer_len](#)[3]

Definition at line 38 of file `scsi_pt.h`.

Referenced by `scsi_send_receive()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_pt.h`

6.239 scsi_send_volume_tag Struct Reference

```
#include <scsi_ch.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t byte2](#)
- [u_int8_t ea \[2\]](#)
- [u_int8_t reserved2](#)
- [u_int8_t sac](#)
- [u_int8_t reserved4 \[2\]](#)
- [u_int8_t pll \[2\]](#)
- [u_int8_t reserved5](#)
- [u_int8_t control](#)

6.239.1 Detailed Description

Definition at line 272 of file scsi_ch.h.

6.239.2 Field Documentation

6.239.2.1 [u_int8_t scsi_send_volume_tag::byte2](#)

Definition at line 275 of file scsi_ch.h.

6.239.2.2 [u_int8_t scsi_send_volume_tag::control](#)

Definition at line 290 of file scsi_ch.h.

6.239.2.3 [u_int8_t scsi_send_volume_tag::ea\[2\]](#)

Definition at line 276 of file scsi_ch.h.

6.239.2.4 [u_int8_t scsi_send_volume_tag::opcode](#)

Definition at line 273 of file scsi_ch.h.

6.239.2.5 [u_int8_t scsi_send_volume_tag::pll\[2\]](#)

Definition at line 288 of file scsi_ch.h.

6.239.2.6 [u_int8_t scsi_send_volume_tag::reserved2](#)

Definition at line 277 of file scsi_ch.h.

6.239.2.7 `u_int8_t scsi_send_volume_tag::reserved4[2]`

Definition at line 287 of file `scsi_ch.h`.

6.239.2.8 `u_int8_t scsi_send_volume_tag::reserved5`

Definition at line 289 of file `scsi_ch.h`.

6.239.2.9 `u_int8_t scsi_send_volume_tag::sac`

Definition at line 278 of file `scsi_ch.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_ch.h`

6.240 scsi_send_volume_tag_parameters Struct Reference

```
#include <scsi_ch.h>
```

Data Fields

- `u_int8_t vitf` [32]
- `u_int8_t reserved1` [2]
- `u_int8_t minvsn` [2]
- `u_int8_t reserved2` [2]
- `u_int8_t maxvsn` [2]

6.240.1 Detailed Description

Definition at line 297 of file `scsi_ch.h`.

6.240.2 Field Documentation

6.240.2.1 `u_int8_t scsi_send_volume_tag_parameters::maxvsn`[2]

Definition at line 302 of file `scsi_ch.h`.

6.240.2.2 `u_int8_t scsi_send_volume_tag_parameters::minvsn`[2]

Definition at line 300 of file `scsi_ch.h`.

6.240.2.3 `u_int8_t scsi_send_volume_tag_parameters::reserved1`[2]

Definition at line 299 of file `scsi_ch.h`.

6.240.2.4 `u_int8_t scsi_send_volume_tag_parameters::reserved2`[2]

Definition at line 301 of file `scsi_ch.h`.

6.240.2.5 `u_int8_t scsi_send_volume_tag_parameters::vitf`[32]

Definition at line 298 of file `scsi_ch.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_ch.h`

6.241 `scsi_sense` Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- `u_int8_t opcode`
- `u_int8_t byte2`
- `u_int8_t unused` [2]
- `u_int8_t length`
- `u_int8_t control`

6.241.1 Detailed Description

Definition at line 144 of file `scsi_all.h`.

6.241.2 Field Documentation

6.241.2.1 `u_int8_t scsi_sense::byte2`

Definition at line 147 of file `scsi_all.h`.

6.241.2.2 `u_int8_t scsi_sense::control`

Definition at line 150 of file `scsi_all.h`.

6.241.2.3 `u_int8_t scsi_sense::length`

Definition at line 149 of file `scsi_all.h`.

6.241.2.4 `u_int8_t scsi_sense::opcode`

Definition at line 146 of file `scsi_all.h`.

6.241.2.5 `u_int8_t scsi_sense::unused`[2]

Definition at line 148 of file `scsi_all.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.242 scsi_sense_data Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- [u_int8_t error_code](#)
- [u_int8_t segment](#)
- [u_int8_t flags](#)
- [u_int8_t info](#) [4]
- [u_int8_t extra_len](#)
- [u_int8_t cmd_spec_info](#) [4]
- [u_int8_t add_sense_code](#)
- [u_int8_t add_sense_code_qual](#)
- [u_int8_t fru](#)
- [u_int8_t sense_key_spec](#) [3]
- [u_int8_t extra_bytes](#) [14]

6.242.1 Detailed Description

Definition at line 751 of file `scsi_all.h`.

6.242.2 Field Documentation

6.242.2.1 [u_int8_t scsi_sense_data::add_sense_code](#)

Definition at line 783 of file `scsi_all.h`.

Referenced by `scsi_extract_sense()`, `scsi_low_setup_done()`, and `scsi_sense_sbuf()`.

6.242.2.2 [u_int8_t scsi_sense_data::add_sense_code_qual](#)

Definition at line 784 of file `scsi_all.h`.

Referenced by `scsi_extract_sense()`, `scsi_low_setup_done()`, and `scsi_sense_sbuf()`.

6.242.2.3 [u_int8_t scsi_sense_data::cmd_spec_info](#)[4]

Definition at line 782 of file `scsi_all.h`.

Referenced by `scsi_sense_sbuf()`.

6.242.2.4 [u_int8_t scsi_sense_data::error_code](#)

Definition at line 753 of file `scsi_all.h`.

Referenced by `camperiphdone()`, `cddone()`, `dadone()`, `saerror()`, `scsi_extract_sense()`, `scsi_low_setup_done()`, and `scsi_sense_sbuf()`.

6.242.2.5 `u_int8_t scsi_sense_data::extra_bytes[14]`

Definition at line 792 of file `scsi_all.h`.

6.242.2.6 `u_int8_t scsi_sense_data::extra_len`

Definition at line 781 of file `scsi_all.h`.

Referenced by `saerror()`, `scsi_extract_sense()`, and `scsi_sense_sbuf()`.

6.242.2.7 `u_int8_t scsi_sense_data::flags`

Definition at line 759 of file `scsi_all.h`.

Referenced by `camperiphdone()`, `saerror()`, `scsi_extract_sense()`, `scsi_low_setup_done()`, and `scsi_sense_sbuf()`.

6.242.2.8 `u_int8_t scsi_sense_data::fru`

Definition at line 785 of file `scsi_all.h`.

Referenced by `scsi_sense_sbuf()`.

6.242.2.9 `u_int8_t scsi_sense_data::info[4]`

Definition at line 780 of file `scsi_all.h`.

Referenced by `saerror()`, and `scsi_sense_sbuf()`.

6.242.2.10 `u_int8_t scsi_sense_data::segment`

Definition at line 758 of file `scsi_all.h`.

Referenced by `scsi_low_setup_done()`.

6.242.2.11 `u_int8_t scsi_sense_data::sense_key_spec[3]`

Definition at line 786 of file `scsi_all.h`.

Referenced by `scsi_sense_sbuf()`.

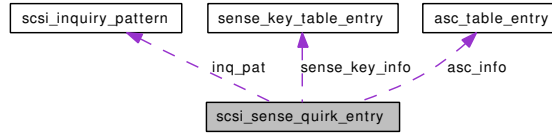
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.243 scsi_sense_quirk_entry Struct Reference

```
#include <scsi_all.h>
```

Collaboration diagram for `scsi_sense_quirk_entry`:



Data Fields

- `scsi_inquiry_pattern` `inq_pat`
- `int` `num_sense_keys`
- `int` `num_ascs`
- `sense_key_table_entry` * `sense_key_info`
- `asc_table_entry` * `asc_info`

6.243.1 Detailed Description

Definition at line 864 of file `scsi_all.h`.

6.243.2 Field Documentation

6.243.2.1 `struct asc_table_entry* scsi_sense_quirk_entry::asc_info`

Definition at line 869 of file `scsi_all.h`.

Referenced by `fetchtableentries()`.

6.243.2.2 `struct scsi_inquiry_pattern scsi_sense_quirk_entry::inq_pat`

Definition at line 865 of file `scsi_all.h`.

6.243.2.3 `int scsi_sense_quirk_entry::num_ascs`

Definition at line 867 of file `scsi_all.h`.

Referenced by `fetchtableentries()`.

6.243.2.4 `int scsi_sense_quirk_entry::num_sense_keys`

Definition at line 866 of file `scsi_all.h`.

Referenced by `fetchtableentries()`.

6.243.2.5 struct [sense_key_table_entry](#)* [scsi_sense_quirk_entry::sense_key_info](#)

Definition at line 868 of file [scsi_all.h](#).

Referenced by [fetchtableentries\(\)](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_all.h](#)

6.244 scsi_set_speed Struct Reference

```
#include <scsi_cd.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t byte2](#)
- [u_int8_t readspeed](#) [2]
- [u_int8_t writespeed](#) [2]
- [u_int8_t reserved](#) [5]
- [u_int8_t control](#)

6.244.1 Detailed Description

Definition at line 172 of file [scsi_cd.h](#).

6.244.2 Field Documentation

6.244.2.1 [u_int8_t scsi_set_speed::byte2](#)

Definition at line 175 of file [scsi_cd.h](#).

6.244.2.2 [u_int8_t scsi_set_speed::control](#)

Definition at line 179 of file [scsi_cd.h](#).

6.244.2.3 [u_int8_t scsi_set_speed::opcode](#)

Definition at line 174 of file [scsi_cd.h](#).

6.244.2.4 [u_int8_t scsi_set_speed::readspeed](#)[2]

Definition at line 176 of file [scsi_cd.h](#).

6.244.2.5 [u_int8_t scsi_set_speed::reserved](#)[5]

Definition at line 178 of file [scsi_cd.h](#).

6.244.2.6 [u_int8_t scsi_set_speed::writespeed](#)[2]

Definition at line 177 of file [scsi_cd.h](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_cd.h](#)

6.245 `scsi_space` Struct Reference

```
#include <scsi_sa.h>
```

Data Fields

- `u_int8_t opcode`
- `u_int8_t code`
- `u_int8_t count` [3]
- `u_int8_t control`

6.245.1 Detailed Description

Definition at line 95 of file `scsi_sa.h`.

6.245.2 Field Documentation

6.245.2.1 `u_int8_t scsi_space::code`

Definition at line 98 of file `scsi_sa.h`.

Referenced by `scsi_space()`.

6.245.2.2 `u_int8_t scsi_space::control`

Definition at line 101 of file `scsi_sa.h`.

Referenced by `scsi_space()`.

6.245.2.3 `u_int8_t scsi_space::count`[3]

Definition at line 100 of file `scsi_sa.h`.

Referenced by `scsi_space()`.

6.245.2.4 `u_int8_t scsi_space::opcode`

Definition at line 97 of file `scsi_sa.h`.

Referenced by `scsi_space()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_sa.h`

6.246 scsi_start_stop_unit Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t byte2](#)
- [u_int8_t reserved](#) [2]
- [u_int8_t how](#)
- [u_int8_t control](#)

6.246.1 Detailed Description

Definition at line 459 of file `scsi_all.h`.

6.246.2 Field Documentation

6.246.2.1 [u_int8_t scsi_start_stop_unit::byte2](#)

Definition at line 462 of file `scsi_all.h`.

6.246.2.2 [u_int8_t scsi_start_stop_unit::control](#)

Definition at line 468 of file `scsi_all.h`.

6.246.2.3 [u_int8_t scsi_start_stop_unit::how](#)

Definition at line 465 of file `scsi_all.h`.

Referenced by `camperiphdone()`.

6.246.2.4 [u_int8_t scsi_start_stop_unit::opcode](#)

Definition at line 461 of file `scsi_all.h`.

Referenced by `camperiphdone()`.

6.246.2.5 [u_int8_t scsi_start_stop_unit::reserved](#)[2]

Definition at line 464 of file `scsi_all.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.247 scsi_static_inquiry_pattern Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- `u_int8_t type`
- `u_int8_t media_type`
- `char vendor [SID_VENDOR_SIZE+1]`
- `char product [SID_PRODUCT_SIZE+1]`
- `char revision [SID_REVISION_SIZE+1]`

6.247.1 Detailed Description

Definition at line 856 of file `scsi_all.h`.

6.247.2 Field Documentation

6.247.2.1 `u_int8_t scsi_static_inquiry_pattern::media_type`

Definition at line 858 of file `scsi_all.h`.

Referenced by `scsi_static_inquiry_match()`.

6.247.2.2 `char scsi_static_inquiry_pattern::product[SID_PRODUCT_SIZE+1]`

Definition at line 860 of file `scsi_all.h`.

Referenced by `scsi_static_inquiry_match()`.

6.247.2.3 `char scsi_static_inquiry_pattern::revision[SID_REVISION_SIZE+1]`

Definition at line 861 of file `scsi_all.h`.

Referenced by `scsi_static_inquiry_match()`.

6.247.2.4 `u_int8_t scsi_static_inquiry_pattern::type`

Definition at line 857 of file `scsi_all.h`.

Referenced by `scsi_static_inquiry_match()`.

6.247.2.5 `char scsi_static_inquiry_pattern::vendor[SID_VENDOR_SIZE+1]`

Definition at line 859 of file `scsi_all.h`.

Referenced by `scsi_static_inquiry_match()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.248 scsi_status_iu_header Struct Reference

```
#include <scsi_iu.h>
```

Data Fields

- [u_int8_t reserved](#) [2]
- [u_int8_t flags](#)
- [u_int8_t status](#)
- [u_int8_t sense_length](#) [4]
- [u_int8_t pkt_failures_length](#) [4]
- [u_int8_t pkt_failures](#) [1]

6.248.1 Detailed Description

Definition at line 8 of file [scsi_iu.h](#).

6.248.2 Field Documentation

6.248.2.1 [u_int8_t scsi_status_iu_header::flags](#)

Definition at line 11 of file [scsi_iu.h](#).

6.248.2.2 [u_int8_t scsi_status_iu_header::pkt_failures](#)[1]

Definition at line 17 of file [scsi_iu.h](#).

6.248.2.3 [u_int8_t scsi_status_iu_header::pkt_failures_length](#)[4]

Definition at line 16 of file [scsi_iu.h](#).

6.248.2.4 [u_int8_t scsi_status_iu_header::reserved](#)[2]

Definition at line 10 of file [scsi_iu.h](#).

6.248.2.5 [u_int8_t scsi_status_iu_header::sense_length](#)[4]

Definition at line 15 of file [scsi_iu.h](#).

6.248.2.6 [u_int8_t scsi_status_iu_header::status](#)

Definition at line 14 of file [scsi_iu.h](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_iu.h](#)

6.249 `scsi_sync_cache` Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- `u_int8_t opcode`
- `u_int8_t byte2`
- `u_int8_t begin_lba` [4]
- `u_int8_t reserved`
- `u_int8_t lb_count` [2]
- `u_int8_t control`

6.249.1 Detailed Description

Definition at line 361 of file `scsi_all.h`.

6.249.2 Field Documentation

6.249.2.1 `u_int8_t scsi_sync_cache::begin_lba`[4]

Definition at line 365 of file `scsi_all.h`.

6.249.2.2 `u_int8_t scsi_sync_cache::byte2`

Definition at line 364 of file `scsi_all.h`.

6.249.2.3 `u_int8_t scsi_sync_cache::control`

Definition at line 368 of file `scsi_all.h`.

6.249.2.4 `u_int8_t scsi_sync_cache::lb_count`[2]

Definition at line 367 of file `scsi_all.h`.

6.249.2.5 `u_int8_t scsi_sync_cache::opcode`

Definition at line 363 of file `scsi_all.h`.

6.249.2.6 `u_int8_t scsi_sync_cache::reserved`

Definition at line 366 of file `scsi_all.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.250 scsi_tape_locate Struct Reference

```
#include <scsi_sa.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t byte1](#)
- [u_int8_t reserved1](#)
- [u_int8_t blkaddr \[4\]](#)
- [u_int8_t reserved2](#)
- [u_int8_t partition](#)
- [u_int8_t control](#)

6.250.1 Detailed Description

Definition at line 248 of file `scsi_sa.h`.

6.250.2 Field Documentation

6.250.2.1 [u_int8_t scsi_tape_locate::blkaddr\[4\]](#)

Definition at line 255 of file `scsi_sa.h`.

6.250.2.2 [u_int8_t scsi_tape_locate::byte1](#)

Definition at line 250 of file `scsi_sa.h`.

6.250.2.3 [u_int8_t scsi_tape_locate::control](#)

Definition at line 258 of file `scsi_sa.h`.

6.250.2.4 [u_int8_t scsi_tape_locate::opcode](#)

Definition at line 249 of file `scsi_sa.h`.

6.250.2.5 [u_int8_t scsi_tape_locate::partition](#)

Definition at line 257 of file `scsi_sa.h`.

6.250.2.6 [u_int8_t scsi_tape_locate::reserved1](#)

Definition at line 254 of file `scsi_sa.h`.

6.250.2.7 `u_int8_t scsi_tape_locate::reserved2`

Definition at line 256 of file `scsi_sa.h`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_sa.h](#)

6.251 scsi_tape_position_data Struct Reference

```
#include <scsi_sa.h>
```

Data Fields

- [u_int8_t flags](#)
- [u_int8_t partition](#)
- [u_int8_t reserved](#) [2]
- [u_int8_t firstblk](#) [4]
- [u_int8_t lastblk](#) [4]
- [u_int8_t reserved2](#)
- [u_int8_t nbufblk](#) [3]
- [u_int8_t nbufbyte](#) [4]

6.251.1 Detailed Description

Definition at line 230 of file `scsi_sa.h`.

6.251.2 Field Documentation

6.251.2.1 [u_int8_t scsi_tape_position_data::firstblk](#)[4]

Definition at line 241 of file `scsi_sa.h`.

Referenced by `sardpos()`.

6.251.2.2 [u_int8_t scsi_tape_position_data::flags](#)

Definition at line 231 of file `scsi_sa.h`.

Referenced by `sardpos()`.

6.251.2.3 [u_int8_t scsi_tape_position_data::lastblk](#)[4]

Definition at line 242 of file `scsi_sa.h`.

6.251.2.4 [u_int8_t scsi_tape_position_data::nbufblk](#)[3]

Definition at line 244 of file `scsi_sa.h`.

6.251.2.5 [u_int8_t scsi_tape_position_data::nbufbyte](#)[4]

Definition at line 245 of file `scsi_sa.h`.

6.251.2.6 [u_int8_t scsi_tape_position_data::partition](#)

Definition at line 239 of file `scsi_sa.h`.

6.251.2.7 `u_int8_t scsi_tape_position_data::reserved[2]`

Definition at line 240 of file `scsi_sa.h`.

6.251.2.8 `u_int8_t scsi_tape_position_data::reserved2`

Definition at line 243 of file `scsi_sa.h`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_sa.h](#)

6.252 scsi_tape_read_position Struct Reference

```
#include <scsi_sa.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t byte1](#)
- [u_int8_t reserved](#) [8]

6.252.1 Detailed Description

Definition at line 224 of file [scsi_sa.h](#).

6.252.2 Field Documentation

6.252.2.1 [u_int8_t scsi_tape_read_position::byte1](#)

Definition at line 226 of file [scsi_sa.h](#).

6.252.2.2 [u_int8_t scsi_tape_read_position::opcode](#)

Definition at line 225 of file [scsi_sa.h](#).

6.252.2.3 [u_int8_t scsi_tape_read_position::reserved](#)[8]

Definition at line 227 of file [scsi_sa.h](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_sa.h](#)

6.253 `scsi_test_unit_ready` Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- `u_int8_t opcode`
- `u_int8_t byte2`
- `u_int8_t unused` [3]
- `u_int8_t control`

6.253.1 Detailed Description

Definition at line 123 of file `scsi_all.h`.

6.253.2 Field Documentation

6.253.2.1 `u_int8_t scsi_test_unit_ready::byte2`

Definition at line 126 of file `scsi_all.h`.

6.253.2.2 `u_int8_t scsi_test_unit_ready::control`

Definition at line 128 of file `scsi_all.h`.

6.253.2.3 `u_int8_t scsi_test_unit_ready::opcode`

Definition at line 125 of file `scsi_all.h`.

6.253.2.4 `u_int8_t scsi_test_unit_ready::unused`[3]

Definition at line 127 of file `scsi_all.h`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.254 scsi_verify Struct Reference

```
#include <scsi_da.h>
```

Data Fields

- [uint8_t opcode](#)
- [uint8_t byte2](#)
- [uint8_t addr](#) [4]
- [uint8_t reserved0](#) [1]
- [uint8_t len](#) [2]
- [uint8_t reserved1](#) [3]

6.254.1 Detailed Description

Definition at line 198 of file [scsi_da.h](#).

6.254.2 Field Documentation

6.254.2.1 [uint8_t scsi_verify::addr](#)[4]

Definition at line 206 of file [scsi_da.h](#).

6.254.2.2 [uint8_t scsi_verify::byte2](#)

Definition at line 201 of file [scsi_da.h](#).

6.254.2.3 [uint8_t scsi_verify::len](#)[2]

Definition at line 208 of file [scsi_da.h](#).

6.254.2.4 [uint8_t scsi_verify::opcode](#)

Definition at line 200 of file [scsi_da.h](#).

6.254.2.5 [uint8_t scsi_verify::reserved0](#)[1]

Definition at line 207 of file [scsi_da.h](#).

6.254.2.6 [uint8_t scsi_verify::reserved1](#)[3]

Definition at line 209 of file [scsi_da.h](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_da.h](#)

6.255 scsi_vpd_unit_serial_number Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- [u_int8_t device](#)
- [u_int8_t page_code](#)
- [u_int8_t reserved](#)
- [u_int8_t length](#)
- [u_int8_t serial_num](#) [SVPD_SERIAL_NUM_SIZE]

6.255.1 Detailed Description

Definition at line 666 of file `scsi_all.h`.

6.255.2 Field Documentation

6.255.2.1 [u_int8_t scsi_vpd_unit_serial_number::device](#)

Definition at line 668 of file `scsi_all.h`.

6.255.2.2 [u_int8_t scsi_vpd_unit_serial_number::length](#)

Definition at line 672 of file `scsi_all.h`.

Referenced by `probedone()`.

6.255.2.3 [u_int8_t scsi_vpd_unit_serial_number::page_code](#)

Definition at line 669 of file `scsi_all.h`.

6.255.2.4 [u_int8_t scsi_vpd_unit_serial_number::reserved](#)

Definition at line 671 of file `scsi_all.h`.

6.255.2.5 [u_int8_t scsi_vpd_unit_serial_number::serial_num](#)[SVPD_SERIAL_NUM_SIZE]

Definition at line 674 of file `scsi_all.h`.

Referenced by `probedone()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.256 scsi_write_and_verify Struct Reference

```
#include <scsi_da.h>
```

Data Fields

- [uint8_t opcode](#)
- [uint8_t byte2](#)
- [uint8_t addr](#) [4]
- [uint8_t reserved0](#) [1]
- [uint8_t len](#) [2]
- [uint8_t reserved1](#) [3]

6.256.1 Detailed Description

Definition at line 212 of file [scsi_da.h](#).

6.256.2 Field Documentation

6.256.2.1 [uint8_t scsi_write_and_verify::addr](#)[4]

Definition at line 220 of file [scsi_da.h](#).

6.256.2.2 [uint8_t scsi_write_and_verify::byte2](#)

Definition at line 215 of file [scsi_da.h](#).

6.256.2.3 [uint8_t scsi_write_and_verify::len](#)[2]

Definition at line 222 of file [scsi_da.h](#).

6.256.2.4 [uint8_t scsi_write_and_verify::opcode](#)

Definition at line 214 of file [scsi_da.h](#).

6.256.2.5 [uint8_t scsi_write_and_verify::reserved0](#)[1]

Definition at line 221 of file [scsi_da.h](#).

6.256.2.6 [uint8_t scsi_write_and_verify::reserved1](#)[3]

Definition at line 223 of file [scsi_da.h](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_da.h](#)

6.257 scsi_write_buffer Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t byte2](#)
- [u_int8_t buffer_id](#)
- [u_int8_t offset](#) [3]
- [u_int8_t length](#) [3]
- [u_int8_t control](#)

6.257.1 Detailed Description

Definition at line 398 of file [scsi_all.h](#).

6.257.2 Field Documentation

6.257.2.1 [u_int8_t scsi_write_buffer::buffer_id](#)

Definition at line 402 of file [scsi_all.h](#).

6.257.2.2 [u_int8_t scsi_write_buffer::byte2](#)

Definition at line 401 of file [scsi_all.h](#).

6.257.2.3 [u_int8_t scsi_write_buffer::control](#)

Definition at line 405 of file [scsi_all.h](#).

6.257.2.4 [u_int8_t scsi_write_buffer::length](#)[3]

Definition at line 404 of file [scsi_all.h](#).

6.257.2.5 [u_int8_t scsi_write_buffer::offset](#)[3]

Definition at line 403 of file [scsi_all.h](#).

6.257.2.6 [u_int8_t scsi_write_buffer::opcode](#)

Definition at line 400 of file [scsi_all.h](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_all.h](#)

6.258 scsi_write_filemarks Struct Reference

```
#include <scsi_sa.h>
```

Data Fields

- [u_int8_t opcode](#)
- [u_int8_t byte2](#)
- [u_int8_t num_marks](#) [3]
- [u_int8_t control](#)

6.258.1 Detailed Description

Definition at line 104 of file `scsi_sa.h`.

6.258.2 Field Documentation

6.258.2.1 [u_int8_t scsi_write_filemarks::byte2](#)

Definition at line 107 of file `scsi_sa.h`.

6.258.2.2 [u_int8_t scsi_write_filemarks::control](#)

Definition at line 111 of file `scsi_sa.h`.

6.258.2.3 [u_int8_t scsi_write_filemarks::num_marks](#)[3]

Definition at line 110 of file `scsi_sa.h`.

6.258.2.4 [u_int8_t scsi_write_filemarks::opcode](#)

Definition at line 106 of file `scsi_sa.h`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_sa.h](#)

6.259 `sense_key_table_entry` Struct Reference

```
#include <scsi_all.h>
```

Data Fields

- `u_int8_t` [sense_key](#)
- `u_int32_t` [action](#)
- `const char *` [desc](#)

6.259.1 Detailed Description

Definition at line 872 of file `scsi_all.h`.

6.259.2 Field Documentation

6.259.2.1 `u_int32_t` [sense_key_table_entry::action](#)

Definition at line 874 of file `scsi_all.h`.

Referenced by `scsi_error_action()`.

6.259.2.2 `const char*` [sense_key_table_entry::desc](#)

Definition at line 875 of file `scsi_all.h`.

Referenced by `scsi_sense_desc()`.

6.259.2.3 `u_int8_t` [sense_key_table_entry::sense_key](#)

Definition at line 873 of file `scsi_all.h`.

Referenced by `scsi_error_action()`, and `senseentrycomp()`.

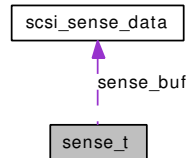
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_all.h`

6.260 sense_t Union Reference

```
#include <cam_ccb.h>
```

Collaboration diagram for sense_t:



Data Fields

- `u_int8_t * sense_ptr`
- `scsi_sense_data sense_buf`

6.260.1 Detailed Description

Definition at line 570 of file cam_ccb.h.

6.260.2 Field Documentation

6.260.2.1 struct `scsi_sense_data` `sense_t::sense_buf`

Definition at line 576 of file cam_ccb.h.

6.260.2.2 `u_int8_t*` `sense_t::sense_ptr`

Definition at line 571 of file cam_ccb.h.

The documentation for this union was generated from the following file:

- `/usr/src/sys/cam/cam_ccb.h`

6.261 ses_hlptxt Union Reference

```
#include <scsi_ses.h>
```

Data Fields

- unsigned int [obj_id](#)
- char [obj_text](#) [1]

6.261.1 Detailed Description

Definition at line 176 of file scsi_ses.h.

6.261.2 Field Documentation

6.261.2.1 unsigned int [ses_hlptxt::obj_id](#)

Definition at line 177 of file scsi_ses.h.

6.261.2.2 char [ses_hlptxt::obj_text](#)[1]

Definition at line 178 of file scsi_ses.h.

The documentation for this union was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_ses.h](#)

6.262 ses_object Struct Reference

```
#include <scsi_ses.h>
```

Data Fields

- unsigned int [obj_id](#)
- unsigned char [subencid](#)
- unsigned char [object_type](#)

6.262.1 Detailed Description

Definition at line 84 of file `scsi_ses.h`.

6.262.2 Field Documentation

6.262.2.1 unsigned int [ses_object::obj_id](#)

Definition at line 85 of file `scsi_ses.h`.

Referenced by `sesioctl()`.

6.262.2.2 unsigned char [ses_object::object_type](#)

Definition at line 87 of file `scsi_ses.h`.

Referenced by `sesioctl()`.

6.262.2.3 unsigned char [ses_object::subencid](#)

Definition at line 86 of file `scsi_ses.h`.

Referenced by `sesioctl()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_ses.h`

6.263 ses_objstat Struct Reference

```
#include <scsi_ses.h>
```

Data Fields

- unsigned int [obj_id](#)
- unsigned char [cstat](#) [4]

6.263.1 Detailed Description

Definition at line 125 of file [scsi_ses.h](#).

6.263.2 Field Documentation

6.263.2.1 unsigned char [ses_objstat::cstat](#)[4]

Definition at line 127 of file [scsi_ses.h](#).

Referenced by [safte_get_objstat\(\)](#), [safte_set_objstat\(\)](#), [ses_get_objstat\(\)](#), [ses_set_objstat\(\)](#), and [set_objstat_sel\(\)](#).

6.263.2.2 unsigned int [ses_objstat::obj_id](#)

Definition at line 126 of file [scsi_ses.h](#).

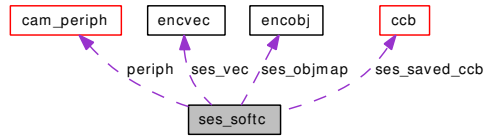
Referenced by [safte_get_objstat\(\)](#), [safte_set_objstat\(\)](#), [ses_get_objstat\(\)](#), [ses_set_objstat\(\)](#), [sesioctl\(\)](#), and [set_objstat_sel\(\)](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_ses.h](#)

6.264 ses_softc Struct Reference

Collaboration diagram for ses_softc:



Data Fields

- `enctyp` `ses_type`
- `encvec` `ses_vec`
- `void *` `ses_private`
- `encobj *` `ses_objmap`
- `u_int32_t` `ses_nobjects`
- `ses_encstat` `ses_encstat`
- `u_int8_t` `ses_flags`
- `ccb` `ses_saved_ccb`
- `cdev *` `ses_dev`
- `cam_periph *` `periph`

6.264.1 Detailed Description

Definition at line 141 of file `scsi_ses.c`.

6.264.2 Field Documentation

6.264.2.1 `struct cam_periph*` `ses_softc::periph`

Definition at line 151 of file `scsi_ses.c`.

Referenced by `ses_log()`, `ses_runcmd()`, `sescleanup()`, `sesclose()`, `seserror()`, `sesioctl()`, `sesoninvalidate()`, `sesopen()`, and `sesregister()`.

6.264.2.2 `struct cdev*` `ses_softc::ses_dev`

Definition at line 150 of file `scsi_ses.c`.

Referenced by `sescleanup()`, and `sesregister()`.

6.264.2.3 `ses_encstat` `ses_softc::ses_encstat`

Definition at line 147 of file `scsi_ses.c`.

Referenced by `saftc_get_objstat()`, `saftc_rdstat()`, `saftc_set_encstat()`, `saftc_softc_init()`, `ses_get_encstat()`, `ses_set_encstat()`, `ses_softc_init()`, and `sesioctl()`.

6.264.2.4 `u_int8_t ses_softc::ses_flags`

Definition at line 148 of file `scsi_ses.c`.

Referenced by `sesclose()`, `sesioctl()`, and `sesopen()`.

6.264.2.5 `u_int32_t ses_softc::ses_nobjects`

Definition at line 146 of file `scsi_ses.c`.

Referenced by `saft_e_rdstat()`, `saft_e_softc_init()`, `ses_getconfig()`, `ses_getputstat()`, `ses_softc_init()`, and `sesioctl()`.

6.264.2.6 `encobj* ses_softc::ses_objmap`

Definition at line 145 of file `scsi_ses.c`.

Referenced by `saft_e_get_objstat()`, `saft_e_rdstat()`, `saft_e_set_objstat()`, `saft_e_softc_init()`, `ses_get_objstat()`, `ses_getconfig()`, `ses_set_objstat()`, `ses_softc_init()`, `sesioctl()`, `set_objstat_sel()`, and `wrslot_stat()`.

6.264.2.7 `void* ses_softc::ses_private`

Definition at line 144 of file `scsi_ses.c`.

Referenced by `perf_slotop()`, `saft_e_getconfig()`, `saft_e_rdstat()`, `saft_e_set_encstat()`, `saft_e_set_objstat()`, `saft_e_softc_init()`, `ses_getconfig()`, `ses_getputstat()`, `ses_softc_init()`, `set_objstat_sel()`, `wrbuf16()`, and `wrslot_stat()`.

6.264.2.8 `union ccb ses_softc::ses_saved_ccb`

Definition at line 149 of file `scsi_ses.c`.

Referenced by `seserror()`.

6.264.2.9 `enctyp ses_softc::ses_type`

Definition at line 142 of file `scsi_ses.c`.

Referenced by `sesregister()`.

6.264.2.10 `encvec ses_softc::ses_vec`

Definition at line 143 of file `scsi_ses.c`.

Referenced by `sesioctl()`, `sesopen()`, and `sesregister()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_ses.c`

6.265 SesCfgHdr Struct Reference

Data Fields

- [uint32_t GenCode](#)
- [uint8_t Nsubenc](#)

6.265.1 Detailed Description

Definition at line 820 of file `scsi_ses.c`.

6.265.2 Field Documentation

6.265.2.1 [uint32_t SesCfgHdr::GenCode](#)

Definition at line 821 of file `scsi_ses.c`.

Referenced by `ses_cfghdr()`, and `ses_getconfig()`.

6.265.2.2 [uint8_t SesCfgHdr::Nsubenc](#)

Definition at line 822 of file `scsi_ses.c`.

Referenced by `ses_cfghdr()`, and `ses_getconfig()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_ses.c`

6.266 SesComStat Struct Reference

Data Fields

- [uint8_t comstatus](#)
- [uint8_t comstat \[3\]](#)

6.266.1 Detailed Description

Definition at line 846 of file `scsi_ses.c`.

6.266.2 Field Documentation

6.266.2.1 [uint8_t SesComStat::comstat\[3\]](#)

Definition at line 848 of file `scsi_ses.c`.

Referenced by `ses_decode()`, `ses_encode()`, `ses_get_objstat()`, and `ses_set_objstat()`.

6.266.2.2 [uint8_t SesComStat::comstatus](#)

Definition at line 847 of file `scsi_ses.c`.

Referenced by `ses_decode()`, `ses_encode()`, `ses_get_encstat()`, `ses_get_objstat()`, `ses_set_encstat()`, and `ses_set_objstat()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_ses.c`

6.267 SesEncDesc Struct Reference

Data Fields

- `uint8_t encWWN` [8]
- `uint8_t encVid` [8]
- `uint8_t encPid` [16]
- `uint8_t encRev` [4]
- `uint8_t encVen` [1]

6.267.1 Detailed Description

Definition at line 831 of file `scsi_ses.c`.

6.267.2 Field Documentation

6.267.2.1 `uint8_t SesEncDesc::encPid`[16]

Definition at line 834 of file `scsi_ses.c`.

6.267.2.2 `uint8_t SesEncDesc::encRev`[4]

Definition at line 835 of file `scsi_ses.c`.

6.267.2.3 `uint8_t SesEncDesc::encVen`[1]

Definition at line 836 of file `scsi_ses.c`.

6.267.2.4 `uint8_t SesEncDesc::encVid`[8]

Definition at line 833 of file `scsi_ses.c`.

6.267.2.5 `uint8_t SesEncDesc::encWWN`[8]

Definition at line 832 of file `scsi_ses.c`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_ses.c`

6.268 SesEncHdr Struct Reference

Data Fields

- `uint8_t` [Subencid](#)
- `uint8_t` [Ntypes](#)
- `uint8_t` [VEnclen](#)

6.268.1 Detailed Description

Definition at line 825 of file `scsi_ses.c`.

6.268.2 Field Documentation

6.268.2.1 `uint8_t` [SesEncHdr::Ntypes](#)

Definition at line 827 of file `scsi_ses.c`.

Referenced by `ses_enchdr()`, and `ses_getconfig()`.

6.268.2.2 `uint8_t` [SesEncHdr::Subencid](#)

Definition at line 826 of file `scsi_ses.c`.

Referenced by `ses_enchdr()`, and `ses_getconfig()`.

6.268.2.3 `uint8_t` [SesEncHdr::VEnclen](#)

Definition at line 828 of file `scsi_ses.c`.

Referenced by `ses_enchdr()`, and `ses_getconfig()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_ses.c`

6.269 SesThdr Struct Reference

Data Fields

- [uint8_t enc_type](#)
- [uint8_t enc_maxelt](#)
- [uint8_t enc_subenc](#)
- [uint8_t enc_tlen](#)

6.269.1 Detailed Description

Definition at line 839 of file `scsi_ses.c`.

6.269.2 Field Documentation

6.269.2.1 [uint8_t SesThdr::enc_maxelt](#)

Definition at line 841 of file `scsi_ses.c`.

Referenced by `ses_getconfig()`, and `ses_getthdr()`.

6.269.2.2 [uint8_t SesThdr::enc_subenc](#)

Definition at line 842 of file `scsi_ses.c`.

Referenced by `ses_getconfig()`, and `ses_getthdr()`.

6.269.2.3 [uint8_t SesThdr::enc_tlen](#)

Definition at line 843 of file `scsi_ses.c`.

Referenced by `ses_getconfig()`, and `ses_getthdr()`.

6.269.2.4 [uint8_t SesThdr::enc_type](#)

Definition at line 840 of file `scsi_ses.c`.

Referenced by `ses_getconfig()`, and `ses_getthdr()`.

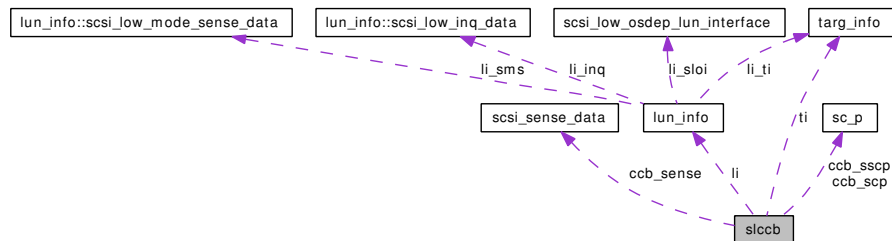
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_ses.c`

6.270 slccb Struct Reference

```
#include <scsi_low.h>
```

Collaboration diagram for slccb:



Public Member Functions

- [TAILQ_ENTRY](#) (`slccb`) `ccb_chain`

Data Fields

- `void * osdep`
- `targ_info * ti`
- `lun_info * li`
- `buf * bp`
- `scsi_low_tag_t ccb_tag`
- `scsi_low_tag_t ccb_otag`
- `sc_p ccb_scp`
- `sc_p ccb_sscp`
- `int ccb_dataalen`
- `u_int ccb_msgoutflag`
- `u_int ccb_omsgoutflag`
- `u_int ccb_flags`
- `u_int ccb_error`
- `int ccb_rcnt`
- `int ccb_selrcnt`
- `int ccb_tc`
- `int ccb_tcmx`
- `u_int8_t ccb_scsi_cmd [12]`
- `scsi_low_osdep_sense_data_t ccb_sense`

6.270.1 Detailed Description

Definition at line 232 of file `scsi_low.h`.

6.270.2 Member Function Documentation

6.270.2.1 `slccb::TAILQ_ENTRY (slccb)`

6.270.3 Field Documentation

6.270.3.1 `struct buf* slccb::bp`

Definition at line 239 of file `scsi_low.h`.

Referenced by `scsi_low_data()`, and `scsi_low_message_enqueue()`.

6.270.3.2 `int slccb::ccb_dataalen`

Definition at line 249 of file `scsi_low.h`.

Referenced by `scsi_low_done()`, `scsi_low_msginfunc_lcc()`, and `scsi_low_start()`.

6.270.3.3 `u_int slccb::ccb_error`

Definition at line 274 of file `scsi_low.h`.

Referenced by `scsi_low_abort_ccb()`, `scsi_low_disconnected()`, `scsi_low_done()`, `scsi_low_enqueue()`, `scsi_low_establish_ccb()`, `scsi_low_msginfunc_lcc()`, `scsi_low_revoke_ccb()`, `scsi_low_setup_done()`, `scsi_low_start()`, and `scsi_low_timeout_check()`.

6.270.3.4 `u_int slccb::ccb_flags`

Definition at line 260 of file `scsi_low.h`.

Referenced by `scsi_low_abort_ccb()`, `scsi_low_arbit_fail()`, `scsi_low_disconnected()`, `scsi_low_done()`, `scsi_low_enqueue()`, `scsi_low_establish_ccb()`, `scsi_low_msginfunc_lcc()`, `scsi_low_reset_nexus()`, `scsi_low_reset_nexus_lun()`, `scsi_low_revoke_ccb()`, `scsi_low_sense_abort_start()`, `scsi_low_start()`, `scsi_low_synch()`, `scsi_low_test_cmdlnk()`, `scsi_low_timeout_check()`, and `scsi_low_wide()`.

6.270.3.5 `u_int slccb::ccb_msgoutflag`

Definition at line 254 of file `scsi_low.h`.

Referenced by `scsi_low_ccb_message_exec()`, `scsi_low_disconnected()`, `scsi_low_establish_ccb()`, and `scsi_low_start()`.

6.270.3.6 `u_int slccb::ccb_omsgoutflag`

Definition at line 255 of file `scsi_low.h`.

Referenced by `scsi_low_abort_ccb()`, `scsi_low_disconnected()`, `scsi_low_done()`, and `scsi_low_establish_ccb()`.

6.270.3.7 `scsi_low_tag_t slccb::ccb_otag`

Definition at line 242 of file `scsi_low.h`.

Referenced by `scsi_low_enqueue()`, and `scsi_low_msginfunc_lcc()`.

6.270.3.8 `int slccb::ccb_rcnt`

Definition at line 276 of file `scsi_low.h`.

Referenced by `scsi_low_done()`, `scsi_low_revoke_ccb()`, and `scsi_low_setup_done()`.

6.270.3.9 `struct sc_p slccb::ccb_scp`

Definition at line 247 of file `scsi_low.h`.

Referenced by `scsi_low_done()`, `scsi_low_msginfunc_ext()`, `scsi_low_msginfunc_i_wide_residue()`, `scsi_low_msginfunc_lcc()`, `scsi_low_sense_abort_start()`, `scsi_low_setup_start()`, `scsi_low_start()`, and `scsi_low_unit_ready_cmd()`.

6.270.3.10 `u_int8_t slccb::ccb_scsi_cmd[12]`

Definition at line 284 of file `scsi_low.h`.

Referenced by `scsi_low_sense_abort_start()`, and `scsi_low_test_cmdlnk()`.

6.270.3.11 `int slccb::ccb_selrcnt`

Definition at line 277 of file `scsi_low.h`.

Referenced by `scsi_low_arbit_fail()`, and `scsi_low_timeout_check()`.

6.270.3.12 `scsi_low_osdep_sense_data_t slccb::ccb_sense`

Definition at line 285 of file `scsi_low.h`.

Referenced by `scsi_low_sense_abort_start()`, and `scsi_low_setup_done()`.

6.270.3.13 `struct sc_p slccb::ccb_sscp`

Definition at line 248 of file `scsi_low.h`.

Referenced by `scsi_low_data()`, `scsi_low_done()`, `scsi_low_establish_ccb()`, `scsi_low_msginfunc_lcc()`, `scsi_low_msginfunc_sdp()`, and `scsi_low_start()`.

6.270.3.14 `scsi_low_tag_t slccb::ccb_tag`

Definition at line 241 of file `scsi_low.h`.

Referenced by `scsi_low_abort_ccb()`, `scsi_low_enqueue()`, `scsi_low_establish_ccb()`, `scsi_low_msgfunc_identify()`, `scsi_low_msgfunc_qtag()`, and `scsi_low_msginfunc_lcc()`.

6.270.3.15 `int slccb::ccb_tc`

Definition at line 278 of file `scsi_low.h`.

Referenced by `scsi_low_enqueue()`, `scsi_low_msginfunc_lcc()`, `scsi_low_revoke_ccb()`, `scsi_low_start()`, and `scsi_low_timeout_check()`.

6.270.3.16 `int slccb::ccb_tmax`

Definition at line 279 of file `scsi_low.h`.

Referenced by `scsi_low_enqueue()`, `scsi_low_msginfunc_lcc()`, `scsi_low_revoke_ccb()`, `scsi_low_sense_abort_start()`, `scsi_low_setup_start()`, and `scsi_low_start()`.

6.270.3.17 `struct lun_info* slccb::li`

Definition at line 238 of file `scsi_low.h`.

Referenced by `scsi_low_abort_ccb()`, `scsi_low_disconnected()`, `scsi_low_enqueue()`, `scsi_low_establish_ccb()`, `scsi_low_find_ccb()`, `scsi_low_message_enqueue()`, `scsi_low_msgfunc_identify()`, `scsi_low_msginfunc_lcc()`, `scsi_low_reset_nexus_lun()`, `scsi_low_revoke_ccb()`, `scsi_low_setup_done()`, `scsi_low_start()`, `scsi_low_test_abort()`, and `scsi_low_timeout_check()`.

6.270.3.18 `void* slccb::osdep`

Definition at line 235 of file `scsi_low.h`.

Referenced by `scsi_low_done()`, `scsi_low_find_ccb()`, and `scsi_low_message_enqueue()`.

6.270.3.19 `struct targ_info* slccb::ti`

Definition at line 237 of file `scsi_low.h`.

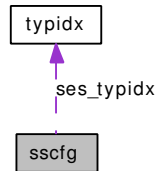
Referenced by `scsi_low_abort_ccb()`, `scsi_low_arbit_fail()`, `scsi_low_ccb_message_exec()`, `scsi_low_done()`, `scsi_low_enqueue()`, `scsi_low_revoke_ccb()`, `scsi_low_setup_done()`, and `scsi_low_start()`.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/scsi/scsi_low.h](#)

6.271 sscfg Struct Reference

Collaboration diagram for sscfg:



Data Fields

- `uint8_t ses_ntypes`
- `typidx * ses_typidx`
- `uint8_t * ses_eltmap`

6.271.1 Detailed Description

Definition at line 856 of file `scsi_ses.c`.

6.271.2 Field Documentation

6.271.2.1 `uint8_t* sscfg::ses_eltmap`

Definition at line 871 of file `scsi_ses.c`.

Referenced by `ses_getconfig()`, `ses_getputstat()`, and `ses_softc_init()`.

6.271.2.2 `uint8_t sscfg::ses_ntypes`

Definition at line 857 of file `scsi_ses.c`.

Referenced by `ses_getconfig()`, `ses_getputstat()`, and `ses_softc_init()`.

6.271.2.3 `struct typidx* sscfg::ses_typidx`

Definition at line 863 of file `scsi_ses.c`.

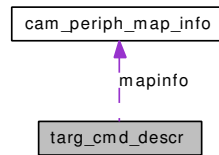
Referenced by `ses_getconfig()`, `ses_getputstat()`, and `ses_softc_init()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_ses.c`

6.272 targ_cmd_descr Struct Reference

Collaboration diagram for targ_cmd_descr:



Data Fields

- [cam_periph_map_info mapinfo](#)

6.272.1 Detailed Description

Definition at line 53 of file `scsi_target.c`.

6.272.2 Field Documentation

6.272.2.1 struct [cam_periph_map_info](#) targ_cmd_descr::mapinfo

Definition at line 54 of file `scsi_target.c`.

Referenced by `targreturnccb()`, and `targsendccb()`.

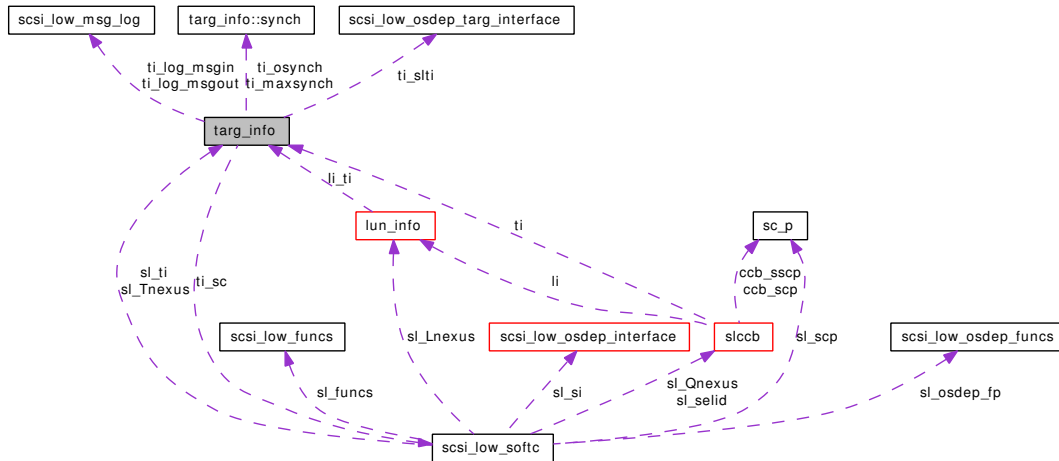
The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_target.c`

6.273 targ_info Struct Reference

```
#include <scsi_low.h>
```

Collaboration diagram for targ_info:



Public Member Functions

- [TAILQ_ENTRY \(targ_info\)](#) `ti_chain`

Data Fields

- `scsi_low_osdep_targ_interface` `ti_slti`
- `scsi_low_softc` * `ti_sc`
- `u_int` `ti_id`
- `lun_info_tab` `ti_litab`
- `int` `ti_disc`
- `u_int` `ti_phase`
- `u_int` `ti_ophase`
- `u_int` `ti_msginptr`
- `u_int` `ti_msginlen`
- `int` `ti_msgin_parity_error`
- `u_int8_t` `ti_msgin` [SCSI_LOW_MAX_MSGLEN]
- `u_int` `ti_msgflags`
- `u_int` `ti_omsgflags`
- `u_int` `ti_emsgflags`
- `u_int8_t` `ti_msgoutstr` [SCSI_LOW_MAX_MSGLEN]
- `u_int` `ti_msgoutlen`
- `u_int` `ti_setup_msg`
- `u_int` `ti_setup_msg_done`
- `u_int` `ti_flags_valid`
- `u_int` `ti_diskflags`
- `u_int` `ti_quirks`
- `targ_info::synch` `ti_osynch`

- [targ_info::synch ti_maxsynch](#)
- [u_int ti_owidth](#)
- [u_int ti_width](#)
- [int ti_lunsize](#)
- [scsi_low_msg_log ti_log_msgout](#)
- [scsi_low_msg_log ti_log_msgin](#)

Data Structures

- struct [synch](#)

6.273.1 Detailed Description

Definition at line 389 of file `scsi_low.h`.

6.273.2 Member Function Documentation

6.273.2.1 [targ_info::TAILQ_ENTRY \(targ_info\)](#)

6.273.3 Field Documentation

6.273.3.1 [int targ_info::ti_disc](#)

Definition at line 405 of file `scsi_low.h`.

Referenced by `scsi_low_bus_release()`, `scsi_low_disconnected()`, `scsi_low_reset_nexus_target()`, `scsi_low_revoke_ccb()`, and `scsi_low_timeout_check()`.

6.273.3.2 [u_int targ_info::ti_diskflags](#)

Definition at line 478 of file `scsi_low.h`.

Referenced by `scsi_low_alloc_ti()`, `scsi_low_reset_nexus_target()`, and `scsi_low_setup_done()`.

6.273.3.3 [u_int targ_info::ti_emsgflags](#)

Definition at line 438 of file `scsi_low.h`.

Referenced by `scsi_low_init_msgsys()`, `scsi_low_msginfunc_ext()`, `scsi_low_msginfunc_msg_reject()`, and `scsi_low_msgout()`.

6.273.3.4 [u_int targ_info::ti_flags_valid](#)

Definition at line 470 of file `scsi_low.h`.

Referenced by `scsi_low_alloc_ti()`, `scsi_low_reset_nexus_target()`, and `scsi_low_setup_done()`.

6.273.3.5 `u_int targ_info::ti_id`

Definition at line 395 of file `scsi_low.h`.

Referenced by `scsi_low_alloc_ti()`, `scsi_low_calcf_lun()`, `scsi_low_calcf_show()`, `scsi_low_cmd()`, `scsi_low_disconnected()`, `scsi_low_done()`, `scsi_low_msgin()`, `scsi_low_msginfunc_ext()`, `scsi_low_msginfunc_simple_qtag()`, `scsi_low_synch()`, and `scsi_low_wide()`.

6.273.3.6 `struct lun_info_tab targ_info::ti_litab`

Definition at line 400 of file `scsi_low.h`.

Referenced by `scsi_low_alloc_ti()`, `scsi_low_free_ti()`, `scsi_low_print()`, `scsi_low_reset_nexus_target()`, and `scsi_low_timeout_check()`.

6.273.3.7 `struct scsi_low_msg_log targ_info::ti_log_msgin`

Definition at line 498 of file `scsi_low.h`.

Referenced by `scsi_low_msgin()`, `scsi_low_msginfunc_lcc()`, and `scsi_low_start()`.

6.273.3.8 `struct scsi_low_msg_log targ_info::ti_log_msgout`

Definition at line 497 of file `scsi_low.h`.

Referenced by `scsi_low_msginfunc_lcc()`, `scsi_low_msgout()`, and `scsi_low_start()`.

6.273.3.9 `int targ_info::ti_lunsize`

Definition at line 494 of file `scsi_low.h`.

Referenced by `scsi_low_attach()`.

6.273.3.10 `struct targ_info::synch targ_info::ti_maxsynch`

Referenced by `scsi_low_calcf_lun()`, `scsi_low_calcf_show()`, `scsi_low_msgfunc_synch()`, and `scsi_low_synch()`.

6.273.3.11 `u_int targ_info::ti_msgflags`

Definition at line 436 of file `scsi_low.h`.

Referenced by `scsi_low_disconnected()`, `scsi_low_init_msgsys()`, and `scsi_low_msgout()`.

6.273.3.12 `u_int8_t targ_info::ti_msgin[SCSI_LOW_MAX_MSGLEN]`

Definition at line 431 of file `scsi_low.h`.

Referenced by `scsi_low_msgin()`, `scsi_low_msginfunc_ext()`, `scsi_low_msginfunc_i_wide_residue()`, `scsi_low_msginfunc_rejop()`, and `scsi_low_msginfunc_simple_qtag()`.

6.273.3.13 `int targ_info::ti_msgin_parity_error`

Definition at line 430 of file `scsi_low.h`.

Referenced by `scsi_low_msgin()`.

6.273.3.14 `u_int targ_info::ti_msginlen`

Definition at line 429 of file `scsi_low.h`.

Referenced by `scsi_low_msgin()`, and `scsi_low_msginfunc_ext()`.

6.273.3.15 `u_int targ_info::ti_msginptr`

Definition at line 428 of file `scsi_low.h`.

Referenced by `scsi_low_init_msgsys()`, `scsi_low_msgin()`, and `scsi_low_msginfunc_ext()`.

6.273.3.16 `u_int targ_info::ti_msgoutlen`

Definition at line 459 of file `scsi_low.h`.

Referenced by `scsi_low_msgfunc_identify()`, `scsi_low_msgfunc_qtag()`, `scsi_low_msgfunc_synch()`, `scsi_low_msgfunc_wide()`, and `scsi_low_msgout()`.

6.273.3.17 `u_int8_t targ_info::ti_msgoutstr[SCSI_LOW_MAX_MSGLEN]`

Definition at line 458 of file `scsi_low.h`.

Referenced by `scsi_low_msgfunc_identify()`, `scsi_low_msgfunc_qtag()`, `scsi_low_msgfunc_synch()`, `scsi_low_msgfunc_wide()`, and `scsi_low_msgout()`.

6.273.3.18 `u_int targ_info::ti_omsgflags`

Definition at line 437 of file `scsi_low.h`.

Referenced by `scsi_low_init_msgsys()`, and `scsi_low_msgout()`.

6.273.3.19 `u_int targ_info::ti_ophase`

Definition at line 423 of file `scsi_low.h`.

Referenced by `scsi_low_data()`, and `scsi_low_msgin()`.

6.273.3.20 `struct targ_info::synch targ_info::ti_osynch`

Referenced by `scsi_low_reset_nexus_target()`, and `scsi_low_synch()`.

6.273.3.21 `u_int targ_info::ti_owidth`

Definition at line 489 of file `scsi_low.h`.

Referenced by `scsi_low_alloc_ti()`, `scsi_low_reset_nexus_target()`, and `scsi_low_wide()`.

6.273.3.22 `u_int targ_info::ti_phase`

Definition at line 422 of file `scsi_low.h`.

Referenced by `scsi_low_data()`, `scsi_low_disconnected()`, `scsi_low_msgfunc_identify()`, `scsi_low_msgfunc_qtag()`, `scsi_low_msgin()`, and `scsi_low_reselected()`.

6.273.3.23 `u_int targ_info::ti_quirks`

Definition at line 479 of file `scsi_low.h`.

Referenced by `scsi_low_alloc_ti()`.

6.273.3.24 `struct scsi_low_softc* targ_info::ti_sc`

Definition at line 394 of file `scsi_low.h`.

Referenced by `scsi_low_alloc_li()`, `scsi_low_alloc_ti()`, `scsi_low_calcf_lun()`, `scsi_low_calcf_show()`, and `scsi_low_calcf_target()`.

6.273.3.25 `u_int targ_info::ti_setup_msg`

Definition at line 464 of file `scsi_low.h`.

Referenced by `scsi_low_message_enqueue()`, and `scsi_low_reset_nexus_target()`.

6.273.3.26 `u_int targ_info::ti_setup_msg_done`

Definition at line 465 of file `scsi_low.h`.

Referenced by `scsi_low_reset_nexus_target()`, `scsi_low_sense_abort_start()`, `scsi_low_synch()`, and `scsi_low_wide()`.

6.273.3.27 `struct scsi_low_osdep_targ_interface targ_info::ti_slfi`

Definition at line 390 of file `scsi_low.h`.

6.273.3.28 `u_int targ_info::ti_width`

Definition at line 489 of file `scsi_low.h`.

Referenced by `scsi_low_calcf_show()`, `scsi_low_msgfunc_wide()`, `scsi_low_msginfunc_i_wide_residue()`, and `scsi_low_wide()`.

The documentation for this struct was generated from the following file:

- /usr/src/sys/cam/scsi/scsi_low.h

6.274 targ_info::synch Struct Reference

```
#include <scsi_low.h>
```

Data Fields

- [u_int8_t offset](#)
- [u_int8_t period](#)

6.274.1 Detailed Description

Definition at line 481 of file `scsi_low.h`.

6.274.2 Field Documentation

6.274.2.1 [u_int8_t targ_info::synch::offset](#)

Definition at line 482 of file `scsi_low.h`.

Referenced by `scsi_low_calcf_lun()`, `scsi_low_calcf_show()`, `scsi_low_msgfunc_synch()`, `scsi_low_reset_nexus_target()`, and `scsi_low_synch()`.

6.274.2.2 [u_int8_t targ_info::synch::period](#)

Definition at line 483 of file `scsi_low.h`.

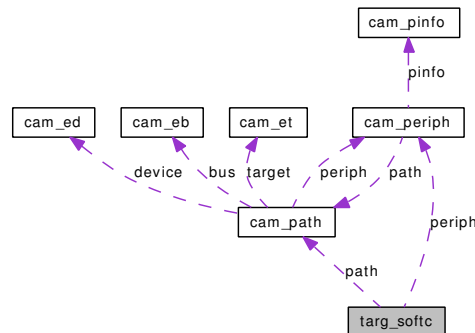
Referenced by `scsi_low_calcf_show()`, `scsi_low_msgfunc_synch()`, `scsi_low_reset_nexus_target()`, and `scsi_low_synch()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_low.h`

6.275 targ_softc Struct Reference

Collaboration diagram for targ_softc:



Data Fields

- `ccb_queue` [pending_ccb_queue](#)
- `descr_queue` [work_queue](#)
- `descr_queue` [abort_queue](#)
- `ccb_queue` [user_ccb_queue](#)
- `cam_periph` * [periph](#)
- `cam_path` * [path](#)
- `targ_state` [state](#)
- `selinfo` [read_select](#)
- `devstat` [device_stats](#)

6.275.1 Detailed Description

Definition at line 73 of file `scsi_target.c`.

6.275.2 Field Documentation

6.275.2.1 struct `descr_queue` [targ_softc::abort_queue](#)

Definition at line 81 of file `scsi_target.c`.

Referenced by `abort_all_pending()`, `targdtor()`, `targpoll()`, `targread()`, `targreadfilt()`, and `targstart()`.

6.275.2.2 struct `devstat` [targ_softc::device_stats](#)

Definition at line 93 of file `scsi_target.c`.

6.275.2.3 struct `cam_path*` [targ_softc::path](#)

Definition at line 90 of file `scsi_target.c`.

Referenced by `abort_all_pending()`, `targctor()`, `targdisable()`, `targdtor()`, `targenable()`, `targgetccb()`, `targioctl()`, `targread()`, `targreturnccb()`, `targsendccb()`, `targstart()`, `targusermerge()`, and `targwrite()`.

6.275.2.4 struct ccb_queue targ_softc::pending_ccb_queue

Definition at line 75 of file scsi_target.c.

Referenced by abort_all_pending(), targdone(), targsendccb(), targusermerge(), and targwrite().

6.275.2.5 struct cam_periph* targ_softc::periph

Definition at line 89 of file scsi_target.c.

Referenced by targclose(), targctor(), targdone(), targdtor(), targetenable(), targstart(), and targwrite().

6.275.2.6 struct selinfo targ_softc::read_select

Definition at line 92 of file scsi_target.c.

Referenced by notify_user(), targkqfilter(), targpoll(), and targreadfiltdetach().

6.275.2.7 targ_state targ_softc::state

Definition at line 91 of file scsi_target.c.

Referenced by targdisable(), targdone(), targetenable(), and targioctl().

6.275.2.8 struct ccb_queue targ_softc::user_ccb_queue

Definition at line 87 of file scsi_target.c.

Referenced by abort_all_pending(), notify_user(), targdone(), targdtor(), targpoll(), targread(), and targreadfilt().

6.275.2.9 struct descr_queue targ_softc::work_queue

Definition at line 78 of file scsi_target.c.

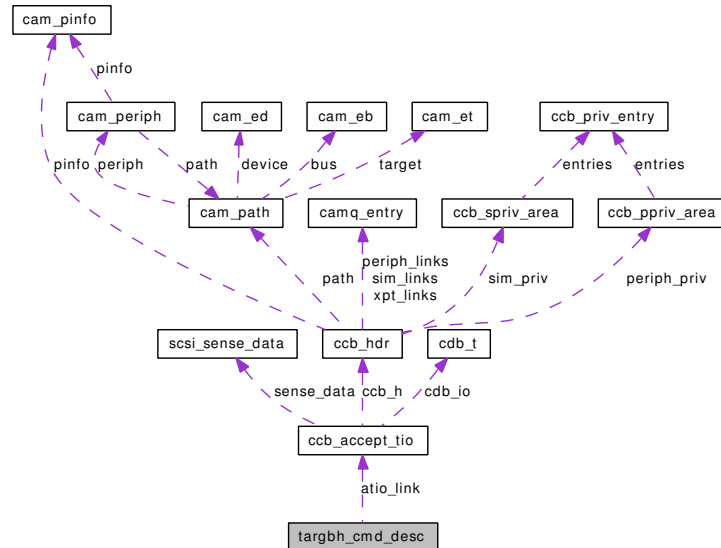
Referenced by abort_all_pending(), targstart(), and targwrite().

The documentation for this struct was generated from the following file:

- /usr/src/sys/cam/scsi/scsi_target.c

6.276 targbh_cmd_desc Struct Reference

Collaboration diagram for targbh_cmd_desc:



Data Fields

- `ccb_accept_tio * atio_link`
- `u_int data_resid`
- `u_int data_increment`
- `void * data`
- `void * backing_store`
- `u_int max_size`
- `u_int32_t timeout`
- `u_int8_t status`

6.276.1 Detailed Description

Definition at line 97 of file `scsi_targ_bh.c`.

6.276.2 Field Documentation

6.276.2.1 `struct ccb_accept_tio * targbh_cmd_desc::atio_link`

Definition at line 98 of file `scsi_targ_bh.c`.

6.276.2.2 `void * targbh_cmd_desc::backing_store`

Definition at line 102 of file `scsi_targ_bh.c`.

Referenced by `targbhfreedescr()`.

6.276.2.3 void* [targbh_cmd_desc::data](#)

Definition at line 101 of file `scsi_targ_bh.c`.

Referenced by `targbhdone()`, and `targbhstart()`.

6.276.2.4 u_int [targbh_cmd_desc::data_increment](#)

Definition at line 100 of file `scsi_targ_bh.c`.

Referenced by `targbhdone()`, and `targbhstart()`.

6.276.2.5 u_int [targbh_cmd_desc::data_resid](#)

Definition at line 99 of file `scsi_targ_bh.c`.

Referenced by `targbhdone()`, and `targbhstart()`.

6.276.2.6 u_int [targbh_cmd_desc::max_size](#)

Definition at line 103 of file `scsi_targ_bh.c`.

6.276.2.7 u_int8_t [targbh_cmd_desc::status](#)

Definition at line 105 of file `scsi_targ_bh.c`.

Referenced by `targbhdone()`, and `targbhstart()`.

6.276.2.8 u_int32_t [targbh_cmd_desc::timeout](#)

Definition at line 104 of file `scsi_targ_bh.c`.

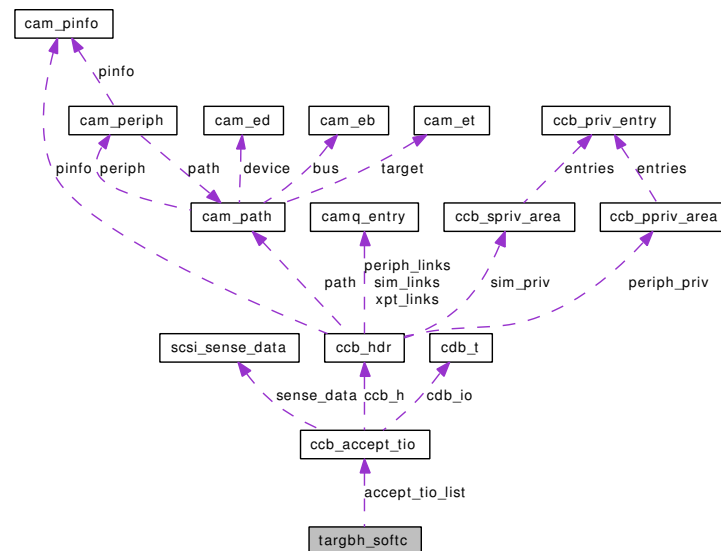
Referenced by `targbhdone()`, and `targbhstart()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_targ_bh.c`

6.277 targbh_softc Struct Reference

Collaboration diagram for targbh_softc:



Data Fields

- `ccb_queue` [pending_queue](#)
- `ccb_queue` [work_queue](#)
- `ccb_queue` [unknown_atio_queue](#)
- `devstat` [device_stats](#)
- `targbh_state` [state](#)
- `targbh_flags` [flags](#)
- `u_int` [init_level](#)
- `u_int` [inq_data_len](#)
- `ccb_accept_tio` * [accept_tio_list](#)
- `ccb_hdr_slist` [immed_notify_slist](#)

6.277.1 Detailed Description

Definition at line 84 of file `scsi_targ_bh.c`.

6.277.2 Field Documentation

6.277.2.1 struct `ccb_accept_tio`* `targbh_softc::accept_tio_list`

Definition at line 93 of file `scsi_targ_bh.c`.

Referenced by `targbhdislun()`, and `targbhenlun()`.

6.277.2.2 struct devstat [targbh_softc::device_stats](#)

Definition at line 88 of file `scsi_targ_bh.c`.

6.277.2.3 [targbh_flags](#) [targbh_softc::flags](#)

Definition at line 90 of file `scsi_targ_bh.c`.

Referenced by `targbhdislun()`, and `targbhenlun()`.

6.277.2.4 struct ccb_hdr_slist [targbh_softc::immed_notify_slist](#)

Definition at line 94 of file `scsi_targ_bh.c`.

Referenced by `targbhdislun()`, and `targbhenlun()`.

6.277.2.5 u_int [targbh_softc::init_level](#)

Definition at line 91 of file `scsi_targ_bh.c`.

Referenced by `targbhdtor()`.

6.277.2.6 u_int [targbh_softc::inq_data_len](#)

Definition at line 92 of file `scsi_targ_bh.c`.

6.277.2.7 struct ccb_queue [targbh_softc::pending_queue](#)

Definition at line 85 of file `scsi_targ_bh.c`.

Referenced by `targbhdone()`, and `targbhstart()`.

6.277.2.8 [targbh_state](#) [targbh_softc::state](#)

Definition at line 89 of file `scsi_targ_bh.c`.

Referenced by `targbhdone()`, and `targbhdtor()`.

6.277.2.9 struct ccb_queue [targbh_softc::unknown_atio_queue](#)

Definition at line 87 of file `scsi_targ_bh.c`.

6.277.2.10 struct ccb_queue [targbh_softc::work_queue](#)

Definition at line 86 of file `scsi_targ_bh.c`.

Referenced by `targbhdone()`, and `targbhstart()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_targ_bh.c`

6.278 typidx Struct Reference

Data Fields

- int [ses_tidx](#)
- int [ses_oidx](#)

6.278.1 Detailed Description

Definition at line 851 of file `scsi_ses.c`.

6.278.2 Field Documentation

6.278.2.1 int [typidx::ses_oidx](#)

Definition at line 853 of file `scsi_ses.c`.

Referenced by `ses_getconfig()`, and `ses_getputstat()`.

6.278.2.2 int [typidx::ses_tidx](#)

Definition at line 852 of file `scsi_ses.c`.

Referenced by `ses_getconfig()`, and `ses_getputstat()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_ses.c`

6.279 volume_tag Struct Reference

```
#include <scsi_ch.h>
```

Data Fields

- `u_int8_t vif` [32]
- `u_int8_t reserved` [2]
- `u_int8_t vsn` [2]

6.279.1 Detailed Description

Definition at line 200 of file `scsi_ch.h`.

6.279.2 Field Documentation

6.279.2.1 `u_int8_t volume_tag::reserved`[2]

Definition at line 202 of file `scsi_ch.h`.

6.279.2.2 `u_int8_t volume_tag::vif`[32]

Definition at line 201 of file `scsi_ch.h`.

Referenced by `copy_votag()`.

6.279.2.3 `u_int8_t volume_tag::vsn`[2]

Definition at line 203 of file `scsi_ch.h`.

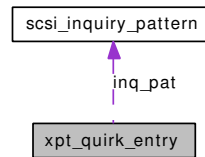
Referenced by `copy_votag()`.

The documentation for this struct was generated from the following file:

- `/usr/src/sys/cam/scsi/scsi_ch.h`

6.280 xpt_quirk_entry Struct Reference

Collaboration diagram for xpt_quirk_entry:



Data Fields

- [scsi_inquiry_pattern inq_pat](#)
- [u_int8_t quirks](#)
- [u_int mintags](#)
- [u_int maxtags](#)

6.280.1 Detailed Description

Definition at line 204 of file cam_xpt.c.

6.280.2 Field Documentation

6.280.2.1 struct [scsi_inquiry_pattern](#) [xpt_quirk_entry::inq_pat](#)

Definition at line 205 of file cam_xpt.c.

6.280.2.2 [u_int](#) [xpt_quirk_entry::maxtags](#)

Definition at line 212 of file cam_xpt.c.

6.280.2.3 [u_int](#) [xpt_quirk_entry::mintags](#)

Definition at line 211 of file cam_xpt.c.

6.280.2.4 [u_int8_t](#) [xpt_quirk_entry::quirks](#)

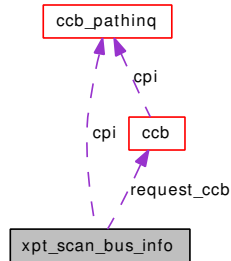
Definition at line 206 of file cam_xpt.c.

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/cam_xpt.c](#)

6.281 xpt_scan_bus_info Struct Reference

Collaboration diagram for xpt_scan_bus_info:



Data Fields

- [ccb * request_ccb](#)
- [ccb_pathinq * cpi](#)
- [int counter](#)

6.281.1 Detailed Description

Definition at line 5266 of file cam_xpt.c.

6.281.2 Field Documentation

6.281.2.1 [int xpt_scan_bus_info::counter](#)

Definition at line 5269 of file cam_xpt.c.

Referenced by [xpt_scan_bus\(\)](#).

6.281.2.2 [struct ccb_pathinq* xpt_scan_bus_info::cpi](#)

Definition at line 5268 of file cam_xpt.c.

Referenced by [xpt_scan_bus\(\)](#).

6.281.2.3 [union ccb* xpt_scan_bus_info::request_ccb](#)

Definition at line 5267 of file cam_xpt.c.

Referenced by [xpt_scan_bus\(\)](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/cam_xpt.c](#)

6.282 xpt_softc Struct Reference

Data Fields

- [xpt_flags flags](#)
- [u_int32_t generation](#)

6.282.1 Detailed Description

Definition at line 244 of file cam_xpt.c.

6.282.2 Field Documentation

6.282.2.1 [xpt_flags xpt_softc::flags](#)

Definition at line 245 of file cam_xpt.c.

Referenced by [xptclose\(\)](#), and [xptopen\(\)](#).

6.282.2.2 [u_int32_t xpt_softc::generation](#)

Definition at line 246 of file cam_xpt.c.

Referenced by [xpt_add_periph\(\)](#), [xpt_remove_periph\(\)](#), and [xptioctl\(\)](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/cam_xpt.c](#)

6.283 xpt_traverse_config Struct Reference

Data Fields

- [xpt_traverse_depth depth](#)
- [void * tr_func](#)
- [void * tr_arg](#)

6.283.1 Detailed Description

Definition at line 630 of file cam_xpt.c.

6.283.2 Field Documentation

6.283.2.1 [xpt_traverse_depth xpt_traverse_config::depth](#)

Definition at line 631 of file cam_xpt.c.

Referenced by [xpt_for_all_busses\(\)](#), [xpt_for_all_devices\(\)](#), [xptdefbusfunc\(\)](#), [xptdefdevicefunc\(\)](#), and [xptdeftargetfunc\(\)](#).

6.283.2.2 [void* xpt_traverse_config::tr_arg](#)

Definition at line 633 of file cam_xpt.c.

Referenced by [xpt_for_all_busses\(\)](#), [xpt_for_all_devices\(\)](#), [xptdefbusfunc\(\)](#), [xptdefdevicefunc\(\)](#), [xptdefperiphfunc\(\)](#), and [xptdeftargetfunc\(\)](#).

6.283.2.3 [void* xpt_traverse_config::tr_func](#)

Definition at line 632 of file cam_xpt.c.

Referenced by [xpt_for_all_busses\(\)](#), [xpt_for_all_devices\(\)](#), [xptdefbusfunc\(\)](#), [xptdefdevicefunc\(\)](#), [xptdefperiphfunc\(\)](#), and [xptdeftargetfunc\(\)](#).

The documentation for this struct was generated from the following file:

- [/usr/src/sys/cam/cam_xpt.c](#)

Chapter 7

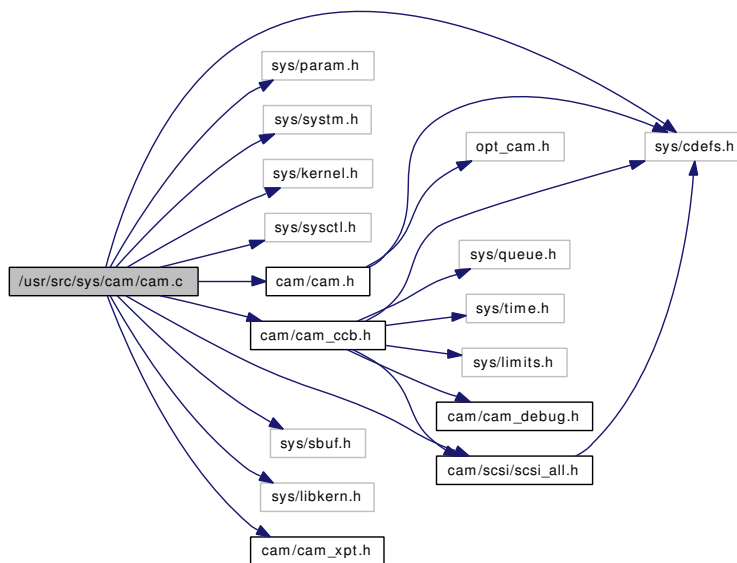
FreeBSD kernel CAM code File Documentation

7.1 notreviewed.dox File Reference

7.2 /usr/src/sys/cam/cam.c File Reference

```
#include <sys/cdefs.h>
#include <sys/param.h>
#include <sys/system.h>
#include <sys/kernel.h>
#include <sys/sysctl.h>
#include <cam/cam.h>
#include <cam/cam_ccb.h>
#include <cam/scsi/scsi_all.h>
#include <sys/sbuf.h>
#include <sys/libkern.h>
#include <cam/cam_xpt.h>
```

Include dependency graph for cam.c:



Functions

- `__FBSDID` ("FreeBSD: src/sys/cam/cam.c,v 1.10 2006/04/18 21:53:39 mjacob Exp \$")
- static int `camstatusentrycomp` (const void *key, const void *member)
- `SYSCTL_NODE` (_kern, OID_AUTO, cam, CTLFLAG_RD, 0,"CAM Subsystem")
- void `cam_strvis` (u_int8_t *dst, const u_int8_t *src, int srclen, int dstlen)
- int `cam_strmatch` (const u_int8_t *str, const u_int8_t *pattern, int str_len)
- caddr_t `cam_quirkmatch` (caddr_t target, caddr_t quirk_table, int num_entries, int entry_size, cam_quirkmatch_t *comp_func)
- `cam_status_entry` * `cam_fetch_status_entry` (cam_status status)

- char * `cam_error_string` (union `ccb` *`ccb`, char *`str`, int `str_len`, `cam_error_string_flags` `flags`, `cam_error_proto_flags` `proto_flags`)
- void `cam_error_print` (union `ccb` *`ccb`, `cam_error_string_flags` `flags`, `cam_error_proto_flags` `proto_flags`)
- void `cam_calc_geometry` (struct `ccb_calc_geometry` *`cgc`, int `extended`)

Variables

- `cam_status_entry` `cam_status_table` []
- const int `num_cam_status_entries`

7.2.1 Function Documentation

7.2.1.1 `__FBSDID ("FreeBSD: src/sys/cam/cam.c, v 1.10 2006/04/18 21:53:39 mjacob Exp $")`

7.2.1.2 void `cam_calc_geometry` (struct `ccb_calc_geometry` *`cgc`, int `extended`)

Definition at line 372 of file `cam.c`.

References `ccb_calc_geometry::block_size`, `CAM_REQ_CMP`, `CAM_REQ_CMP_ERR`, `ccb_calc_geometry::ccb_h`, `ccb_calc_geometry::cylinders`, `ccb_calc_geometry::heads`, `L`, `ccb_calc_geometry::secs_per_track`, `ccb_hdr::status`, and `ccb_calc_geometry::volume_size`.

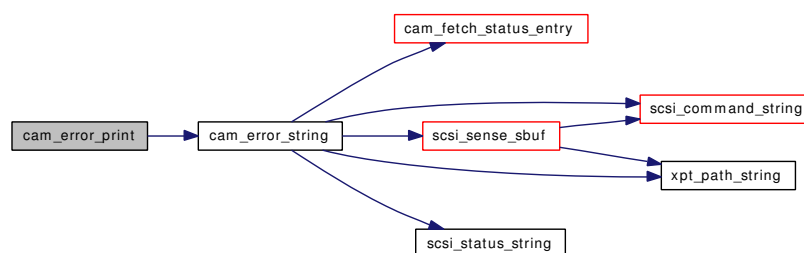
7.2.1.3 void `cam_error_print` (union `ccb` *`ccb`, `cam_error_string_flags` `flags`, `cam_error_proto_flags` `proto_flags`)

Definition at line 337 of file `cam.c`.

References `cam_error_string()`.

Referenced by `camperiphdone()`, and `camperiphscsisenseerror()`.

Here is the call graph for this function:



7.2.1.4 char* `cam_error_string` (union `ccb` *`ccb`, char *`str`, int `str_len`, `cam_error_string_flags` `flags`, `cam_error_proto_flags` `proto_flags`)

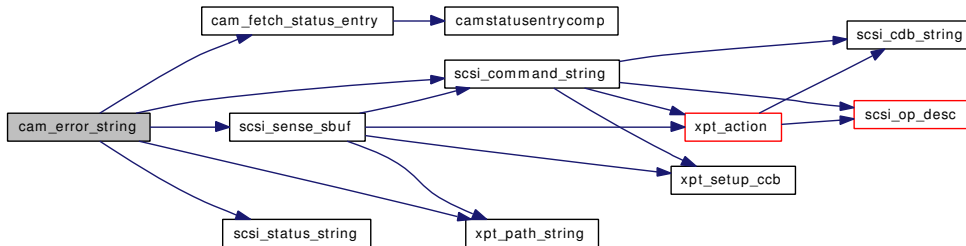
Definition at line 208 of file `cam.c`.

References `CAM_AUTOSNS_VALID`, `CAM_EPF_ALL`, `CAM_EPF_LEVEL_MASK`, `CAM_EPF_MINIMAL`, `CAM_EPF_NONE`, `CAM_EPF_NORMAL`, `CAM_ESF_CAM_STATUS`, `CAM_ESF_COMMAND`, `CAM_ESF_NONE`, `CAM_ESF_PRINT_SENSE`, `CAM_ESF_PRINT_STATUS`, `CAM_`

ESF_PROTO_STATUS, cam_fetch_status_entry(), CAM SCSI_STATUS_ERROR, CAM_STATUS_MASK, ccb::ccb_h, ccb_scsiio::ccb_h, ccb::csio, ccb_hdr::func_code, ccb_hdr::path, scsi_command_string(), scsi_sense_sbuf(), ccb_scsiio::scsi_status, SCSI_STATUS_CHECK_COND, scsi_status_string(), SSS_FLAG_NONE, ccb_hdr::status, cam_status_entry::status_text, xpt_path_string(), and XPT SCSI IO.

Referenced by cam_error_print().

Here is the call graph for this function:



7.2.1.5 struct cam_status_entry* cam_fetch_status_entry (cam_status status)

Definition at line 184 of file cam.c.

References CAM_STATUS_MASK, camstatusentrycomp(), and num_cam_status_entries.

Referenced by cam_error_string(), and passasync().

Here is the call graph for this function:



7.2.1.6 caddr_t cam_quirkmatch (caddr_t target, caddr_t quirk_table, int num_entries, int entry_size, cam_quirkmatch_t * comp_func)

Definition at line 173 of file cam.c.

Referenced by cdregister(), daregister(), fetchtableentries(), saregister(), scsi_op_desc(), xpt_find_quirk(), and xptdevicematch().

7.2.1.7 int cam_strmatch (const u_int8_t * str, const u_int8_t * pattern, int str_len)

Definition at line 150 of file cam.c.

Referenced by scsi_inquiry_match(), and scsi_static_inquiry_match().

7.2.1.8 void cam_strvis (u_int8_t * dst, const u_int8_t * src, int srclen, int dstlen)

Definition at line 107 of file cam.c.

Referenced by scsi_print_inquiry().

7.2.1.9 static int camstatusentrycomp (const void * *key*, const void * *member*) [static]

Definition at line 194 of file cam.c.

References `cam_status_entry::status_code`.

Referenced by `cam_fetch_status_entry()`.

7.2.1.10 SYSCTL_NODE (_kern, OID_AUTO, cam, CTLFLAG_RD, 0, "CAM Subsystem")**7.2.2 Variable Documentation****7.2.2.1 struct [cam_status_entry](#) [cam_status_table](#) []**

Definition at line 55 of file cam.c.

7.2.2.2 const int [num_cam_status_entries](#)

Initial value:

```
sizeof(cam_status_table)/sizeof(*cam_status_table)
```

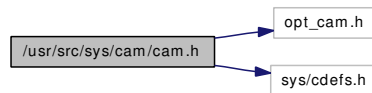
Definition at line 99 of file cam.c.

Referenced by `cam_fetch_status_entry()`.

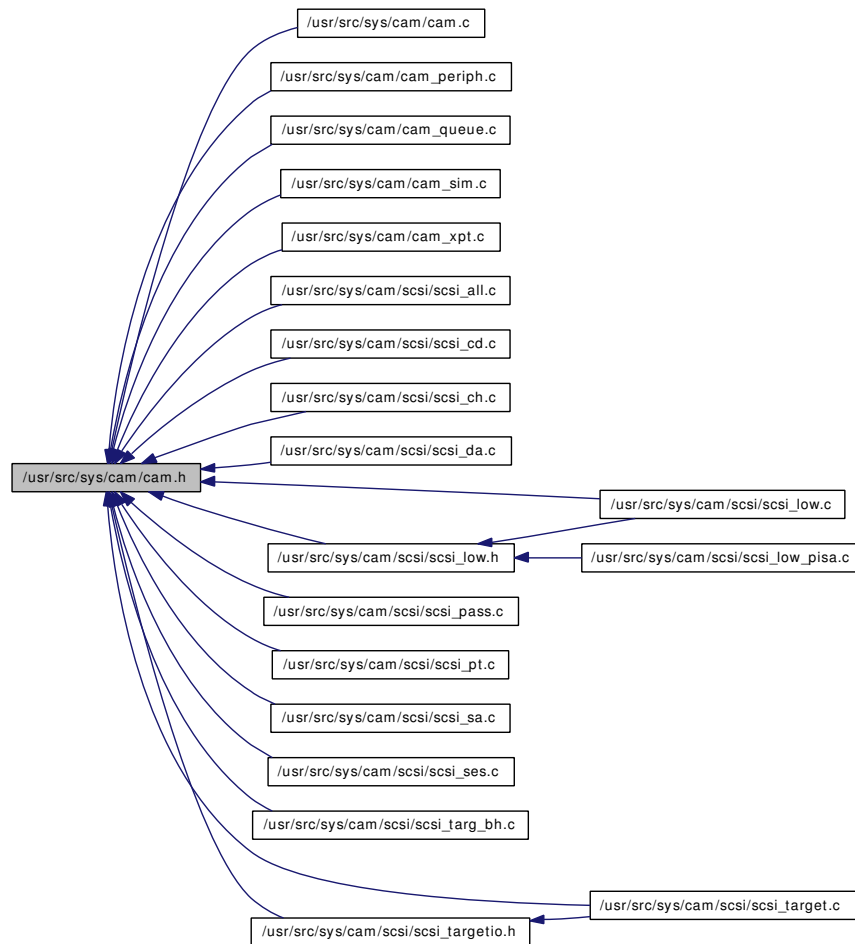
7.3 /usr/src/sys/cam/cam.h File Reference

```
#include <opt_cam.h>
#include <sys/cdefs.h>
```

Include dependency graph for cam.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- struct [cam_pinfo](#)
- struct [cam_status_entry](#)

Defines

- #define `_CAM_CAM_H` 1
- #define `CAM_XPT_PATH_ID` ((path_id_t)~0)
- #define `CAM_BUS_WILDCARD` ((path_id_t)~0)
- #define `CAM_TARGET_WILDCARD` ((target_id_t)~0)
- #define `CAM_LUN_WILDCARD` ((lun_id_t)~0)
- #define `CAM_MAX_CDBLEN` 16
- #define `CAM_PRIORITY_NONE` (u_int32_t)-1
- #define `CAM_UNQUEUED_INDEX` -1
- #define `CAM_ACTIVE_INDEX` -2
- #define `CAM_DONEQ_INDEX` -3
- #define `GENERATIONCMP(x, op, y)` ((int32_t)((x) - (y)) op 0)

Typedefs

- typedef u_int `path_id_t`
- typedef u_int `target_id_t`
- typedef u_int `lun_id_t`

Enumerations

- enum `cam_flags` { `CAM_FLAG_NONE` = 0x00, `CAM_EXPECT_INQ_CHANGE` = 0x01, `CAM_RETRY_SELTO` = 0x02 }
- enum `cam_status` {
 - `CAM_REQ_INPROG`, `CAM_REQ_CMP`, `CAM_REQ_ABORTED`, `CAM_UA_ABORT`,
 - `CAM_REQ_CMP_ERR`, `CAM_BUSY`, `CAM_REQ_INVALID`, `CAM_PATH_INVALID`,
 - `CAM_DEV_NOT_THERE`, `CAM_UA_TERMIO`, `CAM_SEL_TIMEOUT`, `CAM_CMD_TIMEOUT`,
 - `CAM_SCSI_STATUS_ERROR`, `CAM_MSG_REJECT_REC`, `CAM_SCSI_BUS_RESET`, `CAM_UNCOR_PARITY`,
 - `CAM_AUTOSENSE_FAIL` = 0x10, `CAM_NO_HBA`, `CAM_DATA_RUN_ERR`, `CAM_UNEXP_BUSFREE`,
 - `CAM_SEQUENCE_FAIL`, `CAM_CCB_LEN_ERR`, `CAM_PROVIDE_FAIL`, `CAM_BDR_SENT`,
 - `CAM_REQ_TERMIO`, `CAM_UNREC_HBA_ERROR`, `CAM_REQ_TOO_BIG`, `CAM_REQUEUE_REQ`,
 - `CAM_IDE` = 0x33, `CAM_RESRC_UNAVAIL`, `CAM_UNACKED_EVENT`, `CAM_MESSAGE_RECV`,
 - `CAM_INVALID_CDB`, `CAM_LUN_INVALID`, `CAM_TID_INVALID`, `CAM_FUNC_NOTAVAIL`,
 - `CAM_NO_NEXUS`, `CAM_IID_INVALID`, `CAM_CDB_RECVD`, `CAM_LUN_ALRDY_ENA`,
 - `CAM_SCSI_BUSY`, `CAM_DEV_QFRZN` = 0x40, `CAM_AUTOSNS_VALID` = 0x80, `CAM_RELEASE_SIMQ` = 0x100,
 - `CAM_SIM_QUEUED` = 0x200, `CAM_STATUS_MASK` = 0x3F, `CAM_SENT_SENSE` = 0x40000000 }

- enum `cam_error_string_flags` {
`CAM_ESF_NONE` = 0x00, `CAM_ESF_COMMAND` = 0x01, `CAM_ESF_CAM_STATUS` = 0x02,
`CAM_ESF_PROTO_STATUS` = 0x04,
`CAM_ESF_ALL` = 0xff }
- enum `cam_error_proto_flags` {
`CAM_EPF_NONE` = 0x00, `CAM_EPF_MINIMAL` = 0x01, `CAM_EPF_NORMAL` = 0x02, `CAM_-`
`EPF_ALL` = 0x03,
`CAM_EPF_LEVEL_MASK` = 0x0f }
- enum `cam_error_scsi_flags` { `CAM_ESF_PRINT_NONE` = 0x00, `CAM_ESF_PRINT_STATUS` =
0x10, `CAM_ESF_PRINT_SENSE` = 0x20 }

Functions

- `__BEGIN_DECLS` typedef int() `cam_quirkmatch_t` (`caddr_t`, `caddr_t`)
- `caddr_t` `cam_quirkmatch` (`caddr_t` target, `caddr_t` quirk_table, int num_entries, int entry_size, `cam_-`
`quirkmatch_t` *comp_func)
- void `cam_strvis` (`u_int8_t` *dst, const `u_int8_t` *src, int srclen, int dstlen)
- int `cam_strmatch` (const `u_int8_t` *str, const `u_int8_t` *pattern, int str_len)
- `cam_status_entry` * `cam_fetch_status_entry` (`cam_status` status)
- char * `cam_error_string` (`union ccb` *ccb, char *str, int str_len, `cam_error_string_flags` flags, `cam_-`
`error_proto_flags` proto_flags)
- void `cam_error_print` (`union ccb` *ccb, `cam_error_string_flags` flags, `cam_error_proto_flags` proto_-
flags)
- static `__END_DECLS` `__inline` void `cam_init_pinfo` (`cam_pinfo` *pinfo)

Variables

- `cam_status_entry` `cam_status_table` []
- const int `num_cam_status_entries`

7.3.1 Define Documentation

7.3.1.1 #define _CAM_CAM_H 1

Definition at line 32 of file cam.h.

7.3.1.2 #define CAM_ACTIVE_INDEX -2

Definition at line 73 of file cam.h.

Referenced by `cam_ccbq_send_ccb()`.

7.3.1.3 #define CAM_BUS_WILDCARD ((`path_id_t`)~0)

Definition at line 45 of file cam.h.

Referenced by `xpt_path_comp()`.

7.3.1.4 #define CAM_DONEQ_INDEX -3

Definition at line 74 of file cam.h.

Referenced by xpt_action(), and xpt_done().

7.3.1.5 #define CAM_LUN_WILDCARD ((lun_id_t)~0)

Definition at line 47 of file cam.h.

Referenced by cam_periph_error(), chinit(), dainit(), passinit(), ptinit(), sainit(), sesinit(), STAILQ_HEAD(), targbhasync(), targbhdone(), targbhinit(), xpt_alloc_device(), xpt_async(), xpt_bus_deregister(), xpt_bus_register(), xpt_dev_async(), xpt_init(), xpt_path_comp(), xpt_path_lun_id(), xptconfigbuscountfunc(), xptconfigfunc(), xptioctl(), and xptsetasynbusfunc().

7.3.1.6 #define CAM_MAX_CDBLEN 16

Definition at line 52 of file cam.h.

7.3.1.7 #define CAM_PRIORITY_NONE (u_int32_t)-1

Definition at line 69 of file cam.h.

Referenced by cam_init_pinfo(), cam_periph_alloc(), cdstart(), chstart(), dastart(), passstart(), ptstart(), sastart(), sesstart(), and targbhstart().

7.3.1.8 #define CAM_TARGET_WILDCARD ((target_id_t)~0)

Definition at line 46 of file cam.h.

Referenced by chinit(), dainit(), passinit(), ptinit(), sainit(), sesinit(), STAILQ_HEAD(), targbhasync(), targbhdone(), targbhinit(), xpt_async(), xpt_bus_deregister(), xpt_bus_register(), xpt_dev_async(), xpt_init(), xpt_path_comp(), xpt_path_target_id(), xpt_setup_ccb(), xptconfigbuscountfunc(), xptconfigfunc(), xptioctl(), and xptsetasynbusfunc().

7.3.1.9 #define CAM_UNQUEUED_INDEX -1

Definition at line 72 of file cam.h.

Referenced by cam_init_pinfo(), cam_periph_ccbwait(), camisr(), camq_remove(), cdgetccb(), cdoninvalidate(), cdregister(), cdschedule(), device_is_alloc_queued(), device_is_send_queued(), periph_is_queued(), xpt_action(), xpt_release_device(), xpt_schedule_dev(), and xpt_setup_ccb().

7.3.1.10 #define CAM_XPT_PATH_ID ((path_id_t)~0)

Definition at line 44 of file cam.h.

Referenced by chinit(), dainit(), passinit(), ptinit(), sainit(), sesinit(), STAILQ_HEAD(), targbhinit(), xpt_bus_register(), xpt_init(), xptconfigbuscountfunc(), xptconfigfunc(), and xptpathid().

7.3.1.11 #define GENERATIONCMP(x, op, y) ((int32_t)((x) - (y)) op 0)

Definition at line 86 of file cam.h.

7.3.2 Typedef Documentation

7.3.2.1 typedef u_int lun_id_t

Definition at line 42 of file cam.h.

7.3.2.2 typedef u_int path_id_t

Definition at line 40 of file cam.h.

7.3.2.3 typedef u_int target_id_t

Definition at line 41 of file cam.h.

7.3.3 Enumeration Type Documentation

7.3.3.1 enum cam_error_proto_flags

Enumerator:

CAM_EPF_NONE
CAM_EPF_MINIMAL
CAM_EPF_NORMAL
CAM_EPF_ALL
CAM_EPF_LEVEL_MASK

Definition at line 167 of file cam.h.

7.3.3.2 enum cam_error_scsi_flags

Enumerator:

CAM_ESF_PRINT_NONE
CAM_ESF_PRINT_STATUS
CAM_ESF_PRINT_SENSE

Definition at line 176 of file cam.h.

7.3.3.3 enum cam_error_string_flags

Enumerator:

CAM_ESF_NONE
CAM_ESF_COMMAND

CAM_ESF_CAM_STATUS
CAM_ESF_PROTO_STATUS
CAM_ESF_ALL

Definition at line 159 of file cam.h.

7.3.3.4 enum `cam_flags`

Enumerator:

CAM_FLAG_NONE
CAM_EXPECT_INQ_CHANGE
CAM_RETRY_SELTO

Definition at line 89 of file cam.h.

7.3.3.5 enum `cam_status`

Enumerator:

CAM_REQ_INPROG
CAM_REQ_CMP
CAM_REQ_ABORTED
CAM_UA_ABORT
CAM_REQ_CMP_ERR
CAM_BUSY
CAM_REQ_INVALID
CAM_PATH_INVALID
CAM_DEV_NOT_THERE
CAM_UA_TERMIO
CAM_SEL_TIMEOUT
CAM_CMD_TIMEOUT
CAM_SCSI_STATUS_ERROR
CAM_MSG_REJECT_REC
CAM_SCSI_BUS_RESET
CAM_UNCOR_PARITY
CAM_AUTOSENSE_FAIL
CAM_NO_HBA
CAM_DATA_RUN_ERR
CAM_UNEXP_BUSFREE
CAM_SEQUENCE_FAIL
CAM_CCB_LEN_ERR
CAM_PROVIDE_FAIL
CAM_BDR_SENT

CAM_REQ_TERMIO
CAM_UNREC_HBA_ERROR
CAM_REQ_TOO_BIG
CAM_REQUEUE_REQ
CAM_IDE
CAM_RESRC_UNAVAIL
CAM_UNACKED_EVENT
CAM_MESSAGE_RECV
CAM_INVALID_CDB
CAM_LUN_INVALID
CAM_TID_INVALID
CAM_FUNC_NOTAVAIL
CAM_NO_NEXUS
CAM_IID_INVALID
CAM_CDB_RECVD
CAM_LUN_ALRDY_ENA
CAM_SCSI_BUSY
CAM_DEV_QFRZN
CAM_AUTOSNS_VALID
CAM_RELEASE_SIMQ
CAM_SIM_QUEUED
CAM_STATUS_MASK
CAM_SENT_SENSE

Definition at line 96 of file cam.h.

7.3.4 Function Documentation

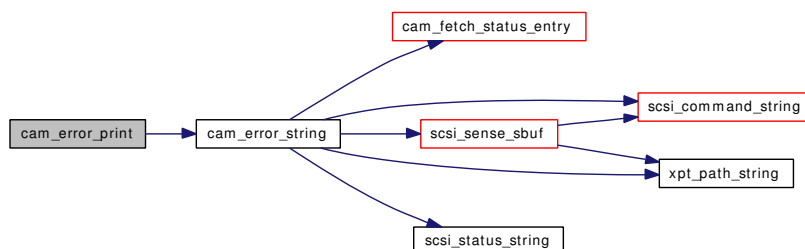
7.3.4.1 **void** `cam_error_print` (**union** `ccb` * `ccb`, **cam_error_string_flags** `flags`, **cam_error_proto_flags** `proto_flags`)

Definition at line 337 of file cam.c.

References `cam_error_string()`.

Referenced by `camperiphdone()`, and `camperiphscsisenseerror()`.

Here is the call graph for this function:



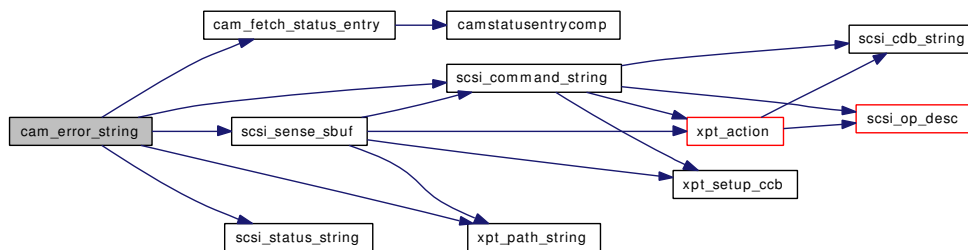
7.3.4.2 `char* cam_error_string (union ccb * ccb, char * str, int str_len, cam_error_string_flags flags, cam_error_proto_flags proto_flags)`

Definition at line 208 of file cam.c.

References CAM_AUTOSNS_VALID, CAM_EPF_ALL, CAM_EPF_LEVEL_MASK, CAM_EPF_MINIMAL, CAM_EPF_NONE, CAM_EPF_NORMAL, CAM_ESF_CAM_STATUS, CAM_ESF_COMMAND, CAM_ESF_NONE, CAM_ESF_PRINT_SENSE, CAM_ESF_PRINT_STATUS, CAM_ESF_PROTO_STATUS, cam_fetch_status_entry(), CAM SCSI_STATUS_ERROR, CAM_STATUS_MASK, ccb_scsiio::ccb_h, ccb::ccb_h, ccb::csio, ccb_hdr::func_code, ccb_hdr::path, scsi_command_string(), scsi_sense_sbuf(), ccb_scsiio::scsi_status, SCSI_STATUS_CHECK_COND, scsi_status_string(), SSS_FLAG_NONE, ccb_hdr::status, cam_status_entry::status_text, xpt_path_string(), and XPT SCSI IO.

Referenced by cam_error_print().

Here is the call graph for this function:



7.3.4.3 `struct cam_status_entry* cam_fetch_status_entry (cam_status status)`

Definition at line 184 of file cam.c.

References CAM_STATUS_MASK, camstatusentrycomp(), and num_cam_status_entries.

Referenced by cam_error_string(), and passasync().

Here is the call graph for this function:



7.3.4.4 `static __inline void cam_init_pinfo (cam_pinfo * pinfo) [static]`

Definition at line 228 of file cam.h.

References CAM_PRIORITY_NONE, CAM_UNQUEUED_INDEX, cam_pinfo::index, and cam_pinfo::priority.

Referenced by cam_periph_alloc(), and xpt_alloc_device().

7.3.4.5 `caddr_t cam_quirkmatch (caddr_t target, caddr_t quirk_table, int num_entries, int entry_size, cam_quirkmatch_t * comp_func)`

Definition at line 173 of file cam.c.

Referenced by `cdregister()`, `daregister()`, `fetchtableentries()`, `saregister()`, `scsi_op_desc()`, `xpt_find_quirk()`, and `xptdevicematch()`.

7.3.4.6 `__BEGIN_DECLS typedef int() cam_quirkmatch_t (caddr_t, caddr_t)`

7.3.4.7 `int cam_strmatch (const u_int8_t * str, const u_int8_t * pattern, int str_len)`

Definition at line 150 of file `cam.c`.

Referenced by `scsi_inquiry_match()`, and `scsi_static_inquiry_match()`.

7.3.4.8 `void cam_strvis (u_int8_t * dst, const u_int8_t * src, int srclen, int dstlen)`

Definition at line 107 of file `cam.c`.

Referenced by `scsi_print_inquiry()`.

7.3.5 Variable Documentation

7.3.5.1 `struct cam_status_entry cam_status_table[]`

Definition at line 55 of file `cam.c`.

7.3.5.2 `const int num_cam_status_entries`

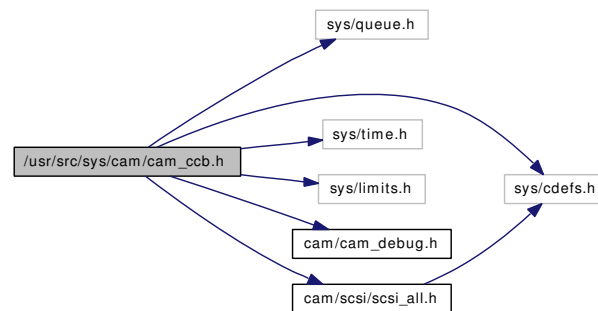
Definition at line 99 of file `cam.c`.

Referenced by `cam_fetch_status_entry()`.

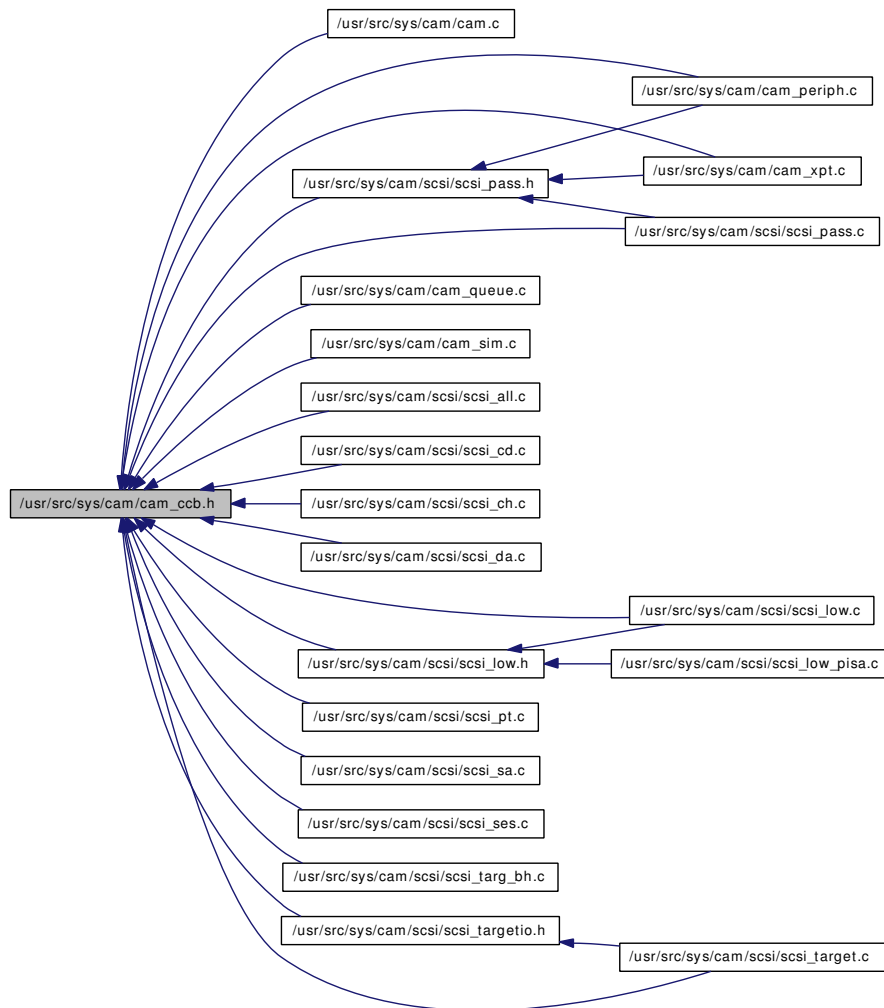
7.4 /usr/src/sys/cam/cam_ccb.h File Reference

```
#include <sys/queue.h>
#include <sys/cdefs.h>
#include <sys/time.h>
#include <sys/limits.h>
#include <cam/cam_debug.h>
#include <cam/scsi/scsi_all.h>
```

Include dependency graph for cam_ccb.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- union [camq_entry](#)
- union [ccb_priv_entry](#)
- union [ccb_ppriv_area](#)
- union [ccb_spriv_area](#)
- struct [ccb_hdr](#)
- struct [ccb_getdev](#)
- struct [ccb_getdevstats](#)
- struct [ccb_getdevlist](#)
- struct [periph_match_pattern](#)
- struct [device_match_pattern](#)
- struct [bus_match_pattern](#)
- union [match_pattern](#)
- struct [dev_match_pattern](#)
- struct [periph_match_result](#)
- struct [device_match_result](#)

- struct [bus_match_result](#)
- union [match_result](#)
- struct [dev_match_result](#)
- struct [ccb_dm_cookie](#)
- struct [ccb_dev_position](#)
- struct [ccb_dev_match](#)
- struct [ccb_pathinq_settings_spi](#)
- struct [ccb_pathinq_settings_fc](#)
- struct [ccb_pathinq_settings_sas](#)
- struct [ccb_pathinq](#)
- struct [ccb_pathstats](#)
- union [sense_t](#)
- union [cdb_t](#)
- struct [ccb_scsiio](#)
- struct [ccb_accept_tio](#)
- struct [ccb_relsim](#)
- struct [ccb_setasync](#)
- struct [ccb_setdev](#)
- struct [ccb_abort](#)
- struct [ccb_resetbus](#)
- struct [ccb_resetdev](#)
- struct [ccb_termio](#)
- struct [ccb_trans_settings_scsi](#)
- struct [ccb_trans_settings_spi](#)
- struct [ccb_trans_settings_fc](#)
- struct [ccb_trans_settings_sas](#)
- struct [ccb_trans_settings](#)
- struct [ccb_calc_geometry](#)
- struct [ccb_rescan](#)
- struct [ccb_debug](#)
- struct [ccb_en_lun](#)
- struct [ccb_immed_notify](#)
- struct [ccb_notify_ack](#)
- struct [ccb_eng_inq](#)
- struct [ccb_eng_exec](#)
- union [ccb](#)

Defines

- #define [_CAM_CAM_CCB_H](#) 1
- #define [IOCDBLEN](#) CAM_MAX_CDBLEN
- #define [VUHBALEN](#) 14
- #define [SIM_IDLEN](#) 16
- #define [HBA_IDLEN](#) 16
- #define [DEV_IDLEN](#) 16
- #define [CCB_PERIPH_PRIV_SIZE](#) 2
- #define [CCB_SIM_PRIV_SIZE](#) 2
- #define [XPT_FC_GROUP_MASK](#) 0xF0
- #define [XPT_FC_GROUP](#)(op) ((op) & XPT_FC_GROUP_MASK)
- #define [XPT_FC_GROUP_COMMON](#) 0x00

- #define XPT_FC_GROUP_SCSI_CONTROL 0x10
- #define XPT_FC_GROUP_HBA_ENGINE 0x20
- #define XPT_FC_GROUP_TMODE 0x30
- #define XPT_FC_GROUP_VENDOR_UNIQUE 0x80
- #define XPT_FC_IS_DEV_QUEUED(ccb) (((ccb) → ccb_h.func_code & XPT_FC_DEV_-QUEUED) == XPT_FC_DEV_QUEUED)
- #define XPT_FC_IS_QUEUED(ccb) (((ccb) → ccb_h.func_code & XPT_FC_QUEUED) != 0)
- #define PROTO_VERSION_UNKNOWN (UINT_MAX - 1)
- #define PROTO_VERSION_UNSPECIFIED UINT_MAX
- #define XPORT_VERSION_UNKNOWN (UINT_MAX - 1)
- #define XPORT_VERSION_UNSPECIFIED UINT_MAX
- #define CAM_BUS_GENERATION 0x00
- #define CAM_TARGET_GENERATION 0x01
- #define CAM_DEV_GENERATION 0x02
- #define CAM_PERIPH_GENERATION 0x03
- #define CAM_VERSION 0x15
- #define PATHINQ_SETTINGS_SIZE 128
- #define CAM_TAG_ACTION_NONE 0x00
- #define RELSIM_ADJUST_OPENINGS 0x01
- #define RELSIM_RELEASE_AFTER_TIMEOUT 0x02
- #define RELSIM_RELEASE_AFTER_CMDCMPLT 0x04
- #define RELSIM_RELEASE_AFTER_QEMPTY 0x08
- #define CTS_SCSI_VALID_TQ 0x01
- #define CTS_SCSI_FLAGS_TAG_ENB 0x01
- #define CTS_SPI_VALID_SYNC_RATE 0x01
- #define CTS_SPI_VALID_SYNC_OFFSET 0x02
- #define CTS_SPI_VALID_BUS_WIDTH 0x04
- #define CTS_SPI_VALID_DISC 0x08
- #define CTS_SPI_VALID_PPR_OPTIONS 0x10
- #define CTS_SPI_FLAGS_DISC_ENB 0x01
- #define CTS_FC_VALID_WWNN 0x8000
- #define CTS_FC_VALID_WWPN 0x4000
- #define CTS_FC_VALID_PORT 0x2000
- #define CTS_FC_VALID_SPEED 0x1000
- #define CTS_SAS_VALID_SPEED 0x1000
- #define CAM_TIME_DEFAULT 0x00000000
- #define CAM_TIME_INFINITY 0xFFFFFFFF
- #define CAM_SUCCESS 0
- #define CAM_FAILURE 1
- #define CAM_FALSE 0
- #define CAM_TRUE 1
- #define XPT_CCB_INVALID -1

Typedefs

- typedef void [ac_callback_t](#) (void *softc, u_int32_t code, struct [cam_path](#) *path, void *args)

Enumerations

- enum `ccb_flags` {
 - `CAM_CDB_POINTER` = 0x00000001, `CAM_QUEUE_ENABLE` = 0x00000002, `CAM_CDB_LINKED` = 0x00000004, `CAM_NEGOTIATE` = 0x00000008,
 - `CAM_SCATTER_VALID` = 0x00000010, `CAM_DIS_AUTOSENSE` = 0x00000020, `CAM_DIR_RESV` = 0x00000000, `CAM_DIR_IN` = 0x00000040,
 - `CAM_DIR_OUT` = 0x00000080, `CAM_DIR_NONE` = 0x000000C0, `CAM_DIR_MASK` = 0x000000C0, `CAM_SOFT_RST_OP` = 0x00000100,
 - `CAM_ENG_SYNC` = 0x00000200, `CAM_DEV_QFRZDIS` = 0x00000400, `CAM_DEV_QFREEZE` = 0x00000800, `CAM_HIGH_POWER` = 0x00001000,
 - `CAM_SENSE_PTR` = 0x00002000, `CAM_SENSE_PHYS` = 0x00004000, `CAM_TAG_ACTION_VALID` = 0x00008000, `CAM_PASS_ERR_RECOVER` = 0x00010000,
 - `CAM_DIS_DISCONNECT` = 0x00020000, `CAM_SG_LIST_PHYS` = 0x00040000, `CAM_MSG_BUF_PHYS` = 0x00080000, `CAM_SNS_BUF_PHYS` = 0x00100000,
 - `CAM_DATA_PHYS` = 0x00200000, `CAM_CDB_PHYS` = 0x00400000, `CAM_ENG_SGLIST` = 0x00800000, `CAM_DIS_AUTOSRP` = 0x01000000,
 - `CAM_DIS_AUTODISC` = 0x02000000, `CAM_TGT_CCB_AVAIL` = 0x04000000, `CAM_TGT_PHASE_MODE` = 0x08000000, `CAM_MSGB_VALID` = 0x10000000,
 - `CAM_STATUS_VALID` = 0x20000000, `CAM_DATAB_VALID` = 0x40000000, `CAM_SEND_SENSE` = 0x08000000, `CAM_TERM_IO` = 0x10000000,
 - `CAM_DISCONNECT` = 0x20000000, `CAM_SEND_STATUS` = 0x40000000 }
- enum `xpt_opcode` {
 - `XPT_FC_QUEUED` = 0x100, `XPT_FC_USER_CCB` = 0x200, `XPT_FC_XPT_ONLY` = 0x400, `XPT_FC_DEV_QUEUED` = 0x800 | `XPT_FC_QUEUED`,
 - `XPT_NOOP` = 0x00, `XPT SCSI_IO` = 0x01 | `XPT_FC_DEV_QUEUED`, `XPT_GDEV_TYPE` = 0x02, `XPT_GDEVLIST` = 0x03,
 - `XPT_PATH_INQ` = 0x04, `XPT_REL_SIMQ` = 0x05, `XPT_SASYNC_CB` = 0x06, `XPT_SDEV_TYPE` = 0x07,
 - `XPT_SCAN_BUS`, `XPT_DEV_MATCH` = 0x09 | `XPT_FC_XPT_ONLY`, `XPT_DEBUG` = 0x0a, `XPT_PATH_STATS` = 0x0b,
 - `XPT_GDEV_STATS` = 0x0c, `XPT_ABORT` = 0x10, `XPT_RESET_BUS` = 0x11 | `XPT_FC_XPT_ONLY`, `XPT_RESET_DEV` = 0x12 | `XPT_FC_DEV_QUEUED`,
 - `XPT_TERM_IO` = 0x13, `XPT_SCAN_LUN`, `XPT_GET_TRAN_SETTINGS` = 0x15, `XPT_SET_TRAN_SETTINGS` = 0x16,
 - `XPT_CALC_GEOMETRY` = 0x17, `XPT_ENG_INQ` = 0x20 | `XPT_FC_XPT_ONLY`, `XPT_ENG_EXEC` = 0x21 | `XPT_FC_DEV_QUEUED`, `XPT_EN_LUN` = 0x30,
 - `XPT_TARGET_IO` = 0x31 | `XPT_FC_DEV_QUEUED`, `XPT_ACCEPT_TARGET_IO` = 0x32 | `XPT_FC_QUEUED` | `XPT_FC_USER_CCB`, `XPT_CONT_TARGET_IO` = 0x33 | `XPT_FC_DEV_QUEUED`, `XPT_IMMED_NOTIFY` = 0x34 | `XPT_FC_QUEUED` | `XPT_FC_USER_CCB`,
 - `XPT_NOTIFY_ACK` = 0x35, `XPT_VUNIQUE` = 0x80 }
- enum `cam_proto` {
 - `PROTO_UNKNOWN`, `PROTO_UNSPECIFIED`, `PROTO_SCSI`, `PROTO_ATA`,
 - `PROTO_ATAPI` }

- enum `cam_xport` {
`XPORT_UNKNOWN`, `XPORT_UNSPECIFIED`, `XPORT_SPI`, `XPORT_FC`,
`XPORT_SSA`, `XPORT_USB`, `XPORT_PPB`, `XPORT_ATA`,
`XPORT_SAS` }
- enum `ccb_getdevlist_status_e` { `CAM_GDEVLIST_LAST_DEVICE`, `CAM_GDEVLIST_LIST_CHANGED`, `CAM_GDEVLIST_MORE_DEVS`, `CAM_GDEVLIST_ERROR` }
- enum `periph_pattern_flags` {
`PERIPH_MATCH_NONE` = 0x000, `PERIPH_MATCH_PATH` = 0x001, `PERIPH_MATCH_TARGET` = 0x002, `PERIPH_MATCH_LUN` = 0x004,
`PERIPH_MATCH_NAME` = 0x008, `PERIPH_MATCH_UNIT` = 0x010, `PERIPH_MATCH_ANY` = 0x01f }
- enum `dev_pattern_flags` {
`DEV_MATCH_NONE` = 0x000, `DEV_MATCH_PATH` = 0x001, `DEV_MATCH_TARGET` = 0x002, `DEV_MATCH_LUN` = 0x004,
`DEV_MATCH_INQUIRY` = 0x008, `DEV_MATCH_ANY` = 0x00f }
- enum `bus_pattern_flags` {
`BUS_MATCH_NONE` = 0x000, `BUS_MATCH_PATH` = 0x001, `BUS_MATCH_NAME` = 0x002, `BUS_MATCH_UNIT` = 0x004,
`BUS_MATCH_BUS_ID` = 0x008, `BUS_MATCH_ANY` = 0x00f }
- enum `dev_match_type` { `DEV_MATCH_PERIPH`, `DEV_MATCH_DEVICE`, `DEV_MATCH_BUS` }
- enum `dev_result_flags` { `DEV_RESULT_NOFLAG` = 0x00, `DEV_RESULT_UNCONFIGURED` = 0x01 }
- enum `ccb_dev_match_status` {
`CAM_DEV_MATCH_LAST`, `CAM_DEV_MATCH_MORE`, `CAM_DEV_MATCH_LIST_CHANGED`, `CAM_DEV_MATCH_SIZE_ERROR`,
`CAM_DEV_MATCH_ERROR` }
- enum `dev_pos_type` {
`CAM_DEV_POS_NONE` = 0x000, `CAM_DEV_POS_BUS` = 0x001, `CAM_DEV_POS_TARGET` = 0x002, `CAM_DEV_POS_DEVICE` = 0x004,
`CAM_DEV_POS_PERIPH` = 0x008, `CAM_DEV_POS_PDPTR` = 0x010, `CAM_DEV_POS_TYPMASK` = 0xf00, `CAM_DEV_POS_EDT` = 0x100,
`CAM_DEV_POS_PDRV` = 0x200 }
- enum `pi_inqflag` {
`PI_MDP_ABLE` = 0x80, `PI_WIDE_32` = 0x40, `PI_WIDE_16` = 0x20, `PI_SDTR_ABLE` = 0x10,
`PI_LINKED_CDB` = 0x08, `PI_TAG_ABLE` = 0x02, `PI_SOFT_RST` = 0x01 }
- enum `pi_tmflag` {
`PIT_PROCESSOR` = 0x80, `PIT_PHASE` = 0x40, `PIT_DISCONNECT` = 0x20, `PIT_TERM_IO` = 0x10,
`PIT_GRP_6` = 0x08, `PIT_GRP_7` = 0x04 }
- enum `pi_miscflag` {
`PIM_SCANHILO` = 0x80, `PIM_NOREMOVE` = 0x40, `PIM_NOINITIATOR` = 0x20, `PIM_NOBUSRESET` = 0x10,
`PIM_NO_6_BYTE` = 0x08, `PIM_SEQSCAN` = 0x04 }

- enum `ac_code` {
`AC_GETDEV_CHANGED` = 0x800, `AC_INQ_CHANGED` = 0x400, `AC_TRANSFER_NEG` = 0x200, `AC_LOST_DEVICE` = 0x100,
`AC_FOUND_DEVICE` = 0x080, `AC_PATH_DEREGISTERED` = 0x040, `AC_PATH_REGISTERED` = 0x020, `AC_SENT_BDR` = 0x010,
`AC_SCSI_AEN` = 0x008, `AC_UNSOL_RESEL` = 0x002, `AC_BUS_RESET` = 0x001 }
- enum `cts_type` { `CTS_TYPE_CURRENT_SETTINGS`, `CTS_TYPE_USER_SETTINGS` }
- enum `ei_type` { `EIT_BUFFER`, `EIT_LOSSLESS`, `EIT_LOSSY`, `EIT_ENCRYPT` }
- enum `ei_algo` { `EAD_VUNIQUE`, `EAD_LZ1V1`, `EAD_LZ2V1`, `EAD_LZ2V2` }

Functions

- static `__BEGIN_DECLS` `__inline` void `cam_fill_csio` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int32_t flags, u_int8_t tag_action, u_int8_t *data_ptr, u_int32_t dxfer_len, u_int8_t sense_len, u_int8_t cdb_len, u_int32_t timeout)
- static `__inline` void `cam_fill_ctio` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int32_t flags, u_int tag_action, u_int tag_id, u_int init_id, u_int scsi_status, u_int8_t *data_ptr, u_int32_t dxfer_len, u_int32_t timeout)
- void `cam_calc_geometry` (struct `ccb_calc_geometry` *ccg, int extended)

7.4.1 Define Documentation

7.4.1.1 `#define CAM_CAM_CCB_H 1`

Definition at line 32 of file `cam_ccb.h`.

7.4.1.2 `#define CAM_BUS_GENERATION 0x00`

Definition at line 464 of file `cam_ccb.h`.

Referenced by `xptedtbusfunc()`, `xptedtdevicefunc()`, `xptedtmatch()`, and `xptedtperiphfunc()`.

7.4.1.3 `#define CAM_DEV_GENERATION 0x02`

Definition at line 466 of file `cam_ccb.h`.

Referenced by `xptedtdevicefunc()`, `xptedtperiphfunc()`, and `xptedttargetfunc()`.

7.4.1.4 `#define CAM_FAILURE 1`

Definition at line 867 of file `cam_ccb.h`.

7.4.1.5 `#define CAM_FALSE 0`

Definition at line 869 of file `cam_ccb.h`.

7.4.1.6 #define CAM_PERIPH_GENERATION 0x03

Definition at line 467 of file cam_ccb.h.

Referenced by xptedtdevicefunc(), xptedtperiphfunc(), xptplistpdrvfunc(), and xptplistperiphfunc().

7.4.1.7 #define CAM_SUCCESS 0

Definition at line 866 of file cam_ccb.h.

Referenced by xpt_bus_register(), and xpt_init().

7.4.1.8 #define CAM_TAG_ACTION_NONE 0x00

Definition at line 612 of file cam_ccb.h.

Referenced by camperiphscsisenseerror(), and xpt_run_dev_sendq().

7.4.1.9 #define CAM_TARGET_GENERATION 0x01

Definition at line 465 of file cam_ccb.h.

Referenced by xptedtbusfunc(), xptedtdevicefunc(), and xptedtperiphfunc().

7.4.1.10 #define CAM_TIME_DEFAULT 0x00000000

Definition at line 863 of file cam_ccb.h.

7.4.1.11 #define CAM_TIME_INFINITY 0xFFFFFFFF

Definition at line 864 of file cam_ccb.h.

7.4.1.12 #define CAM_TRUE 1

Definition at line 870 of file cam_ccb.h.

7.4.1.13 #define CAM_VERSION 0x15

Definition at line 487 of file cam_ccb.h.

7.4.1.14 #define CCB_PERIPH_PRIV_SIZE 2

Definition at line 51 of file cam_ccb.h.

7.4.1.15 #define CCB_SIM_PRIV_SIZE 2

Definition at line 52 of file cam_ccb.h.

7.4.1.16 #define CTS_FC_VALID_PORT 0x2000

Definition at line 732 of file cam_ccb.h.

Referenced by xpt_announce_periph().

7.4.1.17 #define CTS_FC_VALID_SPEED 0x1000

Definition at line 733 of file cam_ccb.h.

Referenced by xpt_announce_periph().

7.4.1.18 #define CTS_FC_VALID_WWNN 0x8000

Definition at line 730 of file cam_ccb.h.

Referenced by xpt_announce_periph().

7.4.1.19 #define CTS_FC_VALID_WWPN 0x4000

Definition at line 731 of file cam_ccb.h.

Referenced by xpt_announce_periph().

7.4.1.20 #define CTS_SAS_VALID_SPEED 0x1000

Definition at line 742 of file cam_ccb.h.

Referenced by xpt_announce_periph().

7.4.1.21 #define CTS_SCSI_FLAGS_TAG_ENB 0x01

Definition at line 709 of file cam_ccb.h.

Referenced by xpt_set_transfer_settings(), and xpt_toggle_tags().

7.4.1.22 #define CTS_SCSI_VALID_TQ 0x01

Definition at line 707 of file cam_ccb.h.

Referenced by xpt_set_transfer_settings(), and xpt_toggle_tags().

7.4.1.23 #define CTS_SPI_FLAGS_DISC_ENB 0x01

Definition at line 721 of file cam_ccb.h.

Referenced by xpt_set_transfer_settings().

7.4.1.24 #define CTS_SPI_VALID_BUS_WIDTH 0x04

Definition at line 717 of file cam_ccb.h.

Referenced by xpt_announce_periph(), and xpt_set_transfer_settings().

7.4.1.25 #define CTS_SPI_VALID_DISC 0x08

Definition at line 718 of file cam_ccb.h.

Referenced by xpt_set_transfer_settings().

7.4.1.26 #define CTS_SPI_VALID_PPR_OPTIONS 0x10

Definition at line 719 of file cam_ccb.h.

Referenced by xpt_set_transfer_settings().

7.4.1.27 #define CTS_SPI_VALID_SYNC_OFFSET 0x02

Definition at line 716 of file cam_ccb.h.

Referenced by proberequestbackoff(), xpt_announce_periph(), and xpt_set_transfer_settings().

7.4.1.28 #define CTS_SPI_VALID_SYNC_RATE 0x01

Definition at line 715 of file cam_ccb.h.

Referenced by proberequestbackoff(), and xpt_set_transfer_settings().

7.4.1.29 #define DEV_IDLEN 16

Definition at line 50 of file cam_ccb.h.

Referenced by xpt_action(), xptaction(), xptbusmatch(), xptedtbusfunc(), xptedtperiphfunc(), xptperiphmatch(), and xptplistperiphfunc().

7.4.1.30 #define HBA_IDLEN 16

Definition at line 49 of file cam_ccb.h.

Referenced by xptaction().

7.4.1.31 #define IOCDBLEN CAM_MAX_CDBLEN

Definition at line 46 of file cam_ccb.h.

Referenced by ses_runcmd().

7.4.1.32 #define PATHINQ_SETTINGS_SIZE 128

Definition at line 530 of file cam_ccb.h.

7.4.1.33 #define PROTO_VERSION_UNKNOWN (UINT_MAX - 1)

Definition at line 230 of file cam_ccb.h.

Referenced by xpt_set_transfer_settings().

7.4.1.34 #define PROTO_VERSION_UNSPECIFIED UINT_MAX

Definition at line 231 of file cam_ccb.h.

Referenced by xpt_set_transfer_settings(), xpt_toggle_tags(), and xptaction().

7.4.1.35 #define RELSIM_ADJUST_OPENINGS 0x01

Definition at line 632 of file cam_ccb.h.

Referenced by camperiphscsistatuserror(), and xpt_action().

7.4.1.36 #define RELSIM_RELEASE_AFTER_CMDCMPLT 0x04

Definition at line 634 of file cam_ccb.h.

Referenced by camperiphscsistatuserror(), and xpt_action().

7.4.1.37 #define RELSIM_RELEASE_AFTER_QEMPTY 0x08

Definition at line 635 of file cam_ccb.h.

Referenced by xpt_action(), xpt_set_transfer_settings(), and xpt_start_tags().

7.4.1.38 #define RELSIM_RELEASE_AFTER_TIMEOUT 0x02

Definition at line 633 of file cam_ccb.h.

Referenced by cam_periph_error(), cam_periph_freeze_after_event(), camperiphdone(), camperiphscsisenseerror(), camperiphscsistatuserror(), and xpt_action().

7.4.1.39 #define SIM_IDLEN 16

Definition at line 48 of file cam_ccb.h.

Referenced by xptaction().

7.4.1.40 #define VUHBALEN 14

Definition at line 47 of file cam_ccb.h.

7.4.1.41 #define XPORT_VERSION_UNKNOWN (UINT_MAX - 1)

Definition at line 232 of file cam_ccb.h.

Referenced by xpt_set_transfer_settings().

7.4.1.42 #define XPORT_VERSION_UNSPECIFIED UINT_MAX

Definition at line 233 of file cam_ccb.h.

Referenced by xpt_set_transfer_settings(), xpt_toggle_tags(), and xptaction().

7.4.1.43 #define XPT_CCB_INVALID -1

Definition at line 872 of file cam_ccb.h.

7.4.1.44 #define XPT_FC_GROUP(op) ((op) & XPT_FC_GROUP_MASK)

Definition at line 198 of file cam_ccb.h.

7.4.1.45 #define XPT_FC_GROUP_COMMON 0x00

Definition at line 199 of file cam_ccb.h.

7.4.1.46 #define XPT_FC_GROUP_HBA_ENGINE 0x20

Definition at line 201 of file cam_ccb.h.

7.4.1.47 #define XPT_FC_GROUP_MASK 0xF0

Definition at line 197 of file cam_ccb.h.

7.4.1.48 #define XPT_FC_GROUP_SCSI_CONTROL 0x10

Definition at line 200 of file cam_ccb.h.

7.4.1.49 #define XPT_FC_GROUP_TMODE 0x30

Definition at line 202 of file cam_ccb.h.

7.4.1.50 #define XPT_FC_GROUP_VENDOR_UNIQUE 0x80

Definition at line 203 of file cam_ccb.h.

7.4.1.51 #define XPT_FC_IS_DEV_QUEUED(ccb) (((ccb) → ccb_h.func_code & XPT_FC_DEV_QUEUED) == XPT_FC_DEV_QUEUED)

Definition at line 205 of file cam_ccb.h.

Referenced by xpt_action().

7.4.1.52 #define XPT_FC_IS_QUEUED(ccb) (((ccb) → ccb_h.func_code & XPT_FC_QUEUED) != 0)

Definition at line 207 of file cam_ccb.h.

Referenced by targfreccb(), targsendccb(), and xpt_action().

7.4.2 Typedef Documentation

7.4.2.1 typedef void [ac_callback_t](#)(void *softc, u_int32_t code, struct [cam_path](#) *path, void *args)

Definition at line 658 of file cam_ccb.h.

7.4.3 Enumeration Type Documentation

7.4.3.1 enum [ac_code](#)

Enumerator:

AC_GETDEV_CHANGED
AC_INQ_CHANGED
AC_TRANSFER_NEG
AC_LOST_DEVICE
AC_FOUND_DEVICE
AC_PATH_DEREGISTERED
AC_PATH_REGISTERED
AC_SENT_BDR
AC_SCSI_AEN
AC_UNSOL_RESEL
AC_BUS_RESET

Definition at line 644 of file cam_ccb.h.

7.4.3.2 enum [bus_pattern_flags](#)

Enumerator:

BUS_MATCH_NONE
BUS_MATCH_PATH
BUS_MATCH_NAME
BUS_MATCH_UNIT
BUS_MATCH_BUS_ID
BUS_MATCH_ANY

Definition at line 361 of file cam_ccb.h.

7.4.3.3 enum [cam_proto](#)

Enumerator:

PROTO_UNKNOWN
PROTO_UNSPECIFIED
PROTO_SCSI
PROTO_ATA
PROTO_ATAPI

Definition at line 210 of file cam_ccb.h.

7.4.3.4 enum `cam_xport`

Enumerator:

XPORT_UNKNOWN
XPORT_UNSPECIFIED
XPORT_SPI
XPORT_FC
XPORT_SSA
XPORT_USB
XPORT_PPB
XPORT_ATA
XPORT_SAS

Definition at line 218 of file `cam_ccb.h`.

7.4.3.5 enum `ccb_dev_match_status`

Enumerator:

CAM_DEV_MATCH_LAST
CAM_DEV_MATCH_MORE
CAM_DEV_MATCH_LIST_CHANGED
CAM_DEV_MATCH_SIZE_ERROR
CAM_DEV_MATCH_ERROR

Definition at line 434 of file `cam_ccb.h`.

7.4.3.6 enum `ccb_flags`

Enumerator:

CAM_CDB_POINTER
CAM_QUEUE_ENABLE
CAM_CDB_LINKED
CAM_NEGOTIATE
CAM_SCATTER_VALID
CAM_DIS_AUTOSENSE
CAM_DIR_RESV
CAM_DIR_IN
CAM_DIR_OUT
CAM_DIR_NONE
CAM_DIR_MASK
CAM_SOFT_RST_OP
CAM_ENG_SYNC
CAM_DEV_QFRZDIS

CAM_DEV_QFREEZE
CAM_HIGH_POWER
CAM_SENSE_PTR
CAM_SENSE_PHYS
CAM_TAG_ACTION_VALID
CAM_PASS_ERR_RECOVER
CAM_DIS_DISCONNECT
CAM_SG_LIST_PHYS
CAM_MSG_BUF_PHYS
CAM_SNS_BUF_PHYS
CAM_DATA_PHYS
CAM_CDB_PHYS
CAM_ENG_SGLIST
CAM_DIS_AUTOSRP
CAM_DIS_AUTODISC
CAM_TGT_CCB_AVAIL
CAM_TGT_PHASE_MODE
CAM_MSGB_VALID
CAM_STATUS_VALID
CAM_DATAB_VALID
CAM_SEND_SENSE
CAM_TERM_IO
CAM_DISCONNECT
CAM_SEND_STATUS

Definition at line 58 of file cam_ccb.h.

7.4.3.7 enum [ccb_getdevlist_status_e](#)

Enumerator:

CAM_GDEVLIST_LAST_DEVICE
CAM_GDEVLIST_LIST_CHANGED
CAM_GDEVLIST_MORE_DEVS
CAM_GDEVLIST_ERROR

Definition at line 309 of file cam_ccb.h.

7.4.3.8 enum [cts_type](#)

Enumerator:

CTS_TYPE_CURRENT_SETTINGS
CTS_TYPE_USER_SETTINGS

Definition at line 699 of file cam_ccb.h.

7.4.3.9 enum [dev_match_type](#)

Enumerator:

DEV_MATCH_PERIPH
DEV_MATCH_DEVICE
DEV_MATCH_BUS

Definition at line 384 of file cam_ccb.h.

7.4.3.10 enum [dev_pattern_flags](#)

Enumerator:

DEV_MATCH_NONE
DEV_MATCH_PATH
DEV_MATCH_TARGET
DEV_MATCH_LUN
DEV_MATCH_INQUIRY
DEV_MATCH_ANY

Definition at line 344 of file cam_ccb.h.

7.4.3.11 enum [dev_pos_type](#)

Enumerator:

CAM_DEV_POS_NONE
CAM_DEV_POS_BUS
CAM_DEV_POS_TARGET
CAM_DEV_POS_DEVICE
CAM_DEV_POS_PERIPH
CAM_DEV_POS_PDPTR
CAM_DEV_POS_TYEMASK
CAM_DEV_POS_EDT
CAM_DEV_POS_PDRV

Definition at line 442 of file cam_ccb.h.

7.4.3.12 enum [dev_result_flags](#)

Enumerator:

DEV_RESULT_NOFLAG
DEV_RESULT_UNCONFIGURED

Definition at line 403 of file cam_ccb.h.

7.4.3.13 enum [ei_algo](#)

Enumerator:

EAD_VUNIQUE
EAD_LZ1V1
EAD_LZ2V1
EAD_LZ2V2

Definition at line 829 of file cam_ccb.h.

7.4.3.14 enum [ei_type](#)

Enumerator:

EIT_BUFFER
EIT_LOSSLESS
EIT_LOSSY
EIT_ENCRYPT

Definition at line 822 of file cam_ccb.h.

7.4.3.15 enum [periph_pattern_flags](#)

Enumerator:

PERIPH_MATCH_NONE
PERIPH_MATCH_PATH
PERIPH_MATCH_TARGET
PERIPH_MATCH_LUN
PERIPH_MATCH_NAME
PERIPH_MATCH_UNIT
PERIPH_MATCH_ANY

Definition at line 325 of file cam_ccb.h.

7.4.3.16 enum [pi_inqflag](#)

Enumerator:

PI_MDP_ABLE
PI_WIDE_32
PI_WIDE_16
PI_SDTR_ABLE
PI_LINKED_CDB
PI_TAG_ABLE
PI_SOFT_RST

Definition at line 489 of file cam_ccb.h.

7.4.3.17 enum [pi_miscflag](#)

Enumerator:

PIM_SCANHILO
PIM_NOREMOVE
PIM_NOINITIATOR
PIM_NOBUSRESET
PIM_NO_6_BYTE
PIM_SEQSCAN

Definition at line 508 of file cam_ccb.h.

7.4.3.18 enum [pi_tmflag](#)

Enumerator:

PIT_PROCESSOR
PIT_PHASE
PIT_DISCONNECT
PIT_TERM_IO
PIT_GRP_6
PIT_GRP_7

Definition at line 499 of file cam_ccb.h.

7.4.3.19 enum [xpt_opcode](#)

Enumerator:

XPT_FC_QUEUED
XPT_FC_USER_CCB
XPT_FC_XPT_ONLY
XPT_FC_DEV_QUEUED
XPT_NOOP
XPT_SCSI_IO
XPT_GDEV_TYPE
XPT_GDEVLIST
XPT_PATH_INQ
XPT_REL_SIMQ
XPT_SASYNC_CB
XPT_SDEV_TYPE
XPT_SCAN_BUS
XPT_DEV_MATCH
XPT_DEBUG
XPT_PATH_STATS

XPT_GDEV_STATS
XPT_ABORT
XPT_RESET_BUS
XPT_RESET_DEV
XPT_TERM_IO
XPT_SCAN_LUN
XPT_GET_TRAN_SETTINGS
XPT_SET_TRAN_SETTINGS
XPT_CALC_GEOMETRY
XPT_ENG_INQ
XPT_ENG_EXEC
XPT_EN_LUN
XPT_TARGET_IO
XPT_ACCEPT_TARGET_IO
XPT_CONT_TARGET_IO
XPT_IMMED_NOTIFY
XPT_NOTIFY_ACK
XPT_VUNIQUE

Definition at line 107 of file cam_ccb.h.

7.4.4 Function Documentation

7.4.4.1 void cam_calc_geometry (struct **ccb_calc_geometry** * *cgc*, int *extended*)

Definition at line 372 of file cam.c.

References `ccb_calc_geometry::block_size`, `CAM_REQ_CMP`, `CAM_REQ_CMP_ERR`, `ccb_calc_geometry::ccb_h`, `ccb_calc_geometry::cylinders`, `ccb_calc_geometry::heads`, `L`, `ccb_calc_geometry::secs_per_track`, `ccb_hdr::status`, and `ccb_calc_geometry::volume_size`.

7.4.4.2 static **__inline** void cam_fill_csio (struct **ccb_scsiio** * *csio*, u_int32_t *retries*, void(*) (struct **cam_periph** *, union **ccb** *) *cbfcn*, u_int32_t *flags*, u_int8_t *tag_action*, u_int8_t * *data_ptr*, u_int32_t *dxfer_len*, u_int8_t *sense_len*, u_int8_t *cdb_len*, u_int32_t *timeout*) [static]

Definition at line 926 of file cam_ccb.h.

References `ccb_hdr::cbfcn`, `ccb_scsiio::ccb_h`, `ccb_scsiio::cdb_len`, `ccb_scsiio::data_ptr`, `ccb_scsiio::dxfer_len`, `ccb_hdr::flags`, `ccb_hdr::func_code`, `ccb_hdr::retry_count`, `ccb_scsiio::sense_len`, `ccb_scsiio::tag_action`, `ccb_hdr::timeout`, and `XPT SCSI IO`.

Referenced by `cdpause()`, `cdplay()`, `cdplaymsf()`, `cdplaytracks()`, `cdreadsubchannel()`, `cdreadtoc()`, `cdsetspeed()`, `scsi_erase()`, `scsi_exchange_medium()`, `scsi_initialize_element_status()`, `scsi_inquiry()`, `scsi_load_unload()`, `scsi_log_select()`, `scsi_log_sense()`, `scsi_mode_select_len()`, `scsi_mode_sense_len()`, `scsi_move_medium()`, `scsi_position_to_element()`, `scsi_prevent()`, `scsi_read_block_limits()`, `scsi_read_capacity()`, `scsi_read_capacity_16()`, `scsi_read_dvd_structure()`, `scsi_read_element_status()`, `scsi_read_position()`, `scsi_read_write()`, `scsi_report_key()`, `scsi_report_luns()`, `scsi_request_sense()`, `scsi_reserve_release_unit()`, `scsi_rewind()`, `scsi_sa_read_write()`, `scsi_send_key()`, `scsi_send_receive()`, `scsi_send_volume_tag()`, `scsi_set_position()`, `scsi_space()`, `scsi_start_stop()`, `scsi_synchronize_cache()`, `scsi_test_unit_ready()`, `scsi_write_filemarks()`, and `ses_runcmd()`.

7.4.4.3 `static __inline void cam_fill_ctio (struct ccb_scsiio * csio, u_int32_t retries, void(*) (struct cam_periph *, union ccb *) cbfnp, u_int32_t flags, u_int tag_action, u_int tag_id, u_int init_id, u_int scsi_status, u_int8_t * data_ptr, u_int32_t dxfer_len, u_int32_t timeout)`
[static]

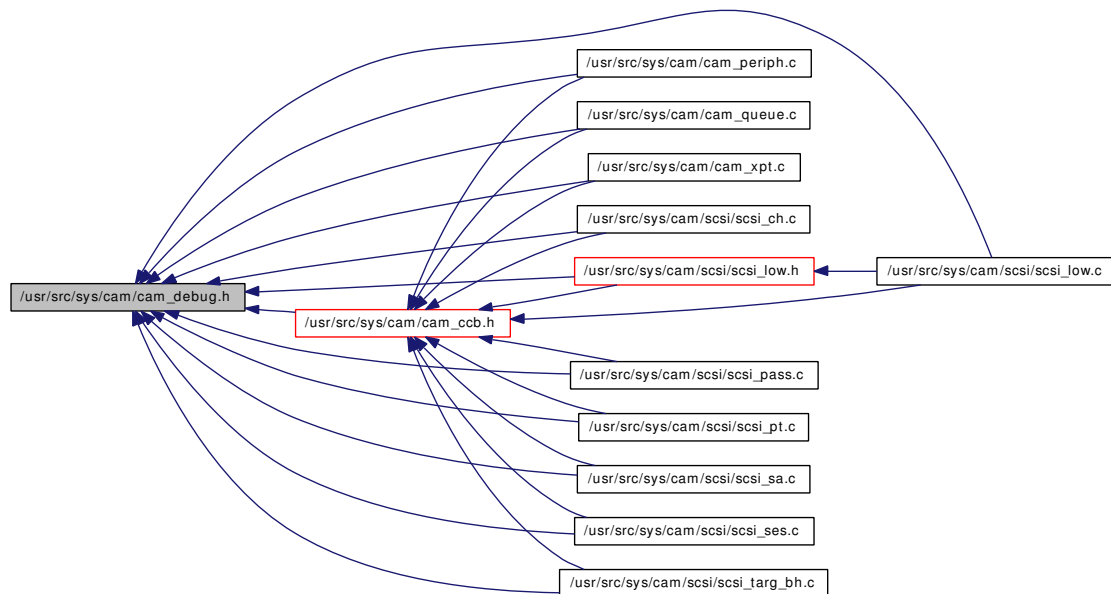
Definition at line 946 of file `cam_ccb.h`.

References `ccb_hdr::cbfnp`, `ccb_scsiio::ccb_h`, `ccb_scsiio::data_ptr`, `ccb_scsiio::dxfer_len`, `ccb_hdr::flags`, `ccb_hdr::func_code`, `ccb_scsiio::init_id`, `ccb_hdr::retry_count`, `ccb_scsiio::scsi_status`, `ccb_scsiio::tag_action`, `ccb_scsiio::tag_id`, `ccb_hdr::timeout`, and `XPT_CONT_TARGET_IO`.

Referenced by `targbhstart()`.

7.5 /usr/src/sys/cam/cam_debug.h File Reference

This graph shows which files directly or indirectly include this file:



Defines

- `#define _CAM_CAM_DEBUG_H 1`
- `#define CAM_DEBUGGED(A, B) 0`
- `#define CAM_DEBUG(A, B, C)`
- `#define CAM_DEBUG_PRINT(A, B)`

Enumerations

- `enum cam_debug_flags {`
`CAM_DEBUG_NONE = 0x00, CAM_DEBUG_INFO = 0x01, CAM_DEBUG_TRACE = 0x02,`
`CAM_DEBUG_SUBTRACE = 0x04,`
`CAM_DEBUG_CDB = 0x08, CAM_DEBUG_XPT = 0x10, CAM_DEBUG_PERIPH = 0x20 }`

7.5.1 Define Documentation

7.5.1.1 `#define _CAM_CAM_DEBUG_H 1`

Definition at line 31 of file `cam_debug.h`.

7.5.1.2 `#define CAM_DEBUG(A, B, C)`

Definition at line 82 of file `cam_debug.h`.

Referenced by `abort_all_pending()`, `cam_periph_getccb()`, `camisr()`, `cddone()`, `cdioctl()`, `cdopen()`, `cdprevent()`, `cdsize()`, `cdstart()`, `cdstrategy()`, `chioctl()`, `probedone()`, `proberequestbackoff()`, `probstart()`, `ptopen()`, `saclose()`, `sadone()`, `saerror()`, `saiioctl()`, `saopen()`, `sastart()`, `sastrategy()`, `scsi_read_write()`, `sesioctl()`, `SLIST_HEAD()`, `targbhdone()`, `targbhstart()`, `targdisable()`, `targdone()`, `targgetccb()`, `targread()`, `targreturnccb()`, `targsendccb()`, `targstart()`, `targusermerge()`, `targwrite()`, `xpt_action()`, `xpt_async()`, `xpt_compile_path()`, `xpt_done()`, `xpt_finishconfig()`, `xpt_free_path()`, `xpt_release_path()`, `xpt_scan_bus()`, `xpt_scan_lun()`, `xpt_schedule()`, `xpt_setup_ccb()`, `xptaction()`, and `xptconfigfunc()`.

7.5.1.3 #define CAM_DEBUG_PRINT(A, B)

Definition at line 83 of file `cam_debug.h`.

Referenced by `dastart()`, `ptstart()`, `sastart()`, `targfreeccb()`, `xpt_release_ccb()`, `xpt_run_dev_allocq()`, `xpt_run_dev_sendq()`, and `xpt_schedule_dev()`.

7.5.1.4 #define CAM_DEBUGGED(A, B) 0

Definition at line 81 of file `cam_debug.h`.

Referenced by `sagetparams()`, `samount()`, `saprevent()`, and `sasetparams()`.

7.5.2 Enumeration Type Documentation

7.5.2.1 enum [cam_debug_flags](#)

Enumerator:

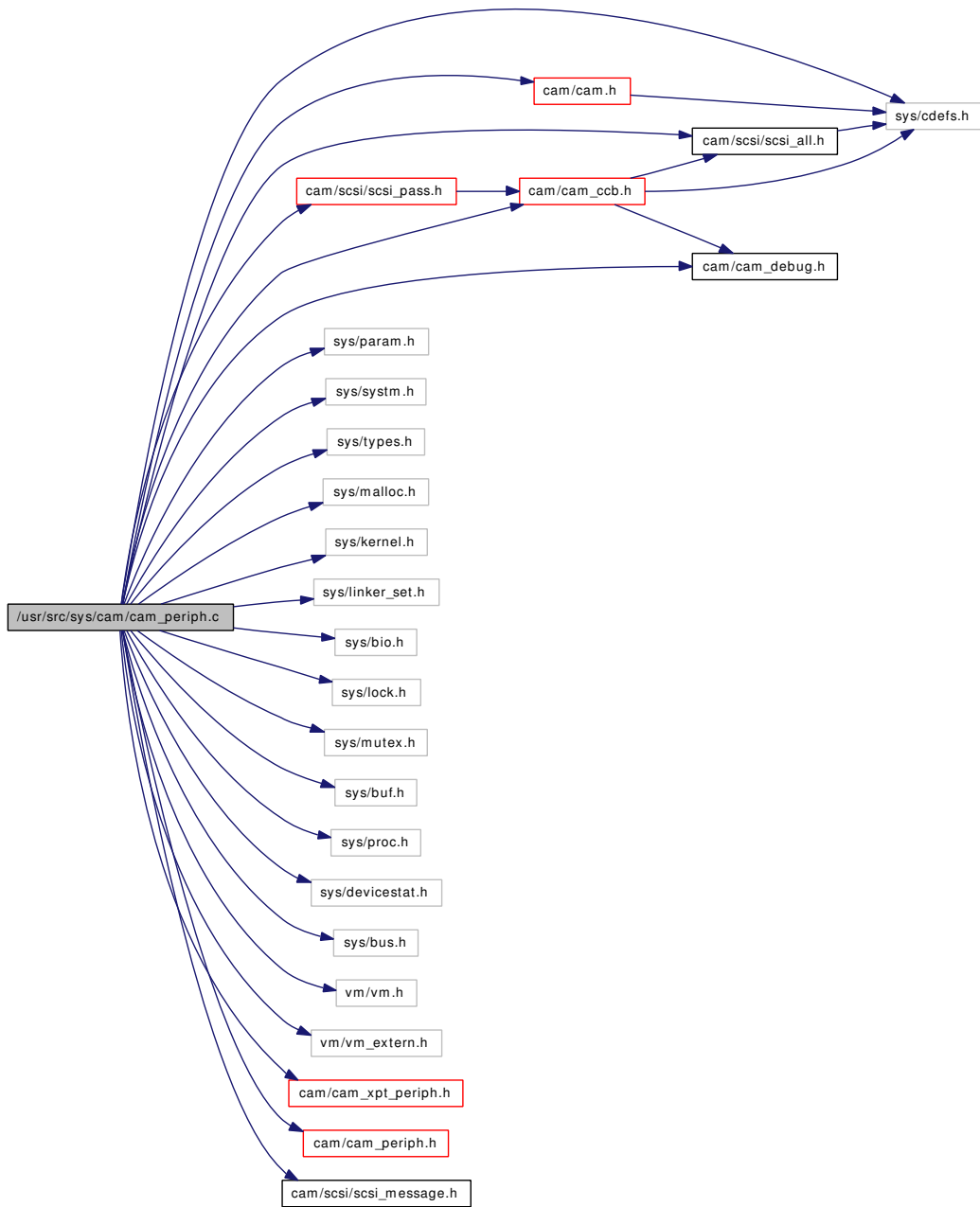
CAM_DEBUG_NONE
CAM_DEBUG_INFO
CAM_DEBUG_TRACE
CAM_DEBUG_SUBTRACE
CAM_DEBUG_CDB
CAM_DEBUG_XPT
CAM_DEBUG_PERIPH

Definition at line 36 of file `cam_debug.h`.

7.6 /usr/src/sys/cam/cam_periph.c File Reference

```
#include <sys/cdefs.h>
#include <sys/param.h>
#include <sys/system.h>
#include <sys/types.h>
#include <sys/malloc.h>
#include <sys/kernel.h>
#include <sys/linker_set.h>
#include <sys/bio.h>
#include <sys/lock.h>
#include <sys/mutex.h>
#include <sys/buf.h>
#include <sys/proc.h>
#include <sys/devicestat.h>
#include <sys/bus.h>
#include <vm/vm.h>
#include <vm/vm_extern.h>
#include <cam/cam.h>
#include <cam/cam_ccb.h>
#include <cam/cam_xpt_periph.h>
#include <cam/cam_periph.h>
#include <cam/cam_debug.h>
#include <cam/scsi/scsi_all.h>
#include <cam/scsi/scsi_message.h>
#include <cam/scsi/scsi_pass.h>
```

Include dependency graph for cam_periph.c:



Defines

- #define `saved_ccb_ptr` `ppriv_ptr0`

Functions

- `__FBSDID` ("FreeBSD: src/sys/cam/cam_periph.c,v 1.64 2006/12/05 07:45:27 mjacob Exp \$")

- static u_int `camperiphnextunit` (struct `periph_driver` *p_drv, u_int newunit, int wired, `path_id_t` pathid, `target_id_t` target, `lun_id_t` lun)
- static u_int `camperiphunit` (struct `periph_driver` *p_drv, `path_id_t` pathid, `target_id_t` target, `lun_id_t` lun)
- static void `camperiphdone` (struct `cam_periph` *periph, union `ccb` *done_ccb)
- static void `camperiphfree` (struct `cam_periph` *periph)
- static int `camperiphscsisstatuserror` (union `ccb` *ccb, `cam_flags` camflags, u_int32_t sense_flags, union `ccb` *save_ccb, int *openings, u_int32_t *relsim_flags, u_int32_t *timeout)
- static int `camperiphscsisenseerror` (union `ccb` *ccb, `cam_flags` camflags, u_int32_t sense_flags, union `ccb` *save_ccb, int *openings, u_int32_t *relsim_flags, u_int32_t *timeout)
- `MALLOC_DEFINE` (M_CAMPERIPH, "CAM periph", "CAM peripheral buffers")
- `TUNABLE_INT` ("kern.cam.periph_selto_delay",&periph_selto_delay)
- `TUNABLE_INT` ("kern.cam.periph_noresrc_delay",&periph_noresrc_delay)
- `TUNABLE_INT` ("kern.cam.periph_busy_delay",&periph_busy_delay)
- void `periphdriver_register` (void *data)
- `cam_status` `cam_periph_alloc` (`periph_ctor_t` *periph_ctor, `periph_oninv_t` *periph_oninvalidate, `periph_dtor_t` *periph_dtor, `periph_start_t` *periph_start, char *name, `cam_periph_type` type, struct `cam_path` *path, `ac_callback_t` *ac_callback, `ac_code` code, void *arg)
- `cam_periph` * `cam_periph_find` (struct `cam_path` *path, char *name)
- `cam_status` `cam_periph_acquire` (struct `cam_periph` *periph)
- void `cam_periph_release` (struct `cam_periph` *periph)
- void `cam_periph_invalidate` (struct `cam_periph` *periph)
- int `cam_periph_lock` (struct `cam_periph` *periph, int priority)
- void `cam_periph_unlock` (struct `cam_periph` *periph)
- int `cam_periph_mapmem` (union `ccb` *ccb, struct `cam_periph_map_info` *mapinfo)
- void `cam_periph_unmapmem` (union `ccb` *ccb, struct `cam_periph_map_info` *mapinfo)
- `ccb` * `cam_periph_getccb` (struct `cam_periph` *periph, u_int32_t priority)
- void `cam_periph_ccbwait` (union `ccb` *ccb)
- int `cam_periph_ioctl` (struct `cam_periph` *periph, int cmd, `caddr_t` addr, int(*error_routine)(union `ccb` *ccb, `cam_flags` camflags, u_int32_t sense_flags))
- int `cam_periph_runccb` (union `ccb` *ccb, int(*error_routine)(union `ccb` *ccb, `cam_flags` camflags, u_int32_t sense_flags), `cam_flags` camflags, u_int32_t sense_flags, struct `devstat` *ds)
- void `cam_freeze_devq` (struct `cam_path` *path)
- u_int32_t `cam_release_devq` (struct `cam_path` *path, u_int32_t relsim_flags, u_int32_t openings, u_int32_t timeout, int getcount_only)
- void `cam_periph_async` (struct `cam_periph` *periph, u_int32_t code, struct `cam_path` *path, void *arg)
- void `cam_periph_bus_settle` (struct `cam_periph` *periph, u_int bus_settle)
- void `cam_periph_freeze_after_event` (struct `cam_periph` *periph, struct `timeval` *event_time, u_int duration_ms)
- int `cam_periph_error` (union `ccb` *ccb, `cam_flags` camflags, u_int32_t sense_flags, union `ccb` *save_ccb)

Variables

- static int `nperiph_drivers`
- `periph_driver` ** `periph_drivers`
- static int `periph_selto_delay` = 1000
- static int `periph_noresrc_delay` = 500
- static int `periph_busy_delay` = 500

7.6.1 Define Documentation

7.6.1.1 #define saved_ccb_ptr ppriv_ptr0

Definition at line 933 of file cam_periph.c.

7.6.2 Function Documentation

7.6.2.1 __FBSDID ("\$FreeBSD: src/sys/cam/cam_periph.c, v 1.64 2006/12/05 07:45:27 mjacob Exp \$")

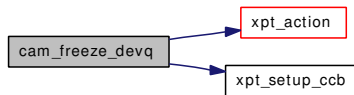
7.6.2.2 void cam_freeze_devq (struct cam_path * path)

Definition at line 905 of file cam_periph.c.

References CAM_DEV_QFREEZE, ccb_hdr::flags, ccb_hdr::func_code, ccb_hdr::path, xpt_action(), XPT_NOOP, and xpt_setup_ccb().

Referenced by cam_periph_freeze_after_event().

Here is the call graph for this function:



7.6.2.3 cam_status cam_periph_acquire (struct cam_periph * periph)

Definition at line 277 of file cam_periph.c.

References CAM_REQ_CMP, CAM_REQ_CMP_ERR, and cam_periph::refcount.

Referenced by cam_periph_lock(), cdopen(), chopen(), passopen(), proberegister(), ptopen(), saopen(), sesopen(), and SLIST_HEAD().

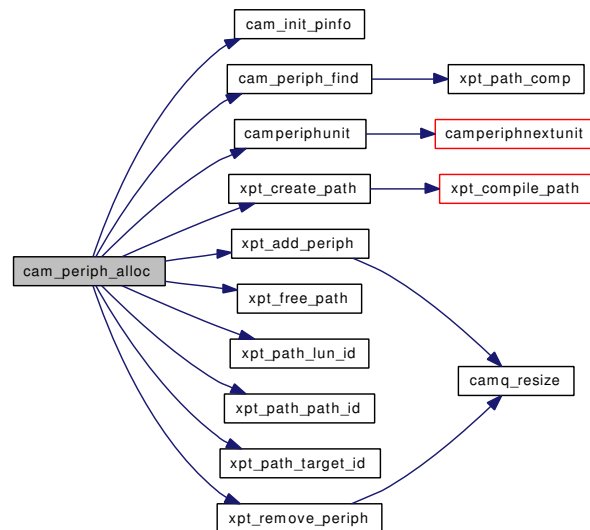
7.6.2.4 cam_status cam_periph_alloc (periph_ctor_t * periph_ctor, periph_oninv_t * periph_oninvalidate, periph_dtor_t * periph_dtor, periph_start_t * periph_start, char * name, cam_periph_type type, struct cam_path * path, ac_callback_t * ac_callback, ac_code code, void * arg)

Definition at line 118 of file cam_periph.c.

References cam_init_pinfo(), cam_periph_find(), CAM_PERIPH_INVALID, CAM_PERIPH_NEW_DEV_FOUND, CAM_PRIORITY_NONE, CAM_REQ_CMP, CAM_REQ_INPROG, CAM_REQ_INVALID, CAM_RESRC_UNAVAIL, camperiphunit(), cam_periph::deferred_ac, cam_periph::deferred_callback, cam_periph::flags, cam_periph::path, periph_drivers, cam_periph::periph_name, cam_periph::unit_number, xpt_add_periph(), xpt_create_path(), xpt_free_path(), xpt_path_lun_id(), xpt_path_path_id(), xpt_path_target_id(), and xpt_remove_periph().

Referenced by cdasync(), chasync(), daasync(), passasync(), ptasync(), saasync(), sesasync(), targbhasync(), targetable(), xpt_init(), and xpt_scan_lun().

Here is the call graph for this function:



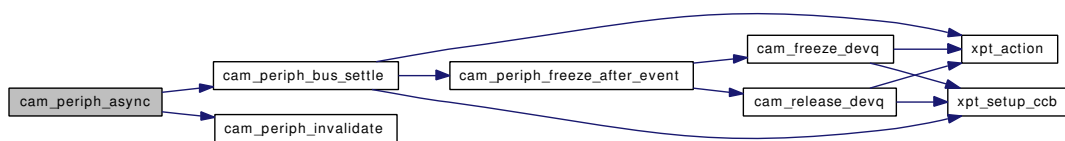
7.6.2.5 void cam_periph_async (struct cam_periph * periph, u_int32_t code, struct cam_path * path, void * arg)

Definition at line 1154 of file cam_periph.c.

References AC_BUS_RESET, AC_LOST_DEVICE, AC_SENT_BDR, cam_periph_bus_settle(), cam_periph_invalidate(), and scsi_delay.

Referenced by cdasync(), chasync(), daasync(), passasync(), ptasync(), saasync(), and sesasync().

Here is the call graph for this function:



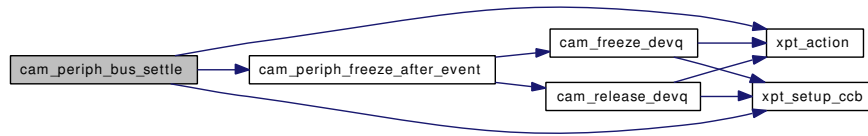
7.6.2.6 void cam_periph_bus_settle (struct cam_periph * periph, u_int bus_settle)

Definition at line 1173 of file cam_periph.c.

References cam_periph_freeze_after_event(), ccb_getdevstats::ccb_h, ccb_hdr::func_code, cam_periph::path, xpt_action(), XPT_GDEV_STATS, and xpt_setup_ccb().

Referenced by cam_periph_async().

Here is the call graph for this function:



7.6.2.7 void cam_periph_ccbwait (union `ccb` * `ccb`)

Definition at line 778 of file `cam_periph.c`.

References `CAM_REQ_INPROG`, `CAM_STATUS_MASK`, `CAM_UNQUEUED_INDEX`, `ccb_hdr::cbfcnp`, `ccb::ccb_h`, `cam_pinfo::index`, `ccb_hdr::pinfo`, and `ccb_hdr::status`.

Referenced by `cam_periph_runccb()`.

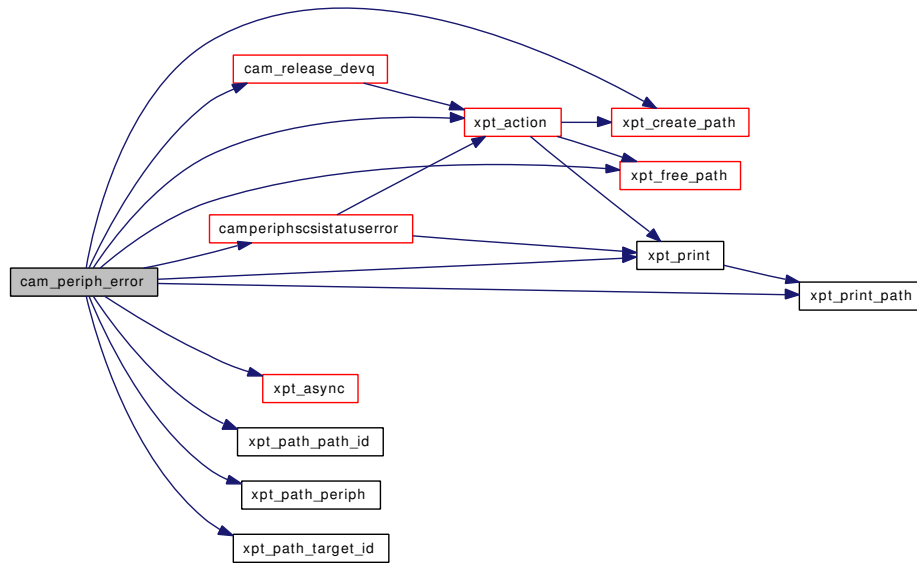
7.6.2.8 int cam_periph_error (union `ccb` * `ccb`, `cam_flags` `camflags`, `u_int32_t` `sense_flags`, union `ccb` * `save_ccb`)

Definition at line 1543 of file `cam_periph.c`.

References `AC_LOST_DEVICE`, `CAM_AUTOSENSE_FAIL`, `CAM_BDR_SENT`, `CAM_BUSY`, `CAM_CMD_TIMEOUT`, `CAM_DATA_RUN_ERR`, `CAM_DEV_NOT_THERE`, `CAM_DEV_QFRZN`, `CAM_LUN_INVALID`, `CAM_LUN_WILDCARD`, `CAM_MSG_REJECT_REC`, `CAM_NO_HBA`, `CAM_PATH_INVALID`, `CAM_PROVIDE_FAIL`, `cam_release_devq()`, `CAM_REQ_CMP`, `CAM_REQ_CMP_ERR`, `CAM_REQ_INVALID`, `CAM_REQ_TOO_BIG`, `CAM_REQUEUE_REQ`, `CAM_RESRC_UNAVAIL`, `CAM_RETRY_SELTO`, `CAM_SCSI_BUS_RESET`, `CAM_SCSI_STATUS_ERROR`, `CAM_SEL_TIMEOUT`, `CAM_STATUS_MASK`, `CAM_TID_INVALID`, `CAM_UA_ABORT`, `CAM_UA_TERMIO`, `CAM_UNCOR_PARITY`, `CAM_UNEXP_BUSFREE`, `camperiphscsistatuserror()`, `ccb::ccb_h`, `ccb_hdr::path`, `periph_busy_delay`, `periph_noresrc_delay`, `periph_selto_delay`, `RELSIM_RELEASE_AFTER_TIMEOUT`, `ccb_hdr::retry_count`, `ccb_hdr::status`, `xpt_action()`, `xpt_async()`, `xpt_create_path()`, `xpt_free_path()`, `xpt_path_path_id()`, `xpt_path_periph()`, `xpt_path_target_id()`, `xpt_print()`, and `xpt_print_path()`.

Referenced by `cderror()`, `cherror()`, `daerror()`, `passerror()`, `probedone()`, `pterror()`, `saerror()`, and `seserror()`.

Here is the call graph for this function:



7.6.2.9 struct **cam_periph*** **cam_periph_find** (struct **cam_path** * *path*, char * *name*)

Definition at line 251 of file `cam_periph.c`.

References `cam_periph::path`, `periph_drivers`, and `xpt_path_comp()`.

Referenced by `cam_periph_alloc()`, `cdregister()`, `targbhasync()`, `targenable()`, and `xpt_scan_lun()`.

Here is the call graph for this function:



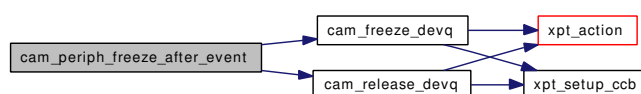
7.6.2.10 void **cam_periph_freeze_after_event** (struct **cam_periph** * *periph*, struct **timeval** * *event_time*, u_int *duration_ms*)

Definition at line 1184 of file `cam_periph.c`.

References `cam_freeze_devq()`, `cam_release_devq()`, `cam_periph::path`, and `RELSIM_RELEASE_AFTER_TIMEOUT`.

Referenced by `cam_periph_bus_settle()`, and `proberegister()`.

Here is the call graph for this function:



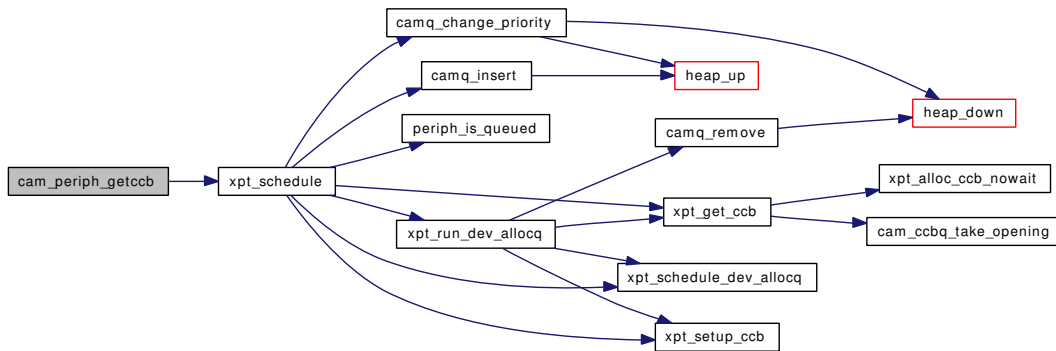
7.6.2.11 union **ccb*** cam_periph_getccb (struct **cam_periph** * *periph*, u_int32_t *priority*)

Definition at line 752 of file cam_periph.c.

References CAM_DEBUG, CAM_DEBUG_TRACE, cam_periph::immediate_priority, cam_periph::path, ccb_hdr::periph_links, and xpt_schedule().

Referenced by cam_periph_ioctl(), cdgetccb(), chexchange(), chgetelemstatus(), chgetparams(), chielem(), chmove(), chposition(), daclose(), dagetcapacity(), daprevent(), passioctl(), saerase(), sagetparams(), sa-loadunload(), samount(), saprevent(), sardpos(), sareserveleaseunit(), saretension(), sarewind(), saset-params(), sasetpos(), saspace(), sawritefilemarks(), and ses_runcmd().

Here is the call graph for this function:



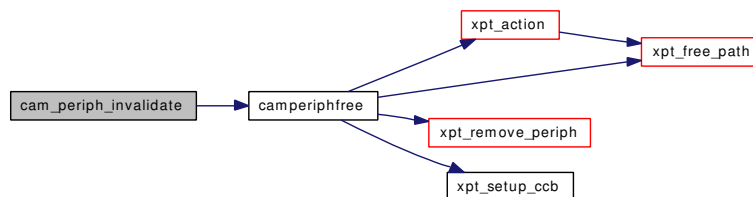
7.6.2.12 void cam_periph_invalidate (struct **cam_periph** * *periph*)

Definition at line 425 of file cam_periph.c.

References CAM_PERIPH_INVALID, CAM_PERIPH_NEW_DEV_FOUND, camperiphfree(), cam_periph::flags, and cam_periph::periph_oninval.

Referenced by cam_periph_async(), cddone(), chdone(), dadone(), probedone(), targbhasync(), targclose(), and targenable().

Here is the call graph for this function:



7.6.2.13 int cam_periph_ioctl (struct **cam_periph** * *periph*, int *cmd*, caddr_t *addr*, int(*) (union **ccb** ***ccb**, **cam_flags** *camflags*, u_int32_t *sense_flags*) *error_routine*)

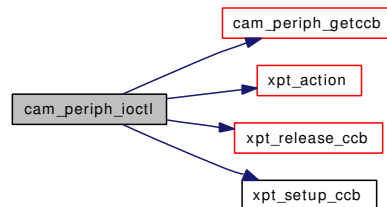
Definition at line 791 of file cam_periph.c.

References CAM_GDEVLIST_LAST_DEVICE, CAM_GDEVLIST_MORE_DEVS, cam_periph_getccb(), CAMGETPASSTHRU, ccb::ccb_h, ccb::cgdl, ccb_hdr::func_code, ccb_getdevlist::index,

ccb_hdr::path, ccb_getdevlist::periph_name, ccb_getdevlist::status, ccb_getdevlist::unit_number, xpt_action(), XPT_GDEVLIST, xpt_release_ccb(), and xpt_setup_ccb().

Referenced by cdiocctl(), chioctl(), passioctl(), ptioctl(), saioctl(), and sesioctl().

Here is the call graph for this function:



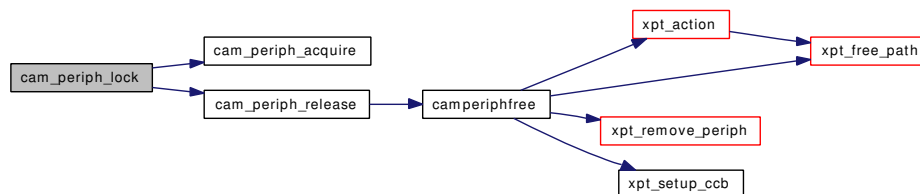
7.6.2.14 int cam_periph_lock (struct [cam_periph](#) * *periph*, int *priority*)

Definition at line 506 of file cam_periph.c.

References cam_periph_acquire(), CAM_PERIPH_LOCK_WANTED, CAM_PERIPH_LOCKED, cam_periph_release(), CAM_REQ_CMP, and cam_periph::flags.

Referenced by cdclose(), cdiocctl(), cdopen(), cdregister(), chclose(), chopen(), chregister(), daclose(), daregister(), passclose(), passopen(), ptclose(), ptioctl(), ptopen(), saclose(), saioctl(), saopen(), sesclose(), sesopen(), and SLIST_HEAD().

Here is the call graph for this function:



7.6.2.15 int cam_periph_mapmem (union [ccb](#) * *ccb*, struct [cam_periph_map_info](#) * *mapinfo*)

Definition at line 554 of file cam_periph.c.

References cam_periph_map_info::bp, CAM_DIR_IN, CAM_DIR_MASK, CAM_DIR_NONE, CAM_DIR_OUT, CAM_PERIPH_MAXMAPS, ccb::ccb_h, ccb::cdm, ccb::csio, ccb_scsiio::data_ptr, ccb_scsiio::dxfer_len, ccb_hdr::flags, ccb_hdr::func_code, ccb_dev_match::match_buf_len, ccb_dev_match::matches, cam_periph_map_info::num_bufs_used, ccb_dev_match::pattern_buf_len, ccb_dev_match::patterns, XPT_CONT_TARGET_IO, XPT_DEV_MATCH, and XPT SCSI_IO.

Referenced by passsendccb(), targsendccb(), and xptioctl().

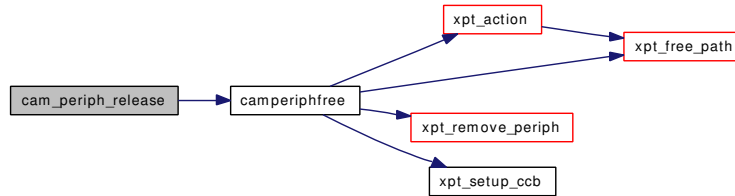
7.6.2.16 void cam_periph_release (struct [cam_periph](#) * *periph*)

Definition at line 292 of file cam_periph.c.

References CAM_PERIPH_INVALID, camperiphfree(), cam_periph::flags, and cam_periph::refcount.

Referenced by `cam_periph_lock()`, `cam_periph_unlock()`, `cdcclose()`, `chclose()`, `chopen()`, `daclose()`, `passclose()`, `probedone()`, `ptclose()`, `saclose()`, `saopen()`, `sesclose()`, `sesopen()`, and `SLIST_HEAD()`.

Here is the call graph for this function:



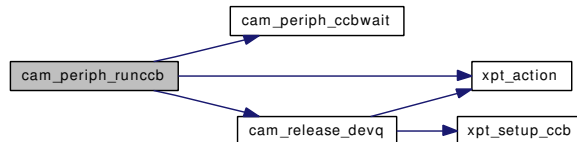
7.6.2.17 `int cam_periph_runccb (union ccb * ccb, int(*)(union ccb * ccb, cam_flags camflags, u_int32_t sense_flags) error_routine, cam_flags camflags, u_int32_t sense_flags, struct devstat * ds)`

Definition at line 853 of file `cam_periph.c`.

References `CAM_DEV_QFRZN`, `CAM_DIR_MASK`, `CAM_DIR_NONE`, `CAM_DIR_OUT`, `cam_periph_ccbwait()`, `cam_release_devq()`, `CAM_REQ_CMP`, `CAM_STATUS_MASK`, `xpt_action()`, and `XPT SCSI_IO`.

Referenced by `cdrunccb()`, `chexchange()`, `chgetelemstatus()`, `chgetparams()`, `chielem()`, `chmove()`, `chposition()`, `daclose()`, `dagetcapacity()`, `daprevent()`, `passsendccb()`, `saerase()`, `sagetparams()`, `saloadunload()`, `samount()`, `saprevent()`, `sardpos()`, `sareservereleaseunit()`, `saretension()`, `sarewind()`, `sasetparams()`, `sasetpos()`, `saspace()`, `sawritefilemarks()`, `ses_runcmd()`, `TAILQ_HEAD()`, and `xptioctl()`.

Here is the call graph for this function:



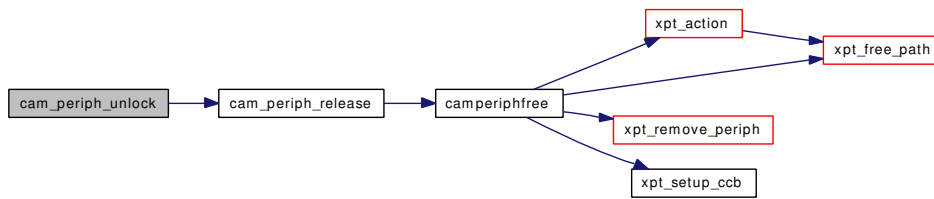
7.6.2.18 `void cam_periph_unlock (struct cam_periph * periph)`

Definition at line 535 of file `cam_periph.c`.

References `CAM_PERIPH_LOCK_WANTED`, `CAM_PERIPH_LOCKED`, `cam_periph_release()`, and `cam_periph::flags`.

Referenced by `cdcclose()`, `cddone()`, `cdioctl()`, `cdopen()`, `chclose()`, `chdone()`, `chopen()`, `daclose()`, `dadone()`, `passclose()`, `passopen()`, `ptclose()`, `ptioctl()`, `ptopen()`, `saclose()`, `saioctl()`, `saopen()`, `sesclose()`, `sesopen()`, and `SLIST_HEAD()`.

Here is the call graph for this function:



7.6.2.19 void cam_periph_unmapmem (union **ccb** * *ccb*, struct **cam_periph_map_info** * *mapinfo*)

Definition at line 702 of file cam_periph.c.

References cam_periph_map_info::bp, CAM_PERIPH_MAXMAPS, ccb::ccb_h, ccb::cdm, ccb::csio, ccb_scsiio::data_ptr, ccb_hdr::func_code, ccb_dev_match::matches, cam_periph_map_info::num_bufs_-, used, ccb_dev_match::patterns, XPT_CONT_TARGET_IO, XPT_DEV_MATCH, and XPT_SCSI_IO.

Referenced by passsendccb(), targreturnccb(), and xptioctl().

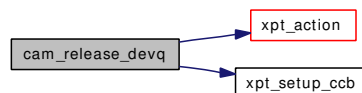
7.6.2.20 u_int32_t cam_release_devq (struct **cam_path** * *path*, u_int32_t *relsim_flags*, u_int32_t *openings*, u_int32_t *timeout*, int *getcount_only*)

Definition at line 916 of file cam_periph.c.

References CAM_DEV_QFREEZE, ccb_relsim::ccb_h, ccb_hdr::flags, ccb_hdr::func_code, ccb_relsim::openings, ccb_relsim::release_flags, ccb_relsim::release_timeout, xpt_action(), XPT_REL_SIMQ, and xpt_setup_ccb().

Referenced by cam_periph_error(), cam_periph_freeze_after_event(), cam_periph_runccb(), cam_periph_done(), cd6byteworkaround(), cddone(), chdone(), cmd6workaround(), daclose(), dadone(), dagetcapacity(), dashutdown(), ptdone(), sadone(), saerase(), sardpos(), saretension(), sarewind(), sasetpos(), saspace(), sawritefilemarks(), ses_runcmd(), targbhdone(), and targbhstart().

Here is the call graph for this function:



7.6.2.21 static void cam_periphdone (struct **cam_periph** * *periph*, union **ccb** * *done_ccb*) [static]

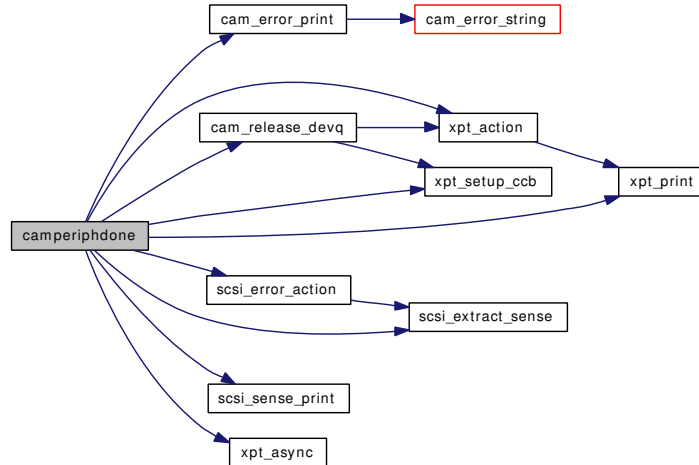
Definition at line 935 of file cam_periph.c.

References AC_INQ_CHANGED, CAM_AUTOSNS_VALID, CAM_DEV_QFRZN, CAM_EPF_ALL, cam_error_print(), CAM_ESF_ALL, CAM_PERIPH_RECOVERY_INPROG, cam_release_devq(), CAM_REQ_CMP, CAM_SCSI_STATUS_ERROR, CAM_STATUS_MASK, ccb_hdr::cbfcnp, ccb_getdev::ccb_h, ccb::ccb_h, cdb_t::cdb_bytes, ccb_scsiio::cdb_io, ccb::csio, scsi_sense_data::error_code, cam_periph::flags, scsi_sense_data::flags, ccb_hdr::func_code, scsi_start_stop_unit::how, scsi_start_stop_unit::opcode, ccb_hdr::path, RELSIM_RELEASE_AFTER_TIMEOUT, REQUEST_SENSE, ccb_hdr::retry_count, scsi_error_action(), scsi_extract_sense(), scsi_sense_print(), ccb_scsiio::sense_data,

SS_FAIL, SS_MASK, SSD_KEY, SSD_KEY_NO_SENSE, SSS_LOEJ, START_STOP_UNIT, ccb_hdr::status, xpt_action(), xpt_async(), XPT_GDEV_TYPE, xpt_print(), XPT SCSI_IO, and xpt_setup_ccb().

Referenced by camperiphscsisenseerror().

Here is the call graph for this function:



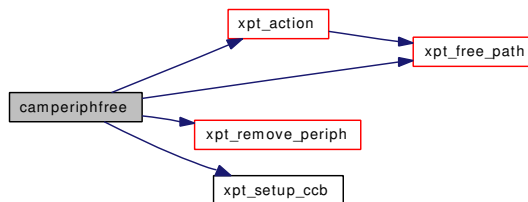
7.6.2.22 static void camperiphfree (struct cam_periph * periph) [static]

Definition at line 450 of file cam_periph.c.

References AC_FOUND_DEVICE, AC_PATH_REGISTERED, CAM_PERIPH_NEW_DEV_FOUND, ccb::ccb_h, ccb_hdr::func_code, ccb_hdr::path, periph_drivers, cam_periph::periph_dtor, cam_periph::periph_name, xpt_action(), xpt_free_path(), XPT_GDEV_TYPE, XPT_PATH_INQ, xpt_remove_periph(), and xpt_setup_ccb().

Referenced by cam_periph_invalidate(), and cam_periph_release().

Here is the call graph for this function:



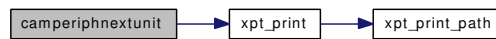
7.6.2.23 static u_int camperiphnextunit (struct periph_driver * p_drv, u_int newunit, int wired, path_id_t pathid, target_id_t target, lun_id_t lun) [static]

Definition at line 317 of file cam_periph.c.

References periph_driver::driver_name, cam_periph::path, cam_periph::periph_name, cam_periph::unit_number, and xpt_print().

Referenced by camperiphunit().

Here is the call graph for this function:



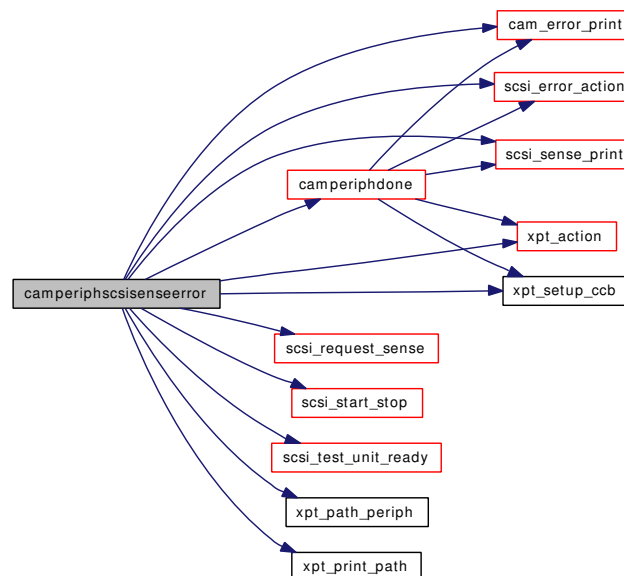
7.6.2.24 `static int camperiphscsisenseerror (union ccb * ccb, cam_flags camflags, u_int32_t sense_flags, union ccb * save_ccb, int * openings, u_int32_t * relsim_flags, u_int32_t * timeout)` [static]

Definition at line 1328 of file cam_periph.c.

References CAM_AUTOSNS_VALID, CAM_DEV_QFREEZE, CAM_DIS_AUTOSENSE, CAM_EPF_ALL, cam_error_print(), CAM_ESF_ALL, CAM_PERIPH_RECOVERY_INPROG, CAM_TAG_ACTION_NONE, camperiphdone(), ccb_getdev::ccb_h, ccb::ccb_h, ccb::cgd, ccb::csio, cam_periph::flags, ccb_hdr::func_code, ccb_getdev::inq_data, MSG_SIMPLE_Q_TAG, ccb_hdr::path, RELSIM_RELEASE_AFTER_TIMEOUT, scsi_error_action(), scsi_request_sense(), scsi_sense_print(), scsi_start_stop(), scsi_test_unit_ready(), ccb_scsiio::sense_data, SID_IS_REMOVABLE, SS_ERRMASK, SS_FAIL, SS_MASK, SS_NOP, SS_REQSENSE, SS_RETRY, SS_START, SS_TUR, SSD_FULL_SIZE, SSQ_DECREMENT_COUNT, SSQ_MANY, SSQ_PRINT_SENSE, xpt_action(), XPT_GDEV_TYPE, xpt_path_periph(), xpt_print_path(), and xpt_setup_ccb().

Referenced by camperiphscsisstatuserror().

Here is the call graph for this function:



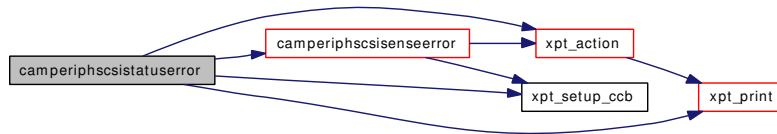
7.6.2.25 `static int camperiphscsisstatuserror (union ccb * ccb, cam_flags camflags, u_int32_t sense_flags, union ccb * save_ccb, int * openings, u_int32_t * relsim_flags, u_int32_t * timeout)` [static]

Definition at line 1213 of file cam_periph.c.

References `camperiphscsisenseerror()`, `ccb::ccb_h`, `ccb_getdevstats::ccb_h`, `ccb::csio`, `ccb_hdr::func_code`, `ccb_hdr::path`, `RELSIM_ADJUST_OPENINGS`, `RELSIM_RELEASE_AFTER_CMDCMPLT`, `RELSIM_RELEASE_AFTER_TIMEOUT`, `ccb_hdr::retry_count`, `ccb_scsiio::scsi_status`, `SCSI_STATUS_BUSY`, `SCSI_STATUS_CHECK_COND`, `SCSI_STATUS_CMD_TERMINATED`, `SCSI_STATUS_COND_MET`, `SCSI_STATUS_INTERMED`, `SCSI_STATUS_INTERMED_COND_MET`, `SCSI_STATUS_OK`, `SCSI_STATUS_QUEUE_FULL`, `SCSI_STATUS_RESERV_CONFLICT`, `xpt_action()`, `XPT_GDEV_STATS`, `xpt_print()`, and `xpt_setup_ccb()`.

Referenced by `cam_periph_error()`.

Here is the call graph for this function:



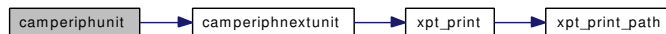
7.6.2.26 `static u_int camperiphunit (struct periph_driver * p_drv, path_id_t pathid, target_id_t target, lun_id_t lun)` [static]

Definition at line 377 of file `cam_periph.c`.

References `camperiphnextunit()`, `periph_driver::driver_name`, and `cam_periph::periph_name`.

Referenced by `cam_periph_alloc()`.

Here is the call graph for this function:



7.6.2.27 `MALLOC_DEFINE (M_CAMPERIPH, "CAM periph", "CAM peripheral buffers")`

7.6.2.28 `void periphdriver_register (void * data)`

Definition at line 98 of file `cam_periph.c`.

References `nperiph_drivers`, and `periph_drivers`.

7.6.2.29 TUNABLE_INT ("kern.cam.periph_busy_delay", & *periph_busy_delay*)

7.6.2.30 TUNABLE_INT ("kern.cam.periph_noresrc_delay", & *periph_noresrc_delay*)

7.6.2.31 TUNABLE_INT ("kern.cam.periph_selto_delay", & *periph_selto_delay*)

7.6.3 Variable Documentation

7.6.3.1 int **nperiph_drivers** [static]

Definition at line 84 of file cam_periph.c.

Referenced by periphdriver_register().

7.6.3.2 int **periph_busy_delay = 500** [static]

Definition at line 93 of file cam_periph.c.

Referenced by cam_periph_error().

7.6.3.3 struct **periph_driver** periph_drivers**

Definition at line 85 of file cam_periph.c.

Referenced by cam_periph_alloc(), cam_periph_find(), camperiphfree(), periphdriver_register(), xpt_finishconfig(), xptioctl(), xptdrvtraverse(), and xptplistperiphfunc().

7.6.3.4 int **periph_noresrc_delay = 500** [static]

Definition at line 91 of file cam_periph.c.

Referenced by cam_periph_error().

7.6.3.5 int **periph_selto_delay = 1000** [static]

Definition at line 89 of file cam_periph.c.

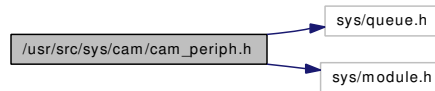
Referenced by cam_periph_error().

7.7 /usr/src/sys/cam/cam_periph.h File Reference

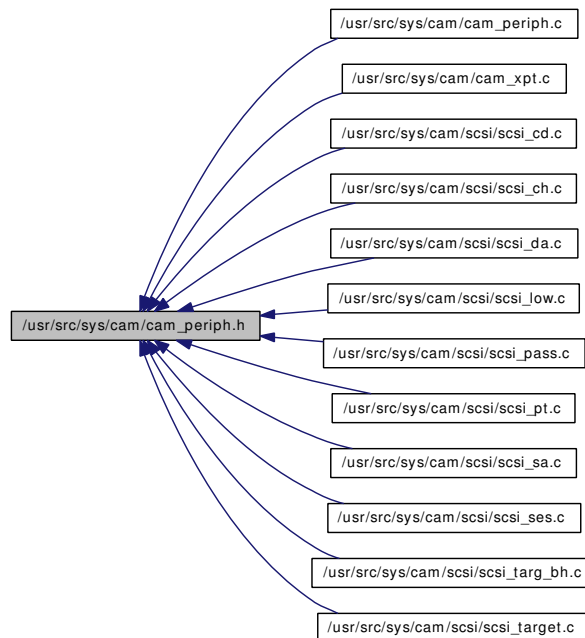
```
#include <sys/queue.h>
```

```
#include <sys/module.h>
```

Include dependency graph for cam_periph.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- struct [periph_driver](#)
- struct [cam_periph](#)
- struct [cam_periph_map_info](#)

Defines

- #define [_CAM_CAM_PERIPH_H](#) 1
- #define [PERIPHDRIVER_DECLARE](#)(name, driver)
- #define [ppriv_ptr0](#) periph_priv.entries[0].ptr
- #define [ppriv_ptr1](#) periph_priv.entries[1].ptr
- #define [ppriv_field0](#) periph_priv.entries[0].field
- #define [ppriv_field1](#) periph_priv.entries[1].field

- #define CAM_PERIPH_RUNNING 0x01
- #define CAM_PERIPH_LOCKED 0x02
- #define CAM_PERIPH_LOCK_WANTED 0x04
- #define CAM_PERIPH_INVALID 0x08
- #define CAM_PERIPH_NEW_DEV_FOUND 0x10
- #define CAM_PERIPH_RECOVERY_INPROG 0x20
- #define CAM_PERIPH_MAXMAPS 2

Typedefs

- typedef void(periph_init_t)(void)
- typedef periph_init_t * periph_init_func_t
- typedef void periph_start_t (struct cam_periph *periph, union ccb *start_ccb)
- typedef cam_status periph_ctor_t (struct cam_periph *periph, void *arg)
- typedef void periph_oninv_t (struct cam_periph *periph)
- typedef void periph_dtor_t (struct cam_periph *periph)

Enumerations

- enum cam_periph_type { CAM_PERIPH_BIO }

Functions

- void periphdriver_register (void *)
- cam_status cam_periph_alloc (periph_ctor_t *periph_ctor, periph_oninv_t *periph_oninvalidate, periph_dtor_t *periph_dtor, periph_start_t *periph_start, char *name, cam_periph_type type, struct cam_path *, ac_callback_t *, ac_code, void *arg)
- cam_periph * cam_periph_find (struct cam_path *path, char *name)
- int cam_periph_lock (struct cam_periph *periph, int priority)
- void cam_periph_unlock (struct cam_periph *periph)
- cam_status cam_periph_acquire (struct cam_periph *periph)
- void cam_periph_release (struct cam_periph *periph)
- void cam_periph_invalidate (struct cam_periph *periph)
- int cam_periph_mapmem (union ccb *ccb, struct cam_periph_map_info *mapinfo)
- void cam_periph_unmapmem (union ccb *ccb, struct cam_periph_map_info *mapinfo)
- ccb * cam_periph_getccb (struct cam_periph *periph, u_int32_t priority)
- void cam_periph_ccbwait (union ccb *ccb)
- int cam_periph_runccb (union ccb *ccb, int(*error_routine)(union ccb *ccb, cam_flags camflags, u_int32_t sense_flags), cam_flags camflags, u_int32_t sense_flags, struct devstat *ds)
- int cam_periph_ioctl (struct cam_periph *periph, int cmd, caddr_t addr, int(*error_routine)(union ccb *ccb, cam_flags camflags, u_int32_t sense_flags))
- void cam_freeze_devq (struct cam_path *path)
- u_int32_t cam_release_devq (struct cam_path *path, u_int32_t relsim_flags, u_int32_t opening_reduction, u_int32_t timeout, int getcount_only)
- void cam_periph_async (struct cam_periph *periph, u_int32_t code, struct cam_path *path, void *arg)
- void cam_periph_bus_settle (struct cam_periph *periph, u_int bus_settle_ms)
- void cam_periph_freeze_after_event (struct cam_periph *periph, struct timeval *event_time, u_int duration_ms)
- int cam_periph_error (union ccb *ccb, cam_flags camflags, u_int32_t sense_flags, union ccb *save_ccb)

Variables

- `cam_periph` * `xpt_periph`
- `periph_driver` ** `periph_drivers`

7.7.1 Define Documentation

7.7.1.1 `#define _CAM_CAM_PERIPH_H 1`

Definition at line 32 of file `cam_periph.h`.

7.7.1.2 `#define CAM_PERIPH_INVALID 0x08`

Definition at line 113 of file `cam_periph.h`.

Referenced by `cam_periph_alloc()`, `cam_periph_invalidate()`, and `cam_periph_release()`.

7.7.1.3 `#define CAM_PERIPH_LOCK_WANTED 0x04`

Definition at line 112 of file `cam_periph.h`.

Referenced by `cam_periph_lock()`, and `cam_periph_unlock()`.

7.7.1.4 `#define CAM_PERIPH_LOCKED 0x02`

Definition at line 111 of file `cam_periph.h`.

Referenced by `cam_periph_lock()`, `cam_periph_unlock()`, and `saiocntl()`.

7.7.1.5 `#define CAM_PERIPH_MAXMAPS 2`

Definition at line 125 of file `cam_periph.h`.

Referenced by `cam_periph_mapmem()`, and `cam_periph_unmapmem()`.

7.7.1.6 `#define CAM_PERIPH_NEW_DEV_FOUND 0x10`

Definition at line 114 of file `cam_periph.h`.

Referenced by `cam_periph_alloc()`, `cam_periph_invalidate()`, and `camperiphfree()`.

7.7.1.7 `#define CAM_PERIPH_RECOVERY_INPROG 0x20`

Definition at line 115 of file `cam_periph.h`.

Referenced by `camperiphdone()`, and `camperiphscsisenseerror()`.

7.7.1.8 `#define CAM_PERIPH_RUNNING 0x01`

Definition at line 110 of file `cam_periph.h`.

7.7.1.9 #define PERIPHDRIIVER_DECLARE(name, driver)**Value:**

```
static int name ## _modevent(module_t mod, int type, void *data) \
{ \
    switch (type) { \
        case MOD_LOAD: \
            periphdriver_register(data); \
            break; \
        case MOD_UNLOAD: \
            printf(#name " module unload - not possible for this module type\n"); \
            return EINVAL; \
        default: \
            return EOPNOTSUPP; \
    } \
    return 0; \
} \
static moduledata_t name ## _mod = { \
    #name, \
    name ## _modevent, \
    (void *)&driver \
}; \
DECLARE_MODULE(name, name ## _mod, SI_SUB_DRIVERS, SI_ORDER_ANY); \
MODULE_DEPEND(name, cam, 1, 1, 1)
```

Definition at line 46 of file cam_periph.h.

7.7.1.10 #define ppriv_field0 periph_priv.entries[0].field

Definition at line 90 of file cam_periph.h.

7.7.1.11 #define ppriv_field1 periph_priv.entries[1].field

Definition at line 91 of file cam_periph.h.

7.7.1.12 #define ppriv_ptr0 periph_priv.entries[0].ptr

Definition at line 88 of file cam_periph.h.

7.7.1.13 #define ppriv_ptr1 periph_priv.entries[1].ptr

Definition at line 89 of file cam_periph.h.

7.7.2 Typedef Documentation**7.7.2.1 typedef cam_status periph_ctor_t(struct cam_periph *periph, void *arg)**

Definition at line 95 of file cam_periph.h.

7.7.2.2 typedef void periph_dtor_t(struct cam_periph *periph)

Definition at line 98 of file cam_periph.h.

7.7.2.3 typedef `periph_init_t`* `periph_init_func_t`

Definition at line 74 of file `cam_periph.h`.

7.7.2.4 typedef void(`periph_init_t`)(void)

Definition at line 69 of file `cam_periph.h`.

7.7.2.5 typedef void `periph_oninv_t`(struct `cam_periph` *`periph`)

Definition at line 97 of file `cam_periph.h`.

7.7.2.6 typedef void `periph_start_t`(struct `cam_periph` *`periph`, union `ccb` *`start_ccb`)

Definition at line 93 of file `cam_periph.h`.

7.7.3 Enumeration Type Documentation

7.7.3.1 enum `cam_periph_type`

Enumerator:

`CAM_PERIPH_BIO`

Definition at line 83 of file `cam_periph.h`.

7.7.4 Function Documentation

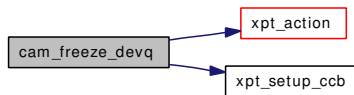
7.7.4.1 void `cam_freeze_devq` (struct `cam_path` *`path`)

Definition at line 905 of file `cam_periph.c`.

References `CAM_DEV_QFREEZE`, `ccb_hdr::flags`, `ccb_hdr::func_code`, `ccb_hdr::path`, `xpt_action()`, `XPT_NOOP`, and `xpt_setup_ccb()`.

Referenced by `cam_periph_freeze_after_event()`.

Here is the call graph for this function:



7.7.4.2 `cam_status` `cam_periph_acquire` (struct `cam_periph` *`periph`)

Definition at line 277 of file `cam_periph.c`.

References `CAM_REQ_CMP`, `CAM_REQ_CMP_ERR`, and `cam_periph::refcount`.

Referenced by `cam_periph_lock()`, `cdopen()`, `chopen()`, `passopen()`, `proberegister()`, `ptopen()`, `saopen()`, `sesopen()`, and `SLIST_HEAD()`.

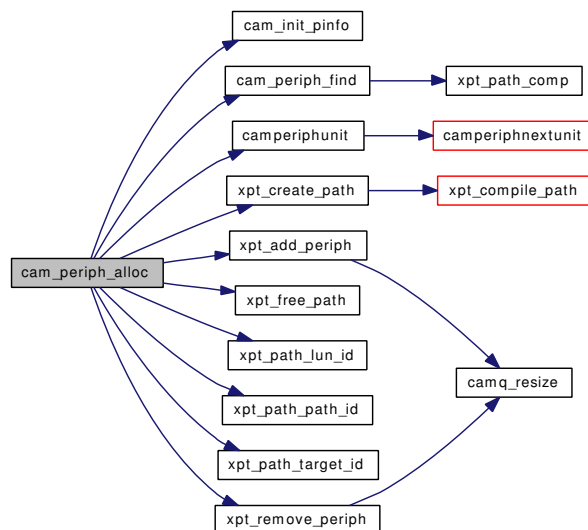
7.7.4.3 `cam_status` `cam_periph_alloc` (`periph_ctor_t * periph_ctor`, `periph_oninv_t * periph_oninvalidate`, `periph_dtor_t * periph_dtor`, `periph_start_t * periph_start`, `char * name`, `cam_periph_type` `type`, `struct cam_path *`, `ac_callback_t *`, `ac_code`, `void * arg`)

Definition at line 118 of file `cam_periph.c`.

References `cam_init_pinfo()`, `cam_periph_find()`, `CAM_PERIPH_INVALID`, `CAM_PERIPH_NEW_DEV_FOUND`, `CAM_PRIORITY_NONE`, `CAM_REQ_CMP`, `CAM_REQ_INPROG`, `CAM_REQ_INVALID`, `CAM_RESRC_UNAVAIL`, `camperiphunit()`, `cam_periph::deferred_ac`, `cam_periph::deferred_callback`, `cam_periph::flags`, `cam_periph::path`, `periph_drivers`, `cam_periph::periph_name`, `cam_periph::unit_number`, `xpt_add_periph()`, `xpt_create_path()`, `xpt_free_path()`, `xpt_path_lun_id()`, `xpt_path_path_id()`, `xpt_path_target_id()`, and `xpt_remove_periph()`.

Referenced by `cdasync()`, `chasync()`, `daasync()`, `passasync()`, `ptasync()`, `saasync()`, `sesasync()`, `targbhasync()`, `targenable()`, `xpt_init()`, and `xpt_scan_lun()`.

Here is the call graph for this function:



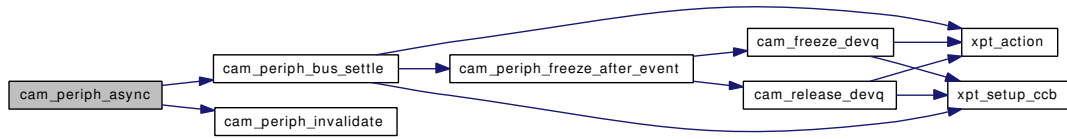
7.7.4.4 `void` `cam_periph_async` (`struct cam_periph * periph`, `u_int32_t code`, `struct cam_path * path`, `void * arg`)

Definition at line 1154 of file `cam_periph.c`.

References `AC_BUS_RESET`, `AC_LOST_DEVICE`, `AC_SENT_BDR`, `cam_periph_bus_settle()`, `cam_periph_invalidate()`, and `scsi_delay`.

Referenced by `cdasync()`, `chasync()`, `daasync()`, `passasync()`, `ptasync()`, `saasync()`, and `sesasync()`.

Here is the call graph for this function:



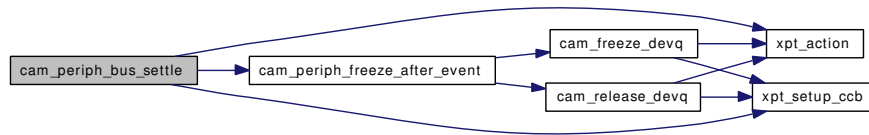
7.7.4.5 void cam_periph_bus_settle (struct cam_periph * periph, u_int bus_settle_ms)

Definition at line 1173 of file cam_periph.c.

References cam_periph_freeze_after_event(), ccb_getdevstats::ccb_h, ccb_hdr::func_code, cam_periph::path, xpt_action(), XPT_GDEV_STATS, and xpt_setup_ccb().

Referenced by cam_periph_async().

Here is the call graph for this function:



7.7.4.6 void cam_periph_ccbwait (union ccb * ccb)

Definition at line 778 of file cam_periph.c.

References CAM_REQ_INPROG, CAM_STATUS_MASK, CAM_UNQUEUED_INDEX, ccb_hdr::cbfcnp, ccb::ccb_h, cam_pinfo::index, ccb_hdr::pinfo, and ccb_hdr::status.

Referenced by cam_periph_runccb().

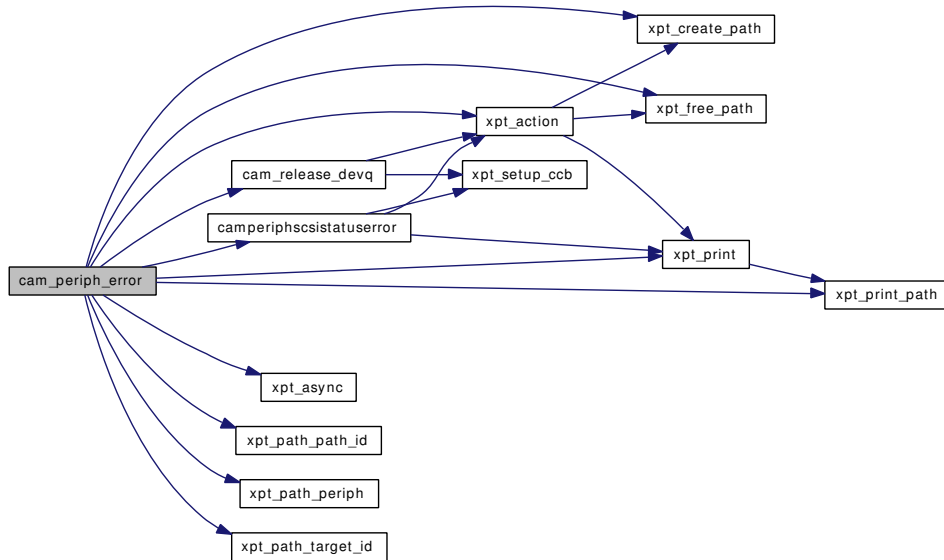
7.7.4.7 int cam_periph_error (union ccb * ccb, cam_flags camflags, u_int32_t sense_flags, union ccb * save_ccb)

Definition at line 1543 of file cam_periph.c.

References AC_LOST_DEVICE, CAM_AUTOSENSE_FAIL, CAM_BDR_SENT, CAM_BUSY, CAM_CMD_TIMEOUT, CAM_DATA_RUN_ERR, CAM_DEV_NOT_THERE, CAM_DEV_QFRZN, CAM_LUN_INVALID, CAM_LUN_WILDCARD, CAM_MSG_REJECT_REC, CAM_NO_HBA, CAM_PATH_INVALID, CAM_PROVIDE_FAIL, cam_release_devq(), CAM_REQ_CMP, CAM_REQ_CMP_ERR, CAM_REQ_INVALID, CAM_REQ_TOO_BIG, CAM_QUEUE_REQ, CAM_RESRC_UNAVAIL, CAM_RETRY_SELTO, CAM SCSI_BUS_RESET, CAM SCSI_STATUS_ERROR, CAM_SEL_TIMEOUT, CAM_STATUS_MASK, CAM_TID_INVALID, CAM_UA_ABORT, CAM_UA_TERMIO, CAM_UNCOR_PARITY, CAM_UNEXP_BUSFREE, camperiphscsistatuserror(), ccb::ccb_h, ccb_hdr::path, periph_busy_delay, periph_noresrc_delay, periph_selto_delay, RELSIM_RELEASE_AFTER_TIMEOUT, ccb_hdr::retry_count, ccb_hdr::status, xpt_action(), xpt_async(), xpt_create_path(), xpt_free_path(), xpt_path_path_id(), xpt_path_periph(), xpt_path_target_id(), xpt_print(), and xpt_print_path().

Referenced by `cderror()`, `cherror()`, `daerror()`, `passerror()`, `probedone()`, `pterror()`, `saerror()`, and `seserror()`.

Here is the call graph for this function:



7.7.4.8 struct `cam_periph`* `cam_periph_find` (struct `cam_path`* `path`, char* `name`)

Definition at line 251 of file `cam_periph.c`.

References `cam_periph::path`, `periph_drivers`, and `xpt_path_comp()`.

Referenced by `cam_periph_alloc()`, `cdregister()`, `targbhasync()`, `targenable()`, and `xpt_scan_lun()`.

Here is the call graph for this function:



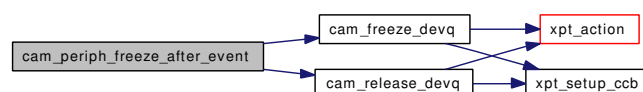
7.7.4.9 void `cam_periph_freeze_after_event` (struct `cam_periph`* `periph`, struct `timeval`* `event_time`, u_int `duration_ms`)

Definition at line 1184 of file `cam_periph.c`.

References `cam_freeze_devq()`, `cam_release_devq()`, `cam_periph::path`, and `RELSIM_RELEASE_AFTER_TIMEOUT`.

Referenced by `cam_periph_bus_settle()`, and `proberegister()`.

Here is the call graph for this function:



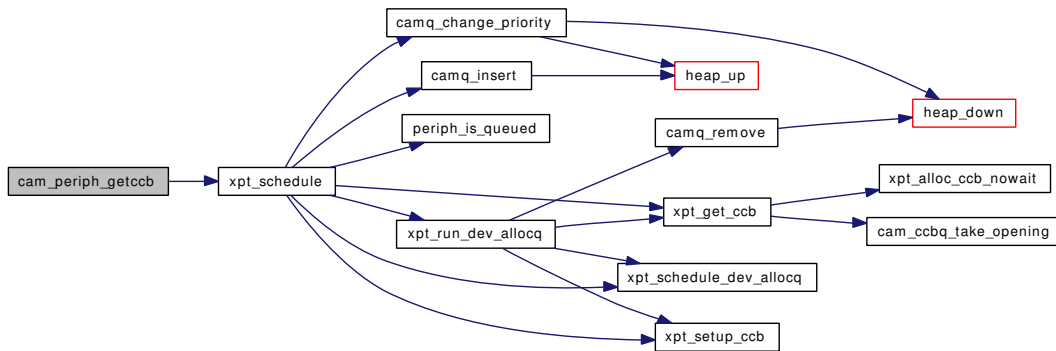
7.7.4.10 union **ccb*** cam_periph_getccb (struct **cam_periph** * *periph*, u_int32_t *priority*)

Definition at line 752 of file cam_periph.c.

References CAM_DEBUG, CAM_DEBUG_TRACE, cam_periph::immediate_priority, cam_periph::path, ccb_hdr::periph_links, and xpt_schedule().

Referenced by cam_periph_ioctl(), cdgetccb(), chexchange(), chgetelemstatus(), chgetparams(), chielem(), chmove(), chposition(), daclose(), dagetcapacity(), daprevent(), passioctl(), saerase(), sagetparams(), sa-loadunload(), samount(), saprevent(), sardpos(), sareserveleaseunit(), saretension(), sarewind(), saset-params(), sasetpos(), saspace(), sawritefilemarks(), and ses_runcmd().

Here is the call graph for this function:



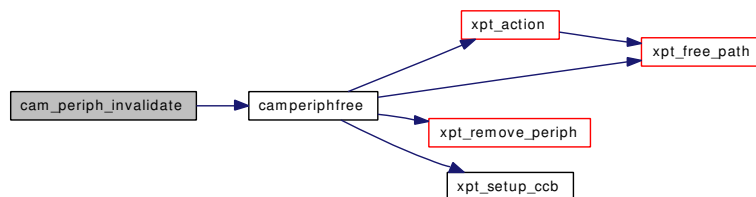
7.7.4.11 void cam_periph_invalidate (struct **cam_periph** * *periph*)

Definition at line 425 of file cam_periph.c.

References CAM_PERIPH_INVALID, CAM_PERIPH_NEW_DEV_FOUND, camperiphfree(), cam_periph::flags, and cam_periph::periph_oninval.

Referenced by cam_periph_async(), cddone(), chdone(), dadone(), probedone(), targbhasync(), targclose(), and targenable().

Here is the call graph for this function:



7.7.4.12 int cam_periph_ioctl (struct **cam_periph** * *periph*, int *cmd*, caddr_t *addr*, int(*) (union **ccb** ***ccb**, **cam_flags** camflags, u_int32_t *sense_flags*) *error_routine*)

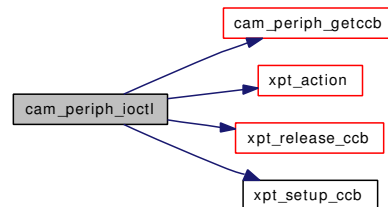
Definition at line 791 of file cam_periph.c.

References CAM_GDEVLIST_LAST_DEVICE, CAM_GDEVLIST_MORE_DEVS, cam_periph_getccb(), CAMGETPASSTHRU, ccb::ccb_h, ccb::cgdl, ccb_hdr::func_code, ccb_getdevlist::index,

ccb_hdr::path, ccb_getdevlist::periph_name, ccb_getdevlist::status, ccb_getdevlist::unit_number, xpt_action(), XPT_GDEVLIST, xpt_release_ccb(), and xpt_setup_ccb().

Referenced by cdiocctl(), chioctl(), passioctl(), ptioctl(), saioctl(), and sesioctl().

Here is the call graph for this function:



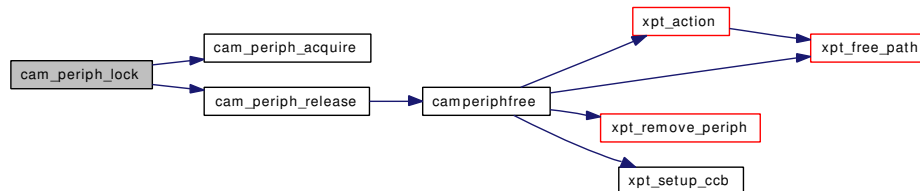
7.7.4.13 int cam_periph_lock (struct [cam_periph](#) * *periph*, int *priority*)

Definition at line 506 of file cam_periph.c.

References cam_periph_acquire(), CAM_PERIPH_LOCK_WANTED, CAM_PERIPH_LOCKED, cam_periph_release(), CAM_REQ_CMP, and cam_periph::flags.

Referenced by cdclose(), cdiocctl(), cdopen(), cdregister(), chclose(), chopen(), chregister(), daclose(), daregister(), passclose(), passopen(), ptclose(), ptioctl(), ptopen(), saclose(), saioctl(), saopen(), sesclose(), sesopen(), and SLIST_HEAD().

Here is the call graph for this function:



7.7.4.14 int cam_periph_mapmem (union [ccb](#) * *ccb*, struct [cam_periph_map_info](#) * *mapinfo*)

Definition at line 554 of file cam_periph.c.

References cam_periph_map_info::bp, CAM_DIR_IN, CAM_DIR_MASK, CAM_DIR_NONE, CAM_DIR_OUT, CAM_PERIPH_MAXMAPS, ccb::ccb_h, ccb::cdm, ccb::csio, ccb_scsiio::data_ptr, ccb_scsiio::dxfer_len, ccb_hdr::flags, ccb_hdr::func_code, ccb_dev_match::match_buf_len, ccb_dev_match::matches, cam_periph_map_info::num_bufs_used, ccb_dev_match::pattern_buf_len, ccb_dev_match::patterns, XPT_CONT_TARGET_IO, XPT_DEV_MATCH, and XPT SCSI_IO.

Referenced by passsendccb(), targsendccb(), and xptioctl().

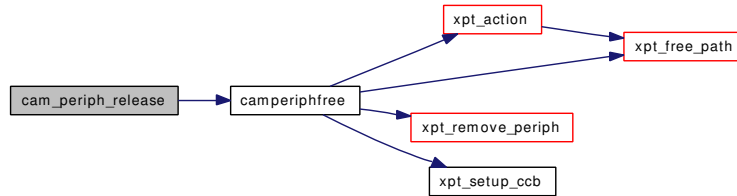
7.7.4.15 void cam_periph_release (struct [cam_periph](#) * *periph*)

Definition at line 292 of file cam_periph.c.

References CAM_PERIPH_INVALID, camperiphfree(), cam_periph::flags, and cam_periph::refcount.

Referenced by `cam_periph_lock()`, `cam_periph_unlock()`, `cdcclose()`, `chclose()`, `chopen()`, `daclose()`, `passclose()`, `probedone()`, `ptclose()`, `saclose()`, `saopen()`, `sesclose()`, `sesopen()`, and `SLIST_HEAD()`.

Here is the call graph for this function:



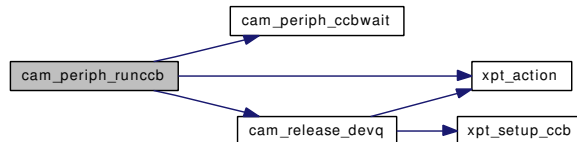
7.7.4.16 `int cam_periph_runccb (union ccb * ccb, int(*)(union ccb * ccb, cam_flags camflags, u_int32_t sense_flags) error_routine, cam_flags camflags, u_int32_t sense_flags, struct devstat * ds)`

Definition at line 853 of file `cam_periph.c`.

References `CAM_DEV_QFRZN`, `CAM_DIR_MASK`, `CAM_DIR_NONE`, `CAM_DIR_OUT`, `cam_periph_ccbwait()`, `cam_release_devq()`, `CAM_REQ_CMP`, `CAM_STATUS_MASK`, `xpt_action()`, and `XPT_SCSI_IO`.

Referenced by `cdrunccb()`, `chexchange()`, `chgetelemstatus()`, `chgetparams()`, `chielem()`, `chmove()`, `chposition()`, `daclose()`, `dagetcapacity()`, `daprevent()`, `passsendccb()`, `saerase()`, `sagetparams()`, `saloadunload()`, `samount()`, `saprevent()`, `sardpos()`, `sareservereleaseunit()`, `saretension()`, `sarewind()`, `sasetparams()`, `sasetpos()`, `saspace()`, `sawritefilemarks()`, `ses_runcmd()`, `TAILQ_HEAD()`, and `xptioctl()`.

Here is the call graph for this function:



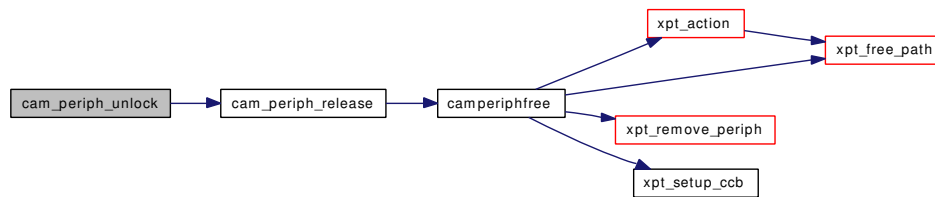
7.7.4.17 `void cam_periph_unlock (struct cam_periph * periph)`

Definition at line 535 of file `cam_periph.c`.

References `CAM_PERIPH_LOCK_WANTED`, `CAM_PERIPH_LOCKED`, `cam_periph_release()`, and `cam_periph::flags`.

Referenced by `cdcclose()`, `cddone()`, `cdioctl()`, `cdopen()`, `chclose()`, `chdone()`, `chopen()`, `daclose()`, `dadone()`, `passclose()`, `passopen()`, `ptclose()`, `ptioctl()`, `ptopen()`, `saclose()`, `saioctl()`, `saopen()`, `sesclose()`, `sesopen()`, and `SLIST_HEAD()`.

Here is the call graph for this function:



7.7.4.18 void cam_periph_unmapmem (union [ccb](#) * *ccb*, struct [cam_periph_map_info](#) * *mapinfo*)

Definition at line 702 of file `cam_periph.c`.

References `cam_periph_map_info::bp`, `CAM_PERIPH_MAXMAPS`, `ccb::ccb_h`, `ccb::cdm`, `ccb::csio`, `ccb_scsiio::data_ptr`, `ccb_hdr::func_code`, `ccb_dev_match::matches`, `cam_periph_map_info::num_bufs_` `used`, `ccb_dev_match::patterns`, `XPT_CONT_TARGET_IO`, `XPT_DEV_MATCH`, and `XPT SCSI_IO`.

Referenced by `passendccb()`, `targetreturnccb()`, and `xptioctl()`.

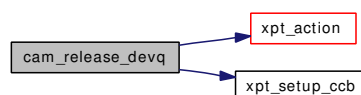
7.7.4.19 u_int32_t cam_release_devq (struct [cam_path](#) * *path*, u_int32_t *relsim_flags*, u_int32_t *opening_reduction*, u_int32_t *timeout*, int *getcount_only*)

Definition at line 916 of file `cam_periph.c`.

References `CAM_DEV_QFREEZE`, `ccb_relsim::ccb_h`, `ccb_hdr::flags`, `ccb_hdr::func_code`, `ccb_relsim::openings`, `ccb_relsim::release_flags`, `ccb_relsim::release_timeout`, `xpt_action()`, `XPT_REL_SIMQ`, and `xpt_setup_ccb()`.

Referenced by `cam_periph_error()`, `cam_periph_freeze_after_event()`, `cam_periph_runccb()`, `camperiphdone()`, `cd6byteworkaround()`, `cddone()`, `chdone()`, `cmd6workaround()`, `daclose()`, `dadone()`, `dagetcapacity()`, `dashutdown()`, `ptdone()`, `sadone()`, `saerase()`, `sardpos()`, `saretension()`, `sarewind()`, `sasetpos()`, `saspace()`, `sawritefilemarks()`, `ses_runcmd()`, `targbhdone()`, and `targbhstart()`.

Here is the call graph for this function:



7.7.4.20 void periphdriver_register (void *)

Definition at line 98 of file `cam_periph.c`.

References `nperiph_drivers`, and `periph_drivers`.

7.7.5 Variable Documentation

7.7.5.1 struct [periph_driver](#)** [periph_drivers](#)

Definition at line 85 of file `cam_periph.c`.

Referenced by `cam_periph_alloc()`, `cam_periph_find()`, `camperiphfree()`, `periphdriver_register()`, `xpt_finishconfig()`, `xptioctl()`, `xptdrvtraverse()`, and `xptlistperiphfunc()`.

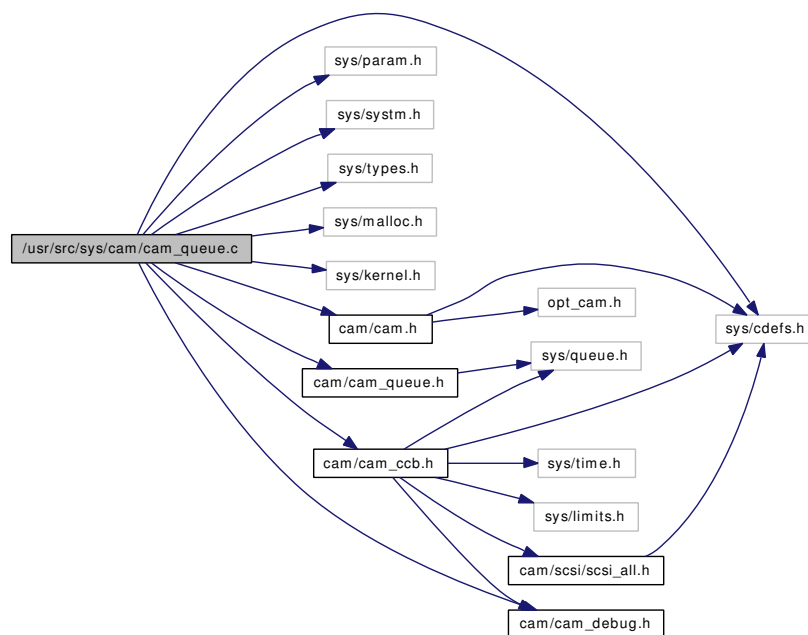
7.7.5.2 struct `cam_periph*` `xpt_periph`

Referenced by `xpt_action()`, `xpt_async()`, `xpt_config()`, `xpt_init()`, `xpt_scan_bus()`, `xpt_scan_lun()`, `xptconfigfunc()`, `xptioctl()`, and `xptregister()`.

7.8 /usr/src/sys/cam/cam_queue.c File Reference

```
#include <sys/cdefs.h>
#include <sys/param.h>
#include <sys/system.h>
#include <sys/types.h>
#include <sys/malloc.h>
#include <sys/kernel.h>
#include <cam/cam.h>
#include <cam/cam_ccb.h>
#include <cam/cam_queue.h>
#include <cam/cam_debug.h>
```

Include dependency graph for cam_queue.c:



Functions

- `__FBSDID` ("\$FreeBSD: src/sys/cam/cam_queue.c,v 1.9 2005/07/01 15:21:29 avatar Exp \$")
- `MALLOC_DEFINE` (M_CAMQ,"CAM queue","CAM queue buffers")
- `MALLOC_DEFINE` (M_CAMDEVQ,"CAM dev queue","CAM dev queue buffers")
- `MALLOC_DEFINE` (M_CAMCCBQ,"CAM ccb queue","CAM ccb queue buffers")
- static `__inline` int `queue_cmp` (cam_pinfo **queue_array, int i, int j)
- static `__inline` void `swap` (cam_pinfo **queue_array, int i, int j)
- static void `heap_up` (cam_pinfo **queue_array, int new_index)
- static void `heap_down` (cam_pinfo **queue_array, int index, int last_index)
- `camq` * `camq_alloc` (int size)

- int `camq_init` (struct `camq` *`camq`, int `size`)
- void `camq_free` (struct `camq` *`queue`)
- void `camq_fini` (struct `camq` *`queue`)
- u_int32_t `camq_resize` (struct `camq` *`queue`, int `new_size`)
- void `camq_insert` (struct `camq` *`queue`, `cam_pinfo` *`new_entry`)
- `cam_pinfo` * `camq_remove` (struct `camq` *`queue`, int `index`)
- void `camq_change_priority` (struct `camq` *`queue`, int `index`, u_int32_t `new_priority`)
- `cam_devq` * `cam_devq_alloc` (int `devices`, int `openings`)
- int `cam_devq_init` (struct `cam_devq` *`devq`, int `devices`, int `openings`)
- void `cam_devq_free` (struct `cam_devq` *`devq`)
- u_int32_t `cam_devq_resize` (struct `cam_devq` *`camq`, int `devices`)
- `cam_ccbq` * `cam_ccbq_alloc` (int `openings`)
- void `cam_ccbq_free` (struct `cam_ccbq` *`ccbq`)
- u_int32_t `cam_ccbq_resize` (struct `cam_ccbq` *`ccbq`, int `new_size`)
- int `cam_ccbq_init` (struct `cam_ccbq` *`ccbq`, int `openings`)

7.8.1 Function Documentation

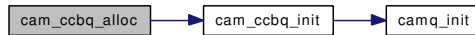
7.8.1.1 `__FBSDID` ("FreeBSD: src/sys/cam/cam_queue.c, v 1.9 2005/07/01 15:21:29 avatar Exp \$")

7.8.1.2 struct `cam_ccbq`* `cam_ccbq_alloc` (int *openings*)

Definition at line 271 of file `cam_queue.c`.

References `cam_ccbq_init()`.

Here is the call graph for this function:

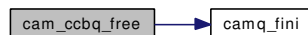


7.8.1.3 void `cam_ccbq_free` (struct `cam_ccbq` * *ccbq*)

Definition at line 289 of file `cam_queue.c`.

References `camq_fini()`, and `cam_ccbq::queue`.

Here is the call graph for this function:



7.8.1.4 int `cam_ccbq_init` (struct `cam_ccbq` * *ccbq*, int *openings*)

Definition at line 329 of file `cam_queue.c`.

References `camq_init()`.

Referenced by `cam_ccbq_alloc()`, and `xpt_alloc_device()`.

Here is the call graph for this function:



7.8.1.5 `u_int32_t cam_ccbq_resize (struct cam_ccbq * ccbq, int new_size)`

Definition at line 298 of file `cam_queue.c`.

References `CAM_REQ_CMP`, `CAM_RESRC_UNAVAIL`, `camq_resize()`, `cam_ccbq::dev_active`, `cam_ccbq::dev_openings`, `cam_ccbq::devq_openings`, `camq::entries`, `cam_ccbq::held`, and `cam_ccbq::queue`.

Referenced by `xpt_dev_ccbq_resize()`, and `xpt_schedule_dev_allocq()`.

Here is the call graph for this function:



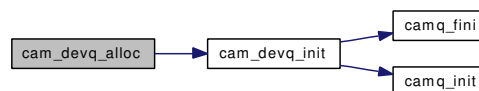
7.8.1.6 `struct cam_devq* cam_devq_alloc (int devices, int openings)`

Definition at line 214 of file `cam_queue.c`.

References `cam_devq_init()`.

Referenced by `cam_simq_alloc()`.

Here is the call graph for this function:



7.8.1.7 `void cam_devq_free (struct cam_devq * devq)`

Definition at line 250 of file `cam_queue.c`.

References `cam_devq::alloc_queue`, `camq_fini()`, and `cam_devq::send_queue`.

Referenced by `cam_simq_free()`.

Here is the call graph for this function:



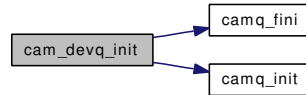
7.8.1.8 `int cam_devq_init (struct cam_devq * devq, int devices, int openings)`

Definition at line 232 of file `cam_queue.c`.

References `camq_fini()`, and `camq_init()`.

Referenced by `cam_devq_alloc()`.

Here is the call graph for this function:



7.8.1.9 `u_int32_t cam_devq_resize (struct cam_devq * camq, int devices)`

Definition at line 258 of file `cam_queue.c`.

References `cam_devq::alloc_queue`, `CAM_REQ_CMP`, `camq_resize()`, and `cam_devq::send_queue`.

Referenced by `xpt_alloc_device()`, and `xpt_release_device()`.

Here is the call graph for this function:



7.8.1.10 `struct camq* camq_alloc (int size)`

Definition at line 56 of file `cam_queue.c`.

References `camq_init()`.

Here is the call graph for this function:



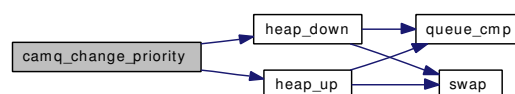
7.8.1.11 `void camq_change_priority (struct camq * queue, int index, u_int32_t new_priority)`

Definition at line 201 of file `cam_queue.c`.

References `camq::entries`, `heap_down()`, `heap_up()`, `cam_pinfo::priority`, and `camq::queue_array`.

Referenced by `xpt_schedule()`, and `xpt_schedule_dev()`.

Here is the call graph for this function:



7.8.1.12 void camq_fini (struct camq * queue)

Definition at line 107 of file cam_queue.c.

References camq::queue_array.

Referenced by cam_ccbq_free(), cam_devq_free(), cam_devq_init(), camq_free(), xpt_alloc_device(), and xpt_release_device().

7.8.1.13 void camq_free (struct camq * queue)

Definition at line 98 of file cam_queue.c.

References camq_fini().

Here is the call graph for this function:

**7.8.1.14 int camq_init (struct camq * camq, int size)**

Definition at line 71 of file cam_queue.c.

Referenced by cam_ccbq_init(), cam_devq_init(), camq_alloc(), cdregister(), and xpt_alloc_device().

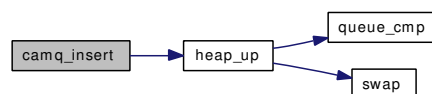
7.8.1.15 void camq_insert (struct camq * queue, cam_pinfo * new_entry)

Definition at line 157 of file cam_queue.c.

References camq::array_size, camq::entries, heap_up(), cam_pinfo::index, and camq::queue_array.

Referenced by cam_ccbq_insert_ccb(), cdgetccb(), cdrunchangerqueue(), cdschedule(), xpt_schedule(), and xpt_schedule_dev().

Here is the call graph for this function:

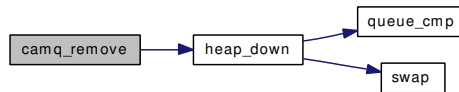
**7.8.1.16 cam_pinfo* camq_remove (struct camq * queue, int index)**

Definition at line 177 of file cam_queue.c.

References CAM_UNQUEUED_INDEX, camq::entries, heap_down(), cam_pinfo::index, and camq::queue_array.

Referenced by cam_ccbq_remove_ccb(), cdoninvalidate(), cdrunchangerqueue(), xpt_bus_deregister(), xpt_run_dev_allocq(), and xpt_run_dev_sendq().

Here is the call graph for this function:



7.8.1.17 `u_int32_t camq_resize (struct camq * queue, int new_size)`

Definition at line 120 of file `cam_queue.c`.

References `camq::array_size`, `CAM_REQ_CMP`, `CAM_RESRC_UNAVAIL`, `camq::entries`, and `camq::queue_array`.

Referenced by `cam_ccbq_resize()`, `cam_devq_resize()`, `cdregister()`, `xpt_add_periph()`, and `xpt_remove_periph()`.

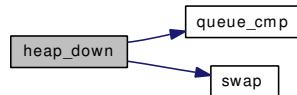
7.8.1.18 `static void heap_down (cam_pinfo ** queue_array, int index, int last_index)` [static]

Definition at line 405 of file `cam_queue.c`.

References `queue_cmp()`, and `swap()`.

Referenced by `camq_change_priority()`, and `camq_remove()`.

Here is the call graph for this function:



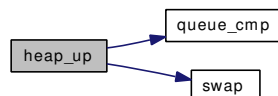
7.8.1.19 `static void heap_up (cam_pinfo ** queue_array, int new_index)` [static]

Definition at line 382 of file `cam_queue.c`.

References `queue_cmp()`, and `swap()`.

Referenced by `camq_change_priority()`, and `camq_insert()`.

Here is the call graph for this function:



7.8.1.20 `MALLOC_DEFINE (M_CAMCCBQ, "CAM ccb queue", "CAM ccb queue buffers")`

7.8.1.21 `MALLOC_DEFINE (M_CAMDEVQ, "CAM dev queue", "CAM dev queue buffers")`

7.8.1.22 `MALLOC_DEFINE (M_CAMQ, "CAM queue", "CAM queue buffers")`

7.8.1.23 `static __inline int queue_cmp (cam_pinfo ** queue_array, int i, int j) [static]`

Definition at line 350 of file cam_queue.c.

Referenced by heap_down(), and heap_up().

7.8.1.24 `static __inline void swap (cam_pinfo ** queue_array, int i, int j) [static]`

Definition at line 365 of file cam_queue.c.

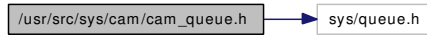
References cam_pinfo::index.

Referenced by heap_down(), and heap_up().

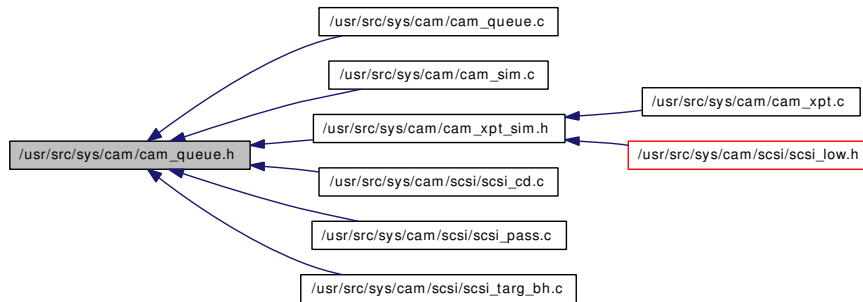
7.9 /usr/src/sys/cam/cam_queue.h File Reference

```
#include <sys/queue.h>
```

Include dependency graph for cam_queue.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- struct [camq](#)
- struct [cam_ccbq](#)
- struct [cam_devq](#)

Defines

- #define [_CAM_CAM_QUEUE_H](#) 1
- #define [CAMQ_HEAD](#) 1
- #define [CAMQ_GET_HEAD](#)(camq) ((camq) → queue_array[CAMQ_HEAD])

Functions

- [TAILQ_HEAD](#) (ccb_hdr_tailq, ccb_hdr)
- [LIST_HEAD](#) (ccb_hdr_list, ccb_hdr)
- [SLIST_HEAD](#) (ccb_hdr_slist, ccb_hdr)
- [cam_devq](#) * [cam_devq_alloc](#) (int devices, int openings)
- int [cam_devq_init](#) (struct [cam_devq](#) *devq, int devices, int openings)
- void [cam_devq_free](#) (struct [cam_devq](#) *devq)
- u_int32_t [cam_devq_resize](#) (struct [cam_devq](#) *camq, int openings)
- [cam_ccbq](#) * [cam_ccbq_alloc](#) (int openings)
- u_int32_t [cam_ccbq_resize](#) (struct [cam_ccbq](#) *ccbq, int devices)
- int [cam_ccbq_init](#) (struct [cam_ccbq](#) *ccbq, int openings)
- void [cam_ccbq_free](#) (struct [cam_ccbq](#) *ccbq)
- void [cam_ccbq_fini](#) (struct [cam_ccbq](#) *ccbq)
- [camq](#) * [camq_alloc](#) (int size)

- `u_int32_t camq_resize` (struct `camq` *queue, int new_size)
- `int camq_init` (struct `camq` *camq, int size)
- `void camq_free` (struct `camq` *queue)
- `void camq_fini` (struct `camq` *queue)
- `void camq_insert` (struct `camq` *queue, `cam_pinfo` *new_entry)
- `cam_pinfo` * `camq_remove` (struct `camq` *queue, int index)
- `void camq_change_priority` (struct `camq` *queue, int index, `u_int32_t` new_priority)
- `static __inline int cam_ccbq_pending_ccb_count` (struct `cam_ccbq` *ccbq)
- `static __inline void cam_ccbq_take_opening` (struct `cam_ccbq` *ccbq)
- `static __inline void cam_ccbq_insert_ccb` (struct `cam_ccbq` *ccbq, union `ccb` *new_ccb)
- `static __inline void cam_ccbq_remove_ccb` (struct `cam_ccbq` *ccbq, union `ccb` *ccb)
- `static __inline union ccb` * `cam_ccbq_peek_ccb` (struct `cam_ccbq` *ccbq, int index)
- `static __inline void cam_ccbq_send_ccb` (struct `cam_ccbq` *queue, union `ccb` *send_ccb)
- `static __inline void cam_ccbq_ccb_done` (struct `cam_ccbq` *ccbq, union `ccb` *done_ccb)
- `static __inline void cam_ccbq_release_opening` (struct `cam_ccbq` *ccbq)

7.9.1 Define Documentation

7.9.1.1 #define _CAM_CAM_QUEUE_H 1

Definition at line 32 of file `cam_queue.h`.

7.9.1.2 #define CAMQ_GET_HEAD(camq) ((camq) → queue_array[CAMQ_HEAD])

Definition at line 142 of file `cam_queue.h`.

Referenced by `xpt_schedule_dev_allocq()`, and `xpt_schedule_dev_sendq()`.

7.9.1.3 #define CAMQ_HEAD 1

Definition at line 139 of file `cam_queue.h`.

Referenced by `cdrunchangerqueue()`, `xpt_bus_deregister()`, `xpt_run_dev_allocq()`, and `xpt_run_dev_sendq()`.

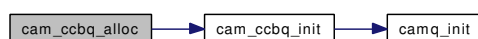
7.9.2 Function Documentation

7.9.2.1 struct `cam_ccbq`* `cam_ccbq_alloc` (int openings)

Definition at line 271 of file `cam_queue.c`.

References `cam_ccbq_init()`.

Here is the call graph for this function:



7.9.2.2 `static __inline void cam_ccbq_ccb_done (struct cam_ccbq * ccbq, union ccb * done_ccb)`
`[static]`

Definition at line 221 of file `cam_queue.h`.

References `cam_ccbq::active_ccbs`, `ccb::ccb_h`, `cam_ccbq::dev_active`, `cam_ccbq::dev_openings`, and `cam_ccbq::held`.

Referenced by `camisr()`.

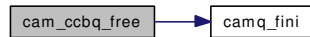
7.9.2.3 `void cam_ccbq_fini (struct cam_ccbq * ccbq)`

7.9.2.4 `void cam_ccbq_free (struct cam_ccbq * ccbq)`

Definition at line 289 of file `cam_queue.c`.

References `camq_fini()`, and `cam_ccbq::queue`.

Here is the call graph for this function:



7.9.2.5 `int cam_ccbq_init (struct cam_ccbq * ccbq, int openings)`

Definition at line 329 of file `cam_queue.c`.

References `camq_init()`.

Referenced by `cam_ccbq_alloc()`, and `xpt_alloc_device()`.

Here is the call graph for this function:



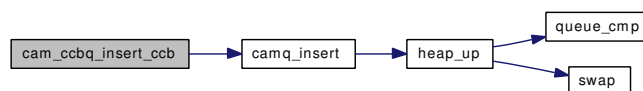
7.9.2.6 `static __inline void cam_ccbq_insert_ccb (struct cam_ccbq * ccbq, union ccb * new_ccb)`
`[static]`

Definition at line 190 of file `cam_queue.h`.

References `camq_insert()`, `ccb::ccb_h`, `cam_ccbq::held`, `ccb_hdr::pinfo`, and `cam_ccbq::queue`.

Referenced by `xpt_action()`.

Here is the call graph for this function:



7.9.2.7 `static __inline union ccb * cam_ccbq_peek_ccb (struct cam_ccbq * ccbq, int index)`
`[static]`

Definition at line 203 of file `cam_queue.h`.

References `cam_ccbq::queue`, and `camq::queue_array`.

Referenced by `xpt_bus_deregister()`, and `xpt_run_dev_sendq()`.

7.9.2.8 `static __inline int cam_ccbq_pending_ccb_count (struct cam_ccbq * ccbq)` `[static]`

Definition at line 177 of file `cam_queue.h`.

References `camq::entries`, and `cam_ccbq::queue`.

7.9.2.9 `static __inline void cam_ccbq_release_opening (struct cam_ccbq * ccbq)` `[static]`

Definition at line 231 of file `cam_queue.h`.

References `cam_ccbq::devq_openings`, and `cam_ccbq::held`.

Referenced by `xpt_release_ccb()`.

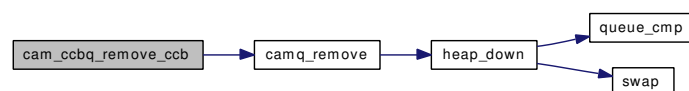
7.9.2.10 `static __inline void cam_ccbq_remove_ccb (struct cam_ccbq * ccbq, union ccb * ccb)`
`[static]`

Definition at line 197 of file `cam_queue.h`.

References `camq_remove()`, `ccb::ccb_h`, `cam_pinfo::index`, `ccb_hdr::pinfo`, and `cam_ccbq::queue`.

Referenced by `xpt_action()`, `xpt_bus_deregister()`, and `xpt_run_dev_sendq()`.

Here is the call graph for this function:



7.9.2.11 `u_int32_t cam_ccbq_resize (struct cam_ccbq * ccbq, int devices)`

Definition at line 298 of file `cam_queue.c`.

References `CAM_REQ_CMP`, `CAM_RESRC_UNAVAIL`, `camq_resize()`, `cam_ccbq::dev_active`, `cam_ccbq::dev_openings`, `cam_ccbq::devq_openings`, `camq::entries`, `cam_ccbq::held`, and `cam_ccbq::queue`.

Referenced by `xpt_dev_ccbq_resize()`, and `xpt_schedule_dev_allocq()`.

Here is the call graph for this function:



7.9.2.12 `static __inline void cam_ccbq_send_ccb (struct cam_ccbq * queue, union ccb * send_ccb)`
`[static]`

Definition at line 209 of file `cam_queue.h`.

References `cam_ccbq::active_ccbs`, `CAM_ACTIVE_INDEX`, `ccb::ccb_h`, `cam_ccbq::dev_active`, `cam_ccbq::dev_openings`, `cam_pinfo::index`, and `ccb_hdr::pinfo`.

Referenced by `xpt_action()`, `xpt_bus_deregister()`, and `xpt_run_dev_sendq()`.

7.9.2.13 `static __inline void cam_ccbq_take_opening (struct cam_ccbq * ccbq)` `[static]`

Definition at line 183 of file `cam_queue.h`.

References `cam_ccbq::devq_openings`, and `cam_ccbq::held`.

Referenced by `xpt_get_ccb()`.

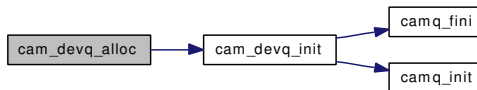
7.9.2.14 `struct cam_devq* cam_devq_alloc (int devices, int openings)`

Definition at line 214 of file `cam_queue.c`.

References `cam_devq_init()`.

Referenced by `cam_simq_alloc()`.

Here is the call graph for this function:



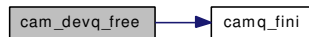
7.9.2.15 `void cam_devq_free (struct cam_devq * devq)`

Definition at line 250 of file `cam_queue.c`.

References `cam_devq::alloc_queue`, `camq_fini()`, and `cam_devq::send_queue`.

Referenced by `cam_simq_free()`.

Here is the call graph for this function:



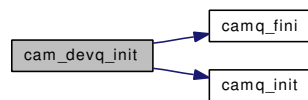
7.9.2.16 `int cam_devq_init (struct cam_devq * devq, int devices, int openings)`

Definition at line 232 of file `cam_queue.c`.

References `camq_fini()`, and `camq_init()`.

Referenced by `cam_devq_alloc()`.

Here is the call graph for this function:



7.9.2.17 `u_int32_t cam_devq_resize (struct cam_devq * camq, int openings)`

Definition at line 258 of file `cam_queue.c`.

References `cam_devq::alloc_queue`, `CAM_REQ_CMP`, `camq_resize()`, and `cam_devq::send_queue`.

Referenced by `xpt_alloc_device()`, and `xpt_release_device()`.

Here is the call graph for this function:



7.9.2.18 `struct camq* camq_alloc (int size)`

Definition at line 56 of file `cam_queue.c`.

References `camq_init()`.

Here is the call graph for this function:



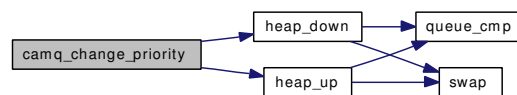
7.9.2.19 `void camq_change_priority (struct camq * queue, int index, u_int32_t new_priority)`

Definition at line 201 of file `cam_queue.c`.

References `camq::entries`, `heap_down()`, `heap_up()`, `cam_pinfo::priority`, and `camq::queue_array`.

Referenced by `xpt_schedule()`, and `xpt_schedule_dev()`.

Here is the call graph for this function:



7.9.2.20 `void camq_fini (struct camq * queue)`

Definition at line 107 of file `cam_queue.c`.

References `camq::queue_array`.

Referenced by `cam_ccbq_free()`, `cam_devq_free()`, `cam_devq_init()`, `camq_free()`, `xpt_alloc_device()`, and `xpt_release_device()`.

7.9.2.21 void camq_free (struct camq * queue)

Definition at line 98 of file cam_queue.c.

References camq_fini().

Here is the call graph for this function:



7.9.2.22 int camq_init (struct camq * camq, int size)

Definition at line 71 of file cam_queue.c.

Referenced by cam_ccbq_init(), cam_devq_init(), camq_alloc(), cdregister(), and xpt_alloc_device().

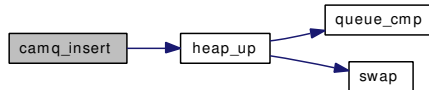
7.9.2.23 void camq_insert (struct camq * queue, cam_pinfo * new_entry)

Definition at line 157 of file cam_queue.c.

References camq::array_size, camq::entries, heap_up(), cam_pinfo::index, and camq::queue_array.

Referenced by cam_ccbq_insert_ccb(), cdgetccb(), cdrunchangerqueue(), cdschedule(), xpt_schedule(), and xpt_schedule_dev().

Here is the call graph for this function:



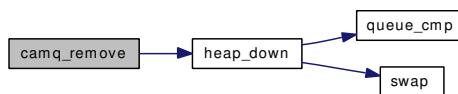
7.9.2.24 cam_pinfo* camq_remove (struct camq * queue, int index)

Definition at line 177 of file cam_queue.c.

References CAM_UNQUEUED_INDEX, camq::entries, heap_down(), cam_pinfo::index, and camq::queue_array.

Referenced by cam_ccbq_remove_ccb(), cdoninvalidate(), cdrunchangerqueue(), xpt_bus_deregister(), xpt_run_dev_allocq(), and xpt_run_dev_sendq().

Here is the call graph for this function:



7.9.2.25 `u_int32_t camq_resize (struct camq * queue, int new_size)`

Definition at line 120 of file `cam_queue.c`.

References `camq::array_size`, `CAM_REQ_CMP`, `CAM_RESRC_UNAVAIL`, `camq::entries`, and `camq::queue_array`.

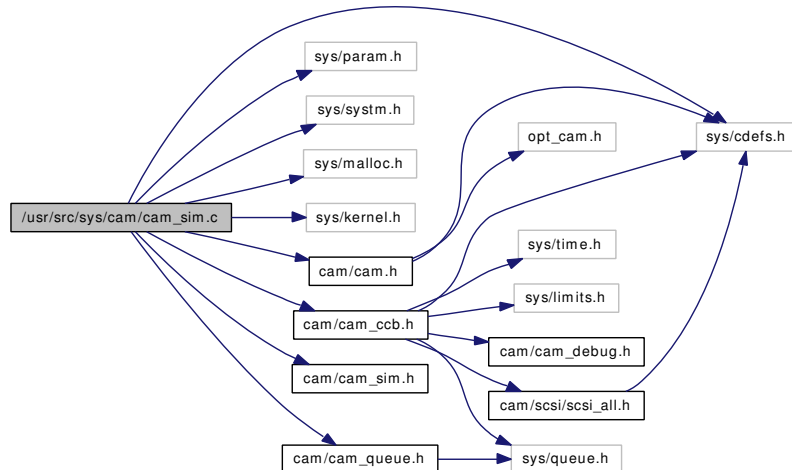
Referenced by `cam_ccbq_resize()`, `cam_devq_resize()`, `cdregister()`, `xpt_add_periph()`, and `xpt_remove_periph()`.

7.9.2.26 `LIST_HEAD (ccb_hdr_list, ccb_hdr)`**7.9.2.27** `SLIST_HEAD (ccb_hdr_slist, ccb_hdr)`**7.9.2.28** `TAILQ_HEAD (ccb_hdr_tailq, ccb_hdr)`

7.10 /usr/src/sys/cam/cam_sim.c File Reference

```
#include <sys/cdefs.h>
#include <sys/param.h>
#include <sys/system.h>
#include <sys/malloc.h>
#include <sys/kernel.h>
#include <cam/cam.h>
#include <cam/cam_ccb.h>
#include <cam/cam_sim.h>
#include <cam/cam_queue.h>
```

Include dependency graph for cam_sim.c:



Defines

- #define [CAM_PATH_ANY](#) (u_int32_t)-1

Functions

- [__FBSDID](#) ("\$FreeBSD: src/sys/cam/cam_sim.c,v 1.9 2005/07/01 15:21:29 avatar Exp \$")
- [MALLOC_DEFINE](#) (M_CAMSIM,"CAM SIM","CAM SIM buffers")
- [cam_devq * cam_simq_alloc](#) (u_int32_t max_sim_transactions)
- void [cam_simq_free](#) (struct [cam_devq](#) *devq)
- [cam_sim * cam_sim_alloc](#) ([sim_action_func](#) sim_action, [sim_poll_func](#) sim_poll, const char *sim_name, void *softc, u_int32_t unit, int max_dev_transactions, int max_tagged_dev_transactions, struct [cam_devq](#) *queue)
- void [cam_sim_free](#) (struct [cam_sim](#) *sim, int free_devq)
- void [cam_sim_set_path](#) (struct [cam_sim](#) *sim, u_int32_t path_id)

7.10.1 Define Documentation

7.10.1.1 #define CAM_PATH_ANY (u_int32_t)-1

Definition at line 42 of file cam_sim.c.

Referenced by cam_sim_alloc().

7.10.2 Function Documentation

7.10.2.1 __FBSDID ("\$FreeBSD: src/sys/cam/cam_sim. c, v 1.9 2005/07/01 15:21:29 avatar Exp \$")

7.10.2.2 struct **cam_sim*** cam_sim_alloc (sim_action_func sim_action, sim_poll_func sim_poll, const char * sim_name, void * softc, u_int32_t unit, int max_dev_transactions, int max_tagged_dev_transactions, struct **cam_devq** * queue)

Definition at line 59 of file cam_sim.c.

References cam_sim::bus_id, cam_sim::c_handle, CAM_PATH_ANY, cam_sim::devq, cam_sim::flags, cam_sim::max_dev_openings, cam_sim::max_tagged_dev_openings, cam_sim::path_id, cam_sim::sim_action, cam_sim::sim_name, cam_sim::sim_poll, cam_sim::softc, and cam_sim::unit_number.

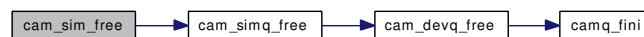
Referenced by xpt_init().

7.10.2.3 void cam_sim_free (struct **cam_sim** * sim, int free_devq)

Definition at line 98 of file cam_sim.c.

References cam_simq_free(), and cam_sim::devq.

Here is the call graph for this function:



7.10.2.4 void cam_sim_set_path (struct **cam_sim** * sim, u_int32_t path_id)

Definition at line 106 of file cam_sim.c.

References cam_sim::path_id.

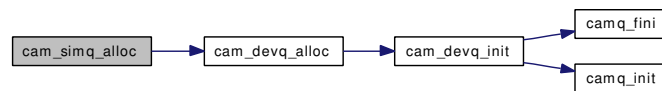
7.10.2.5 struct **cam_devq*** cam_simq_alloc (u_int32_t max_sim_transactions)

Definition at line 47 of file cam_sim.c.

References cam_devq_alloc().

Referenced by xpt_init().

Here is the call graph for this function:



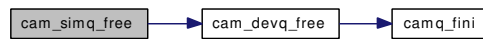
7.10.2.6 void `cam_simq_free` (struct `cam_devq` * `devq`)

Definition at line 53 of file `cam_sim.c`.

References `cam_devq_free`().

Referenced by `cam_sim_free`().

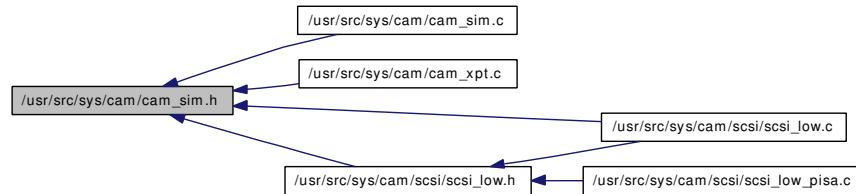
Here is the call graph for this function:



7.10.2.7 `MALLOC_DEFINE` (M_CAMSIM, "CAM SIM", "CAM SIM buffers")

7.11 /usr/src/sys/cam/cam_sim.h File Reference

This graph shows which files directly or indirectly include this file:



Data Structures

- struct [cam_sim](#)

Defines

- #define [_CAM_CAM_SIM_H](#) 1
- #define [spriv_ptr0](#) `sim_priv.entries[0].ptr`
- #define [spriv_ptr1](#) `sim_priv.entries[1].ptr`
- #define [spriv_field0](#) `sim_priv.entries[0].field`
- #define [spriv_field1](#) `sim_priv.entries[1].field`
- #define [CAM_SIM_REL_TIMEOUT_PENDING](#) 0x01

Typedefs

- typedef void(*) [sim_action_func](#) (struct [cam_sim](#) *sim, union [ccb](#) *ccb)
- typedef void(*) [sim_poll_func](#) (struct [cam_sim](#) *sim)

Functions

- [cam_devq * cam_simq_alloc](#) (u_int32_t max_sim_transactions)
- void [cam_simq_free](#) (struct [cam_devq](#) *devq)
- [cam_sim * cam_sim_alloc](#) ([sim_action_func](#) sim_action, [sim_poll_func](#) sim_poll, const char *sim_name, void *softc, u_int32_t unit, int max_dev_transactions, int max_tagged_dev_transactions, struct [cam_devq](#) *queue)
- void [cam_sim_free](#) (struct [cam_sim](#) *sim, int free_devq)
- void [cam_sim_set_path](#) (struct [cam_sim](#) *sim, u_int32_t path_id)
- static __inline u_int32_t [cam_sim_path](#) (struct [cam_sim](#) *sim)
- static __inline const char * [cam_sim_name](#) (struct [cam_sim](#) *sim)
- static __inline void * [cam_sim_softc](#) (struct [cam_sim](#) *sim)
- static __inline u_int32_t [cam_sim_unit](#) (struct [cam_sim](#) *sim)
- static __inline u_int32_t [cam_sim_bus](#) (struct [cam_sim](#) *sim)

7.11.1 Define Documentation

7.11.1.1 #define _CAM_CAM_SIM_H 1

Definition at line 32 of file [cam_sim.h](#).

7.11.1.2 #define CAM_SIM_REL_TIMEOUT_PENDING 0x01

Definition at line 99 of file cam_sim.h.

Referenced by xpt_release_simq().

7.11.1.3 #define spriv_field0 sim_priv.entries[0].field

Definition at line 81 of file cam_sim.h.

7.11.1.4 #define spriv_field1 sim_priv.entries[1].field

Definition at line 82 of file cam_sim.h.

7.11.1.5 #define spriv_ptr0 sim_priv.entries[0].ptr

Definition at line 79 of file cam_sim.h.

7.11.1.6 #define spriv_ptr1 sim_priv.entries[1].ptr

Definition at line 80 of file cam_sim.h.

7.11.2 Typedef Documentation**7.11.2.1 typedef void(*) [sim_action_func](#)(struct [cam_sim](#) *sim, union [ccb](#) *ccb)**

Definition at line 48 of file cam_sim.h.

7.11.2.2 typedef void(*) [sim_poll_func](#)(struct [cam_sim](#) *sim)

Definition at line 49 of file cam_sim.h.

7.11.3 Function Documentation**7.11.3.1 struct [cam_sim](#)* [cam_sim_alloc](#) ([sim_action_func](#) *sim_action*, [sim_poll_func](#) *sim_poll*, const char * *sim_name*, void * *softc*, u_int32_t *unit*, int *max_dev_transactions*, int *max_tagged_dev_transactions*, struct [cam_devq](#) * *queue*)**

Definition at line 59 of file cam_sim.c.

References [cam_sim::bus_id](#), [cam_sim::c_handle](#), [CAM_PATH_ANY](#), [cam_sim::devq](#), [cam_sim::flags](#), [cam_sim::max_dev_openings](#), [cam_sim::max_tagged_dev_openings](#), [cam_sim::path_id](#), [cam_sim::sim_action](#), [cam_sim::sim_name](#), [cam_sim::sim_poll](#), [cam_sim::softc](#), and [cam_sim::unit_number](#).

Referenced by xpt_init().

7.11.3.2 static __inline u_int32_t [cam_sim_bus](#) (struct [cam_sim](#) * *sim*) [static]

Definition at line 129 of file cam_sim.h.

References `cam_sim::bus_id`.

7.11.3.3 void `cam_sim_free` (struct `cam_sim` * *sim*, int *free_devq*)

Definition at line 98 of file `cam_sim.c`.

References `cam_simq_free()`, and `cam_sim::devq`.

Here is the call graph for this function:



7.11.3.4 static __inline const char * `cam_sim_name` (struct `cam_sim` * *sim*) [static]

Definition at line 111 of file `cam_sim.h`.

References `cam_sim::sim_name`.

7.11.3.5 static __inline u_int32_t `cam_sim_path` (struct `cam_sim` * *sim*) [static]

Definition at line 105 of file `cam_sim.h`.

References `cam_sim::path_id`.

7.11.3.6 void `cam_sim_set_path` (struct `cam_sim` * *sim*, u_int32_t *path_id*)

Definition at line 106 of file `cam_sim.c`.

References `cam_sim::path_id`.

7.11.3.7 static __inline void * `cam_sim_softc` (struct `cam_sim` * *sim*) [static]

Definition at line 117 of file `cam_sim.h`.

References `cam_sim::softc`.

7.11.3.8 static __inline u_int32_t `cam_sim_unit` (struct `cam_sim` * *sim*) [static]

Definition at line 123 of file `cam_sim.h`.

References `cam_sim::unit_number`.

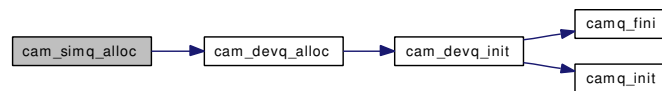
7.11.3.9 struct `cam_devq`* `cam_simq_alloc` (u_int32_t *max_sim_transactions*)

Definition at line 47 of file `cam_sim.c`.

References `cam_devq_alloc()`.

Referenced by `xpt_init()`.

Here is the call graph for this function:



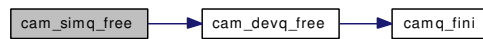
7.11.3.10 void `cam_simq_free` (struct `cam_devq` * `devq`)

Definition at line 53 of file `cam_sim.c`.

References `cam_devq_free()`.

Referenced by `cam_sim_free()`.

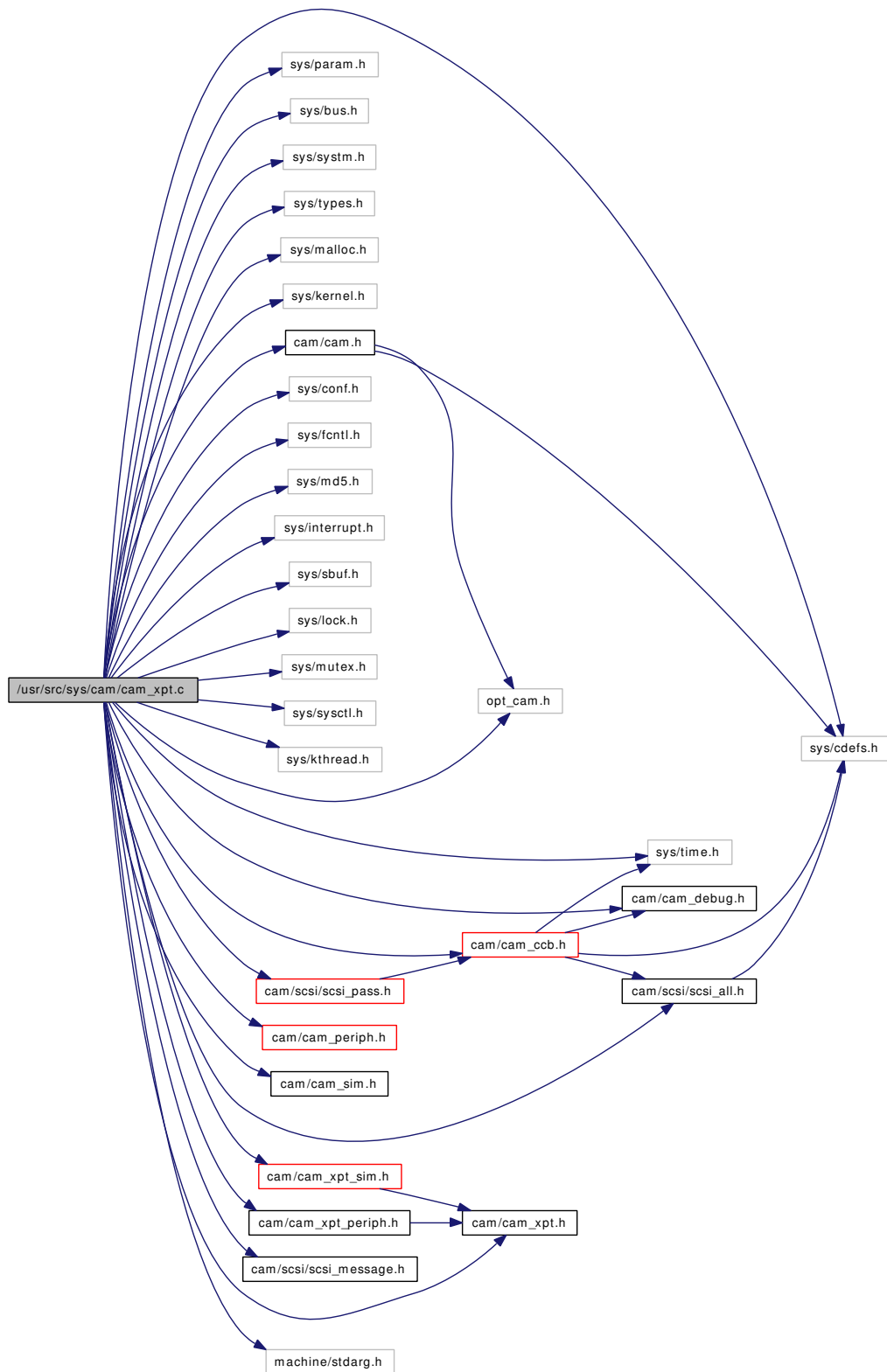
Here is the call graph for this function:



7.12 /usr/src/sys/cam/cam_xpt.c File Reference

```
#include <sys/cdefs.h>
#include <sys/param.h>
#include <sys/bus.h>
#include <sys/system.h>
#include <sys/types.h>
#include <sys/malloc.h>
#include <sys/kernel.h>
#include <sys/time.h>
#include <sys/conf.h>
#include <sys/fcntl.h>
#include <sys/md5.h>
#include <sys/interrupt.h>
#include <sys/sbuf.h>
#include <sys/lock.h>
#include <sys/mutex.h>
#include <sys/sysctl.h>
#include <sys/kthread.h>
#include <cam/cam.h>
#include <cam/cam_ccb.h>
#include <cam/cam_periph.h>
#include <cam/cam_sim.h>
#include <cam/cam_xpt.h>
#include <cam/cam_xpt_sim.h>
#include <cam/cam_xpt_periph.h>
#include <cam/cam_debug.h>
#include <cam/scsi/scsi_all.h>
#include <cam/scsi/scsi_message.h>
#include <cam/scsi/scsi_pass.h>
#include <machine/stdarg.h>
#include "opt_cam.h"
```

Include dependency graph for cam_xpt.c:



Data Structures

- struct [async_node](#)
- struct [cam_ed](#)
- struct [cam_et](#)
- struct [cam_eb](#)
- struct [cam_path](#)
- struct [xpt_quirk_entry](#)
- struct [xpt_softc](#)
- struct [xpt_traverse_config](#)
- struct [xpt_scan_bus_info](#)
- struct [probe_softc](#)

Defines

- #define [CAM_MAX_HIGHPOWER](#) 4
- #define [CAM_DEV_UNCONFIGURED](#) 0x01
- #define [CAM_DEV_REL_TIMEOUT_PENDING](#) 0x02
- #define [CAM_DEV_REL_ON_COMPLETE](#) 0x04
- #define [CAM_DEV_REL_ON_QUEUE_EMPTY](#) 0x08
- #define [CAM_DEV_RESIZE_QUEUE_NEEDED](#) 0x10
- #define [CAM_DEV_TAG_AFTER_COUNT](#) 0x20
- #define [CAM_DEV_INQUIRY_DATA_VALID](#) 0x40
- #define [CAM_DEV_IN_DV](#) 0x80
- #define [CAM_DEV_DV_HIT_BOTTOM](#) 0x100
- #define [CAM_TAG_DELAY_COUNT](#) 5
- #define [CAM_EB_RUNQ_SCHEDULED](#) 0x01
- #define [CAM_QUIRK_NOLUNS](#) 0x01
- #define [CAM_QUIRK_NOSERIAL](#) 0x02
- #define [CAM_QUIRK_HILUNS](#) 0x04
- #define [CAM_QUIRK_NOHILUNS](#) 0x08
- #define [CAM_SCSI2_MAXLUN](#) 8
- #define [CAN_SRCH_HI_SPARSE](#)(dv)
- #define [CAN_SRCH_HI_DENSE](#)(dv)
- #define [SIM_DEAD](#)(sim) ((sim) == &cam_dead_sim)

Typedefs

- typedef int [xpt_busfunc_t](#) (struct [cam_eb](#) *bus, void *arg)
- typedef int [xpt_targetfunc_t](#) (struct [cam_et](#) *target, void *arg)
- typedef int [xpt_devicefunc_t](#) (struct [cam_ed](#) *device, void *arg)
- typedef int [xpt_periphfunc_t](#) (struct [cam_periph](#) *periph, void *arg)
- typedef int [xpt_pdrvfunc_t](#) (struct [periph_driver](#) **pdrv, void *arg)

Enumerations

- enum `xpt_flags` { `XPT_FLAG_OPEN` = 0x01 }
- enum `dev_match_ret` {
`DM_RET_COPY` = 0x01, `DM_RET_FLAG_MASK` = 0x0f, `DM_RET_NONE` = 0x00, `DM_RET_-STOP` = 0x10,
`DM_RET_DESCEND` = 0x20, `DM_RET_ERROR` = 0x30, `DM_RET_ACTION_MASK` = 0xf0 }
- enum `xpt_traverse_depth` { `XPT_DEPTH_BUS`, `XPT_DEPTH_TARGET`, `XPT_DEPTH_DEVICE`, `XPT_DEPTH_PERIPH` }
- enum `probe_action` {
`PROBE_TUR`, `PROBE_INQUIRY`, `PROBE_FULL_INQUIRY`, `PROBE_MODE_SENSE`,
`PROBE_SERIAL_NUM`, `PROBE_TUR_FOR_NEGOTIATION`, `PROBE_INQUIRY_BASIC_-DV1`, `PROBE_INQUIRY_BASIC_DV2`,
`PROBE_DV_EXIT` }
- enum `probe_flags` { `PROBE_INQUIRY_CKSUM` = 0x01, `PROBE_SERIAL_CKSUM` = 0x02, `PROBE_NO_ANNOUNCE` = 0x04 }

Functions

- `__FBSDDID` ("\$FreeBSD: src/sys/cam/cam_xpt.c,v 1.174 2007/02/23 05:47:36 mjacob Exp \$")
- `MALLOC_DEFINE` (M_CAMXPT,"CAM XPT","CAM XPT buffers")
- `SLIST_HEAD` (async_list, async_node)
- `SLIST_HEAD` (periph_list, cam_periph)
- static `TAILQ_HEAD` (highpowerlist, ccb_hdr)
- `TUNABLE_INT` ("kern.cam.cam_srch_hi",&cam_srch_hi)
- static int `sysctl_cam_search_luns` (SYSCTL_HANDLER_ARGS)
- `SYSCTL_PROC` (_kern_cam, OID_AUTO, cam_srch_hi, CTLTYPE_INT|CTLFLAG_RW, 0, 0, sysctl_cam_search_luns,"I","allow search above LUN 7 for SCSI3 and greater devices")
- typedef `TAILQ_HEAD` (cam_isrq, ccb_hdr)
- `PERIPHDRIVER_DECLARE` (xpt, xpt_driver)
- `PERIPHDRIVER_DECLARE` (probe, probe_driver)
- static void `dead_sim_action` (struct cam_sim *sim, union ccb *ccb)
- static void `dead_sim_poll` (struct cam_sim *sim)
- static `TAILQ_HEAD` (cam_eb)
- static void `xpt_init` (void *)
- `DECLARE_MODULE` (cam, cam_moduledata, SI_SUB_CONFIGURE, SI_ORDER_SECOND)
- `MODULE_VERSION` (cam, 1)
- static `cam_status` `xpt_compile_path` (struct cam_path *new_path, struct cam_periph *perph, path_id_t path_id, target_id_t target_id, lun_id_t lun_id)
- static void `xpt_release_path` (struct cam_path *path)
- static void `xpt_async_bcast` (struct async_list *async_head, u_int32_t async_code, struct cam_path *path, void *async_arg)
- static void `xpt_dev_async` (u_int32_t async_code, struct cam_eb *bus, struct cam_et *target, struct cam_ed *device, void *async_arg)
- static `path_id_t` `xptnextfreepathid` (void)
- static `path_id_t` `xptpathid` (const char *sim_name, int sim_unit, int sim_bus)
- static union ccb * `xpt_get_ccb` (struct cam_ed *device)
- static int `xpt_schedule_dev` (struct camq *queue, cam_pinfo *dev_pinfo, u_int32_t new_priority)
- static void `xpt_run_dev_allocq` (struct cam_eb *bus)

- static void `xpt_run_dev_sendq` (struct `cam_eb` *bus)
- static void `xpt_release_bus` (struct `cam_eb` *bus)
- static void `xpt_release_devq_device` (struct `cam_ed` *dev, u_int count, int run_queue)
- static struct `cam_et` * `xpt_alloc_target` (struct `cam_eb` *bus, `target_id_t` target_id)
- static void `xpt_release_target` (struct `cam_eb` *bus, struct `cam_et` *target)
- static struct `cam_ed` * `xpt_alloc_device` (struct `cam_eb` *bus, struct `cam_et` *target, `lun_id_t` lun_id)
- static void `xpt_release_device` (struct `cam_eb` *bus, struct `cam_et` *target, struct `cam_ed` *device)
- static u_int32_t `xpt_dev_ccbq_resize` (struct `cam_path` *path, int newopenings)
- static struct `cam_eb` * `xpt_find_bus` (`path_id_t` path_id)
- static struct `cam_et` * `xpt_find_target` (struct `cam_eb` *bus, `target_id_t` target_id)
- static struct `cam_ed` * `xpt_find_device` (struct `cam_et` *target, `lun_id_t` lun_id)
- static void `xpt_scan_bus` (struct `cam_periph` *periph, union `ccb` *ccb)
- static void `xpt_scan_lun` (struct `cam_periph` *periph, struct `cam_path` *path, `cam_flags` flags, union `ccb` *ccb)
- static void `xptscandone` (struct `cam_periph` *periph, union `ccb` *done_ccb)
- static void `xpt_config` (void *arg)
- static void `xpt_finishconfig` (struct `cam_periph` *periph, union `ccb` *ccb)
- static void `xptaction` (struct `cam_sim` *sim, union `ccb` *work_ccb)
- static void `xtpoll` (struct `cam_sim` *sim)
- static void `camisr` (void *)
- static `dev_match_ret` `xptbusmatch` (struct `dev_match_pattern` *patterns, u_int num_patterns, struct `cam_eb` *bus)
- static `dev_match_ret` `xptdevicematch` (struct `dev_match_pattern` *patterns, u_int num_patterns, struct `cam_ed` *device)
- static `dev_match_ret` `xptperiphmatch` (struct `dev_match_pattern` *patterns, u_int num_patterns, struct `cam_periph` *periph)
- static int `xptedtmatch` (struct `ccb_dev_match` *cdm)
- static int `xptperiphlistmatch` (struct `ccb_dev_match` *cdm)
- static int `xptbustraverse` (struct `cam_eb` *start_bus, `xpt_busfunc_t` *tr_func, void *arg)
- static int `xpttargettraverse` (struct `cam_eb` *bus, struct `cam_et` *start_target, `xpt_targetfunc_t` *tr_func, void *arg)
- static int `xptdevicetraverse` (struct `cam_et` *target, struct `cam_ed` *start_device, `xpt_devicefunc_t` *tr_func, void *arg)
- static int `xptperiphtraverse` (struct `cam_ed` *device, struct `cam_periph` *start_periph, `xpt_periphfunc_t` *tr_func, void *arg)
- static int `xptpdrvtraverse` (struct `periph_driver` **start_pdrv, `xpt_pdrvfunc_t` *tr_func, void *arg)
- static int `xptpdrperiphtraverse` (struct `periph_driver` **pdrv, struct `cam_periph` *start_periph, `xpt_periphfunc_t` *tr_func, void *arg)
- static int `xpt_for_all_busses` (`xpt_busfunc_t` *tr_func, void *arg)
- static int `xpt_for_all_devices` (`xpt_devicefunc_t` *tr_func, void *arg)
- static `cam_status` `xptregister` (struct `cam_periph` *periph, void *arg)
- static `cam_status` `proberegister` (struct `cam_periph` *periph, void *arg)
- static void `probeschedule` (struct `cam_periph` *probe_periph)
- static void `probestart` (struct `cam_periph` *periph, union `ccb` *start_ccb)
- static void `proberequestdefaultnegotiation` (struct `cam_periph` *periph)
- static int `proberequestbackoff` (struct `cam_periph` *periph, struct `cam_ed` *device)
- static void `probedone` (struct `cam_periph` *periph, union `ccb` *done_ccb)
- static void `probecleanup` (struct `cam_periph` *periph)
- static void `xpt_find_quirk` (struct `cam_ed` *device)
- static void `xpt_devise_transport` (struct `cam_path` *path)

- static void `xpt_set_transfer_settings` (struct `ccb_trans_settings` *cts, struct `cam_ed` *device, int async_update)
- static void `xpt_toggle_tags` (struct `cam_path` *path)
- static void `xpt_start_tags` (struct `cam_path` *path)
- static `__inline` int `xpt_schedule_dev_allocq` (struct `cam_eb` *bus, struct `cam_ed` *dev)
- static `__inline` int `xpt_schedule_dev_sendq` (struct `cam_eb` *bus, struct `cam_ed` *dev)
- static `__inline` int `periph_is_queued` (struct `cam_periph` *periph)
- static `__inline` int `device_is_alloc_queued` (struct `cam_ed` *device)
- static `__inline` int `device_is_send_queued` (struct `cam_ed` *device)
- static `__inline` int `dev_allocq_is_runnable` (struct `cam_devq` *devq)
- static void `xpt_periph_init` ()
- static void `probe_periph_init` ()
- static void `xptdone` (struct `cam_periph` *periph, union `ccb` *done_ccb)
- static int `xptopen` (struct `cdev` *dev, int flags, int fmt, struct `thread` *td)
- static int `xptclose` (struct `cdev` *dev, int flag, int fmt, struct `thread` *td)
- static int `xptioctl` (struct `cdev` *dev, u_long cmd, caddr_t addr, int flag, struct `thread` *td)
- static int `cam_module_event_handler` (module_t mod, int what, void *arg)
- static `TAILQ_HEAD` (`ccb_hdr`)
- void `xpt_rescan` (union `ccb` *ccb)
- int32_t `xpt_add_periph` (struct `cam_periph` *periph)
- void `xpt_remove_periph` (struct `cam_periph` *periph)
- void `xpt_announce_periph` (struct `cam_periph` *periph, char *announce_string)
- static int `xptedtbusfunc` (struct `cam_eb` *bus, void *arg)
- static int `xptedttargetfunc` (struct `cam_et` *target, void *arg)
- static int `xptedtdevicefunc` (struct `cam_ed` *device, void *arg)
- static int `xptedtperiphfunc` (struct `cam_periph` *periph, void *arg)
- static int `xptlistpdrvfunc` (struct `periph_driver` **pdrv, void *arg)
- static int `xptlistperiphfunc` (struct `cam_periph` *periph, void *arg)
- static int `xptdefbusfunc` (struct `cam_eb` *bus, void *arg)
- static int `xptdeftargetfunc` (struct `cam_et` *target, void *arg)
- static int `xptdefdevicefunc` (struct `cam_ed` *device, void *arg)
- static int `xptdefperiphfunc` (struct `cam_periph` *periph, void *arg)
- static int `xptsetasyncfunc` (struct `cam_ed` *device, void *arg)
- static int `xptsetasyncbusfunc` (struct `cam_eb` *bus, void *arg)
- void `xpt_action` (union `ccb` *start_ccb)
- void `xpt_polled_action` (union `ccb` *start_ccb)
- void `xpt_schedule` (struct `cam_periph` *periph, u_int32_t new_priority)
- void `xpt_merge_ccb` (union `ccb` *master_ccb, union `ccb` *slave_ccb)
- void `xpt_setup_ccb` (struct `ccb_hdr` *ccb_h, struct `cam_path` *path, u_int32_t priority)
- `cam_status` `xpt_create_path` (struct `cam_path` **new_path_ptr, struct `cam_periph` *periph, `path_id_t` path_id, `target_id_t` target_id, `lun_id_t` lun_id)
- void `xpt_free_path` (struct `cam_path` *path)
- int `xpt_path_comp` (struct `cam_path` *path1, struct `cam_path` *path2)
- void `xpt_print_path` (struct `cam_path` *path)
- void `xpt_print` (struct `cam_path` *path, const char *fmt,...)
- int `xpt_path_string` (struct `cam_path` *path, char *str, size_t str_len)
- `path_id_t` `xpt_path_path_id` (struct `cam_path` *path)
- `target_id_t` `xpt_path_target_id` (struct `cam_path` *path)
- `lun_id_t` `xpt_path_lun_id` (struct `cam_path` *path)
- `cam_sim` * `xpt_path_sim` (struct `cam_path` *path)

- `cam_periph * xpt_path_periph` (struct `cam_path *path`)
- void `xpt_release_ccb` (union `ccb *free_ccb`)
- `int32_t xpt_bus_register` (struct `cam_sim *sim`, `u_int32_t bus`)
- `int32_t xpt_bus_deregister` (`path_id_t pathid`)
- void `xpt_async` (`u_int32_t async_code`, struct `cam_path *path`, void `*async_arg`)
- `u_int32_t xpt_freeze_devq` (struct `cam_path *path`, `u_int count`)
- `u_int32_t xpt_freeze_simq` (struct `cam_sim *sim`, `u_int count`)
- static void `xpt_release_devq_timeout` (void `*arg`)
- void `xpt_release_devq` (struct `cam_path *path`, `u_int count`, `int run_queue`)
- void `xpt_release_simq` (struct `cam_sim *sim`, `int run_queue`)
- static void `xpt_release_simq_timeout` (void `*arg`)
- void `xpt_done` (union `ccb *done_ccb`)
- `ccb * xpt_alloc_ccb` ()
- `ccb * xpt_alloc_ccb_nowait` ()
- void `xpt_free_ccb` (union `ccb *free_ccb`)
- static `int xptconfigbuscountfunc` (struct `cam_eb *bus`, void `*arg`)
- static `int xptconfigfunc` (struct `cam_eb *bus`, void `*arg`)
- static `int xptpassannouncefunc` (struct `cam_ed *device`, void `*arg`)

Variables

- static `int cam_srch_hi` = 0
- static const char `quantum` [] = "QUANTUM"
- static const char `sony` [] = "SONY"
- static const char `west_digital` [] = "WDIGTL"
- static const char `samsung` [] = "SAMSUNG"
- static const char `seagate` [] = "SEAGATE"
- static const char `microp` [] = "MICROP"
- static struct `xpt_quirk_entry xpt_quirk_table` []
- static const `int xpt_quirk_table_size`
- static struct `xpt_softc xsoftc`
- static struct `periph_driver probe_driver`
- static `d_open_t xptopen`
- static `d_close_t xptclose`
- static `d_ioctl_t xptioctl`
- static struct `cdevsw xpt_cdevsw`
- static struct `intr_config_hook * xpt_config_hook`
- static struct `cam_sim cam_dead_sim`
- static `timeout_t xpt_release_devq_timeout`
- static `timeout_t xpt_release_simq_timeout`
- static `xpt_busfunc_t xptconfigbuscountfunc`
- static `xpt_busfunc_t xptconfigfunc`
- static `xpt_devicefunc_t xptpassannouncefunc`
- static `xpt_busfunc_t xptedtbusfunc`
- static `xpt_targetfunc_t xptedttargetfunc`
- static `xpt_devicefunc_t xptedtdevicefunc`
- static `xpt_periphfunc_t xptedtperiphfunc`
- static `xpt_pdrvfunc_t xptlistpdrvfunc`
- static `xpt_periphfunc_t xptlistperiphfunc`
- static `xpt_busfunc_t xptdefbusfunc`

- static [xpt_targetfunc_t](#) [xptdeftargetfunc](#)
- static [xpt_devicefunc_t](#) [xptdefdevicefunc](#)
- static [xpt_periphfunc_t](#) [xptdefperiphfunc](#)
- static [xpt_devicefunc_t](#) [xptsetasyncfunc](#)
- static [xpt_busfunc_t](#) [xptsetasynbusfunc](#)
- static int [busses_to_config](#)
- static int [busses_to_reset](#)

7.12.1 Define Documentation

7.12.1.1 #define CAM_DEV_DV_HIT_BOTTOM 0x100

Referenced by [probedone\(\)](#), and [proberequestbackoff\(\)](#).

7.12.1.2 #define CAM_DEV_IN_DV 0x80

Referenced by [probedone\(\)](#).

7.12.1.3 #define CAM_DEV_INQUIRY_DATA_VALID 0x40

Referenced by [probedone\(\)](#), [xpt_devise_transport\(\)](#), and [xpt_set_transfer_settings\(\)](#).

7.12.1.4 #define CAM_DEV_REL_ON_COMPLETE 0x04

Referenced by [camisr\(\)](#), [xpt_action\(\)](#), and [xpt_release_devq_device\(\)](#).

7.12.1.5 #define CAM_DEV_REL_ON_QUEUE_EMPTY 0x08

Referenced by [camisr\(\)](#), and [xpt_action\(\)](#).

7.12.1.6 #define CAM_DEV_REL_TIMEOUT_PENDING 0x02

Referenced by [xpt_action\(\)](#), [xpt_release_device\(\)](#), and [xpt_release_devq_device\(\)](#).

7.12.1.7 #define CAM_DEV_RESIZE_QUEUE_NEEDED 0x10

Referenced by [xpt_dev_ccbq_resize\(\)](#), and [xpt_schedule_dev_allocq\(\)](#).

7.12.1.8 #define CAM_DEV_TAG_AFTER_COUNT 0x20

Referenced by [camisr\(\)](#), [xpt_announce_periph\(\)](#), [xpt_dev_ccbq_resize\(\)](#), [xpt_set_transfer_settings\(\)](#), [xpt_start_tags\(\)](#), and [xpt_toggle_tags\(\)](#).

7.12.1.9 #define CAM_DEV_UNCONFIGURED 0x01

Referenced by [probedone\(\)](#), [probeschedule\(\)](#), [probestart\(\)](#), [xpt_action\(\)](#), [xpt_alloc_device\(\)](#), [xpt_dev_async\(\)](#), [xpt_release_device\(\)](#), [xptdtdevicefunc\(\)](#), and [xptsetasyncfunc\(\)](#).

7.12.1.10 #define CAM_EB_RUNQ_SCHEDULED 0x01**7.12.1.11 #define CAM_MAX_HIGHPOWER 4**

Referenced by STAILQ_HEAD().

7.12.1.12 #define CAM_QUIRK_HILUNS 0x04

Definition at line 209 of file cam_xpt.c.

7.12.1.13 #define CAM_QUIRK_NOHILUNS 0x08

Definition at line 210 of file cam_xpt.c.

7.12.1.14 #define CAM_QUIRK_NOLUNS 0x01

Definition at line 207 of file cam_xpt.c.

Referenced by xpt_scan_bus().

7.12.1.15 #define CAM_QUIRK_NOSERIAL 0x02

Definition at line 208 of file cam_xpt.c.

Referenced by probestart().

7.12.1.16 #define CAM_SCSI2_MAXLUN 8

Definition at line 222 of file cam_xpt.c.

Referenced by xpt_scan_bus().

7.12.1.17 #define CAM_TAG_DELAY_COUNT 5

Referenced by xpt_set_transfer_settings().

7.12.1.18 #define CAN_SRCH_HI_DENSE(dv)**Value:**

```
((dv->quirk->quirks & CAM_QUIRK_NOHILUNS) == 0)      \  
&& ((dv->quirk->quirks & CAM_QUIRK_HILUNS)          \  
    || (SID_ANSI_REV(&dv->inq_data) > SCSI_REV_2))
```

Definition at line 235 of file cam_xpt.c.

Referenced by xpt_scan_bus().

7.12.1.19 #define CAN_SRCH_HI_SPARSE(dv)**Value:**

```
((dv->quirk->quirks & CAM_QUIRK_NOHILUNS) == 0)          \
  && ((dv->quirk->quirks & CAM_QUIRK_HILUNS)              \
     || (SID_ANSI_REV(&dv->inq_data) > SCSI_REV_2 && cam_srch_hi))
```

Definition at line 230 of file cam_xpt.c.

Referenced by xpt_scan_bus().

7.12.1.20 #define SIM_DEAD(sim) ((sim) == &cam_dead_sim)

Definition at line 706 of file cam_xpt.c.

Referenced by camisr(), xpt_action(), xpt_alloc_device(), xpt_release_device(), and xpt_schedule().

7.12.2 Typedef Documentation**7.12.2.1 typedef int xpt_busfunc_t(struct cam_eb *bus, void *arg)**

Definition at line 636 of file cam_xpt.c.

7.12.2.2 typedef int xpt_devicefunc_t(struct cam_ed *device, void *arg)

Definition at line 638 of file cam_xpt.c.

7.12.2.3 typedef int xpt_pdrvfunc_t(struct periph_driver **pdrv, void *arg)

Definition at line 640 of file cam_xpt.c.

7.12.2.4 typedef int xpt_periphfunc_t(struct cam_periph *periph, void *arg)

Definition at line 639 of file cam_xpt.c.

7.12.2.5 typedef int xpt_targetfunc_t(struct cam_et *target, void *arg)

Definition at line 637 of file cam_xpt.c.

7.12.3 Enumeration Type Documentation**7.12.3.1 enum dev_match_ret****Enumerator:**

```
DM_RET_COPY
DM_RET_FLAG_MASK
DM_RET_NONE
```

DM_RET_STOP
DM_RET_DESCEND
DM_RET_ERROR
DM_RET_ACTION_MASK

Definition at line 613 of file cam_xpt.c.

7.12.3.2 enum [probe_action](#)

Enumerator:

PROBE_TUR
PROBE_INQUIRY
PROBE_FULL_INQUIRY
PROBE_MODE_SENSE
PROBE_SERIAL_NUM
PROBE_TUR_FOR_NEGOTIATION
PROBE_INQUIRY_BASIC_DV1
PROBE_INQUIRY_BASIC_DV2
PROBE_DV_EXIT

Definition at line 5524 of file cam_xpt.c.

7.12.3.3 enum [probe_flags](#)

Enumerator:

PROBE_INQUIRY_CKSUM
PROBE_SERIAL_CKSUM
PROBE_NO_ANNOUNCE

Definition at line 5536 of file cam_xpt.c.

7.12.3.4 enum [xpt_flags](#)

Enumerator:

XPT_FLAG_OPEN

Definition at line 240 of file cam_xpt.c.

7.12.3.5 enum [xpt_traverse_depth](#)

Enumerator:

XPT_DEPTH_BUS
XPT_DEPTH_TARGET
XPT_DEPTH_DEVICE
XPT_DEPTH_PERIPH

Definition at line 623 of file cam_xpt.c.

7.12.4 Function Documentation

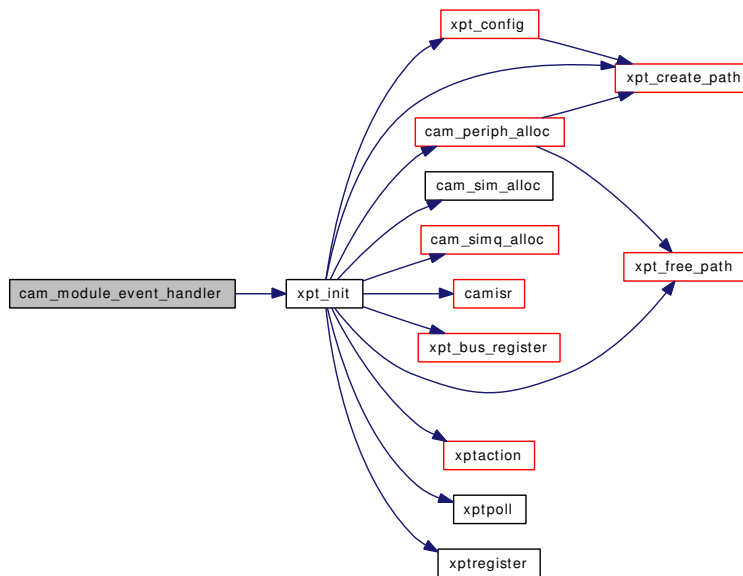
7.12.4.1 `__FBSDID("$FreeBSD: src/sys/cam/cam_xpt.c, v 1.174 2007/02/23 05:47:36 mjacob Exp $")`

7.12.4.2 `static int cam_module_event_handler (module_t mod, int what, void * arg) [static]`

Definition at line 1403 of file cam_xpt.c.

References xpt_init().

Here is the call graph for this function:



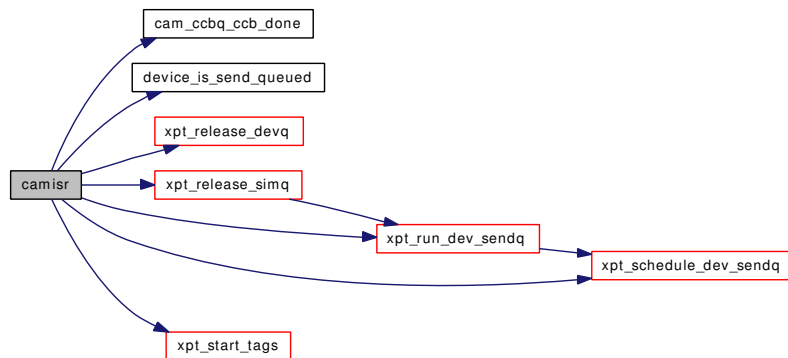
7.12.4.3 `static void camisir (void *) [static]`

Definition at line 7115 of file cam_xpt.c.

References cam_path::bus, cam_ccbq_ccb_done(), CAM_DEBUG, CAM_DEBUG_TRACE, CAM_DEV_QFRZDIS, CAM_DEV_QFRZN, CAM_DEV_REL_ON_COMPLETE, CAM_DEV_REL_ON_QUEUE_EMPTY, CAM_DEV_TAG_AFTER_COUNT, CAM_HIGH_POWER, CAM_RELEASE_SIMQ, CAM_REQUEUE_REQ, CAM_STATUS_MASK, CAM_UNQUEUED_INDEX, ccb_hdr::cbfnp, ccb::ccb_h, cam_path::device, device_is_send_queued(), ccb_hdr::flags, ccb_hdr::func_code, cam_pinfo::index, ccb_hdr::path, cam_path::periph, ccb_hdr::pinfo, SIM_DEAD, ccb_hdr::sim_links, ccb_hdr::status, XPT_FC_USER_CCB, xpt_release_devq(), xpt_release_simq(), xpt_run_dev_sendq(), xpt_schedule_dev_sendq(), and xpt_start_tags().

Referenced by xpt_bus_deregister(), xpt_init(), and xpt_polled_action().

Here is the call graph for this function:

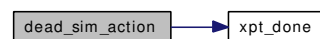


7.12.4.4 `static void dead_sim_action (struct cam_sim * sim, union ccb * ccb)` [static]

Definition at line 7230 of file `cam_xpt.c`.

References `CAM_DEV_NOT_THERE`, `ccb::ccb_h`, `ccb_hdr::status`, and `xpt_done()`.

Here is the call graph for this function:



7.12.4.5 `static void dead_sim_poll (struct cam_sim * sim)` [static]

Definition at line 7238 of file `cam_xpt.c`.

7.12.4.6 `DECLARE_MODULE (cam, cam_moduledata, SI_SUB_CONFIGURE, SI_ORDER_SECOND)`

7.12.4.7 `static __inline int dev_allocq_is_runnable (struct cam_devq * devq)` [static]

Definition at line 965 of file `cam_xpt.c`.

References `cam_devq::alloc_openings`, `cam_devq::alloc_queue`, `camq::entries`, and `camq::qfrozen_cnt`.

Referenced by `xpt_release_ccb()`.

7.12.4.8 `static __inline int device_is_alloc_queued (struct cam_ed * device)` [static]

Definition at line 953 of file `cam_xpt.c`.

References `CAM_UNQUEUED_INDEX`.

Referenced by `xpt_release_ccb()`.

7.12.4.9 `static __inline int device_is_send_queued (struct cam_ed * device)` [static]

Definition at line 959 of file `cam_xpt.c`.

References CAM_UNQUEUED_INDEX.

Referenced by camisr().

7.12.4.10 MALLOC_DEFINE (M_CAMXPT, "CAM XPT", "CAM XPT buffers")

7.12.4.11 MODULE_VERSION (cam, 1)

7.12.4.12 static __inline int periph_is_queued (struct [cam_periph](#) * *periph*) [static]

Definition at line 947 of file cam_xpt.c.

References CAM_UNQUEUED_INDEX, cam_pinfo::index, and cam_periph::pinfo.

Referenced by xpt_schedule().

7.12.4.13 PERIPHDRIIVER_DECLARE (probe, [probe_driver](#))

7.12.4.14 PERIPHDRIIVER_DECLARE (xpt, [xpt_driver](#))

7.12.4.15 static void probe_periph_init () [static]

Definition at line 984 of file cam_xpt.c.

7.12.4.16 static void probecleanup (struct [cam_periph](#) * *periph*) [static]

Definition at line 6392 of file cam_xpt.c.

References cam_periph::softc.

Referenced by xpt_scan_lun().

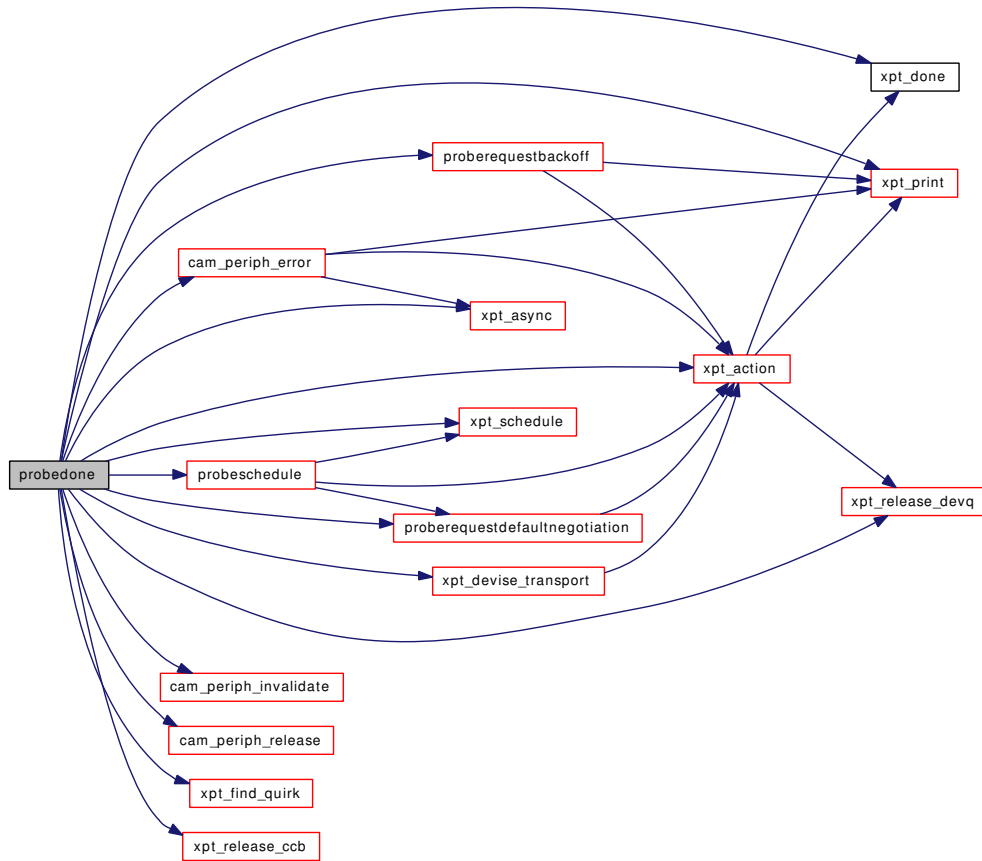
7.12.4.17 static void probedone (struct [cam_periph](#) * *periph*, union [ccb](#) * *done_ccb*) [static]

Definition at line 6035 of file cam_xpt.c.

References AC_FOUND_DEVICE, AC_LOST_DEVICE, scsi_inquiry_data::additional_length, scsi_mode_header_6::blk_desc_len, CAM_DEBUG, CAM_DEBUG_INFO, CAM_DEBUG_TRACE, CAM_DEV_DV_HIT_BOTTOM, CAM_DEV_IN_DV, CAM_DEV_INQUIRY_DATA_VALID, CAM_DEV_QFRZN, CAM_DEV_UNCONFIGURED, cam_periph_error(), cam_periph_invalidate(), cam_periph_release(), CAM_REQ_CMP, CAM_STATUS_MASK, ccb_scsiio::ccb_h, ccb::ccb_h, ccb::csio, ccb_scsiio::data_ptr, cam_path::device, ccb_hdr::func_code, INQ_DATA_TQ_ENABLED, scsi_vpd_unit_serial_number::length, cam_periph::path, ccb_hdr::path, cam_path::periph, ccb_hdr::pinfo, cam_pinfo::priority, PROBE_DV_EXIT, PROBE_FULL_INQUIRY, PROBE_INQUIRY, PROBE_INQUIRY_BASIC_DV1, PROBE_INQUIRY_BASIC_DV2, PROBE_INQUIRY_CKSUM, PROBE_MODE_SENSE, PROBE_NO_ANNOUNCE, PROBE_SERIAL_NUM, PROBE_TUR, PROBE_TUR_FOR_NEGOTIATION, proberequestbackoff(), proberequestdefaultnegotiation(), probeschedule(), scsi_control_page::queue_flags, scsi_vpd_unit_serial_number::serial_num, SF_NO_PRINT, SF_QUIET_IR, SF_RETRY_UA, SHORT_INQUIRY_LENGTH, SID_QUAL, SID_QUAL_LU_CONNECTED, SID_Sync, cam_periph::softc, ccb_hdr::status, ccb_hdr::target_lun, xpt_action(), xpt_async(), xpt_devisetransport(), xpt_done(), xpt_find_quirk(), XPT_GDEV_TYPE, xpt_print(), xpt_release_ccb(), xpt_release_devq(), and xpt_schedule().

Referenced by probestart().

Here is the call graph for this function:



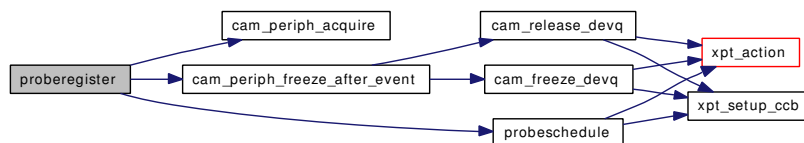
7.12.4.18 static `cam_status` proberegister (`struct cam_periph *periph, void *arg`) [static]

Definition at line 5653 of file `cam_xpt.c`.

References `cam_path::bus`, `cam_periph_acquire()`, `cam_periph_freeze_after_event()`, `CAM_REQ_CMP`, `CAM_REQ_CMP_ERR`, `ccb::ccb_h`, `cam_periph::path`, `probeschedule()`, `scsi_delay`, and `cam_periph::softc`.

Referenced by `xpt_scan_lun()`.

Here is the call graph for this function:



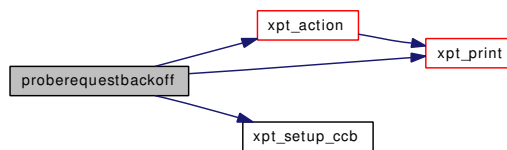
7.12.4.19 static int proberequestbackoff (struct **cam_periph** * *periph*, struct **cam_ed** * *device*) [static]

Definition at line 5933 of file cam_xpt.c.

References CAM_DEBUG, CAM_DEBUG_INFO, CAM_DEV_DV_HIT_BOTTOM, CAM_REQ_CMP, CAM_STATUS_MASK, ccb::ccb_h, CTS_SPI_VALID_SYNC_OFFSET, CTS_SPI_VALID_SYNC_RATE, CTS_TYPE_CURRENT_SETTINGS, ccb_hdr::func_code, cam_periph::path, ccb_hdr::status, ccb_trans_settings_spi::sync_offset, ccb_trans_settings_spi::sync_period, ccb_trans_settings_spi::valid, XPORT_SPI, xpt_action(), XPT_GET_TRAN_SETTINGS, xpt_print(), XPT_SET_TRAN_SETTINGS, and xpt_setup_ccb().

Referenced by probedone().

Here is the call graph for this function:



7.12.4.20 static void proberequestdefaultnegotiation (struct **cam_periph** * *periph*) [static]

Definition at line 5913 of file cam_xpt.c.

References CAM_REQ_CMP, CAM_STATUS_MASK, ccb::ccb_h, ccb_trans_settings::ccb_h, CTS_TYPE_CURRENT_SETTINGS, CTS_TYPE_USER_SETTINGS, ccb_hdr::func_code, cam_periph::path, ccb_hdr::status, ccb_trans_settings::type, xpt_action(), XPT_GET_TRAN_SETTINGS, XPT_SET_TRAN_SETTINGS, and xpt_setup_ccb().

Referenced by probedone(), and probeschedule().

Here is the call graph for this function:



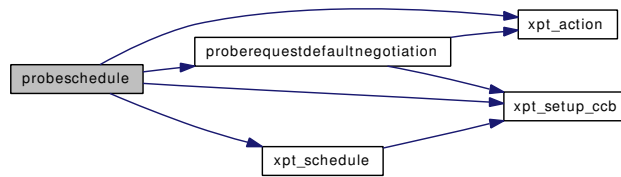
7.12.4.21 static void probeschedule (struct **cam_periph** * *probe_periph*) [static]

Definition at line 5695 of file cam_xpt.c.

References CAM_DEV_UNCONFIGURED, CAM_EXPECT_INQ_CHANGE, ccb_pathinq::ccb_h, ccb::cpi, ccb_hdr::func_code, ccb_pathinq::hba_inquiry, ccb_pathinq::hba_misc, cam_periph::path, PI_SDTR_ABLE, PI_WIDE_16, PI_WIDE_32, PIM_NOBUSRESET, cam_periph::pinfo, cam_pinfo::priority, PROBE_INQUIRY, PROBE_NO_ANNOUNCE, PROBE_TUR, proberequestdefaultnegotiation(), cam_periph::softc, xpt_action(), XPT_PATH_INQ, xpt_schedule(), and xpt_setup_ccb().

Referenced by probedone(), and proberegister().

Here is the call graph for this function:



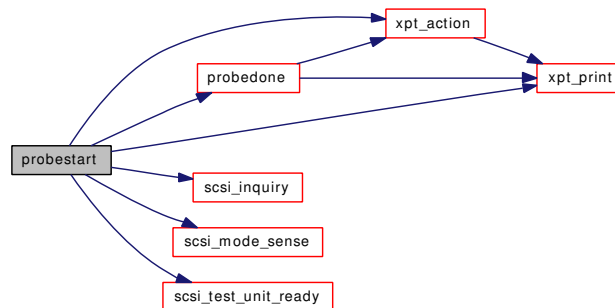
7.12.4.22 static void probestart (struct cam_periph * periph, union ccb * start_ccb) [static]

Definition at line 5746 of file cam_xpt.c.

References CAM_DEBUG, CAM_DEBUG_TRACE, CAM_DEV_UNCONFIGURED, CAM_QUIRK_NOSERIAL, ccb::ccb_h, ccb::csio, ccb_scsiio::data_ptr, cam_path::device, MSG_SIMPLE_Q_TAG, cam_periph::path, ccb_hdr::path, PROBE_DV_EXIT, PROBE_FULL_INQUIRY, PROBE_INQUIRY, PROBE_INQUIRY_BASIC_DV1, PROBE_INQUIRY_BASIC_DV2, PROBE_INQUIRY_CKSUM, PROBE_MODE_SENSE, PROBE_SERIAL_CKSUM, PROBE_SERIAL_NUM, PROBE_TUR, PROBE_TUR_FOR_NEGOTIATION, probedone(), scsi_inquiry(), scsi_mode_sense(), scsi_test_unit_ready(), SHORT_INQUIRY_LENGTH, SID_ADDITIONAL_LENGTH, SMS_CONTROL_MODE_PAGE, SMS_PAGE_CTRL_CURRENT, cam_periph::softc, SSD_FULL_SIZE, SSD_MIN_SIZE, SVPD_UNIT_SERIAL_NUMBER, xpt_action(), and xpt_print().

Referenced by xpt_scan_lun().

Here is the call graph for this function:



7.12.4.23 SLIST_HEAD (periph_list, cam_periph)

7.12.4.24 SLIST_HEAD (async_list, async_node)

7.12.4.25 static STAILQ_HEAD (highpowerlist, ccb_hdr) [static]

Definition at line 87 of file cam_xpt.c.

References CAM_MAX_HIGHPOWER.

7.12.4.26 static int sysctl_cam_search_luns (SYSCTL_HANDLER_ARGS) [static]

Definition at line 6414 of file cam_xpt.c.

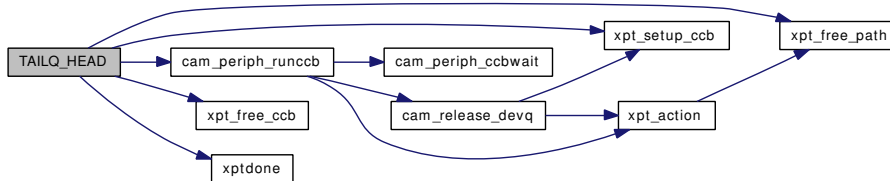
7.12.4.27 `SYSCTL_PROC` (`_kern_cam`, `OID_AUTO`, `cam_srch_hi`, `CTLTYPE_INT` | `CTLFLAG_RW`, `0`, `0`, `sysctl_cam_search_luns`, `"I"`, `"allow search above LUN 7 for SCSI3 and greater devices"`)

7.12.4.28 `static TAILQ_HEAD` (`ccb_hdr`) [`static`]

Definition at line 1417 of file `cam_xpt.c`.

References `cam_periph_runccb()`, `ccb::ccb_h`, `xpt_free_ccb()`, `xpt_free_path()`, `XPT_SCAN_BUS`, `xpt_setup_ccb()`, and `xptdone()`.

Here is the call graph for this function:



7.12.4.29 `static TAILQ_HEAD` (`cam_eb`) [`static`]

Definition at line 709 of file `cam_xpt.c`.

7.12.4.30 `typedef TAILQ_HEAD` (`cam_isrq`, `ccb_hdr`)

Definition at line 646 of file `cam_xpt.c`.

7.12.4.31 `TUNABLE_INT` (`"kern.cam.cam_srch_hi"`, `&cam_srch_hi`)

7.12.4.32 `void xpt_action` (`union ccb * start_ccb`)

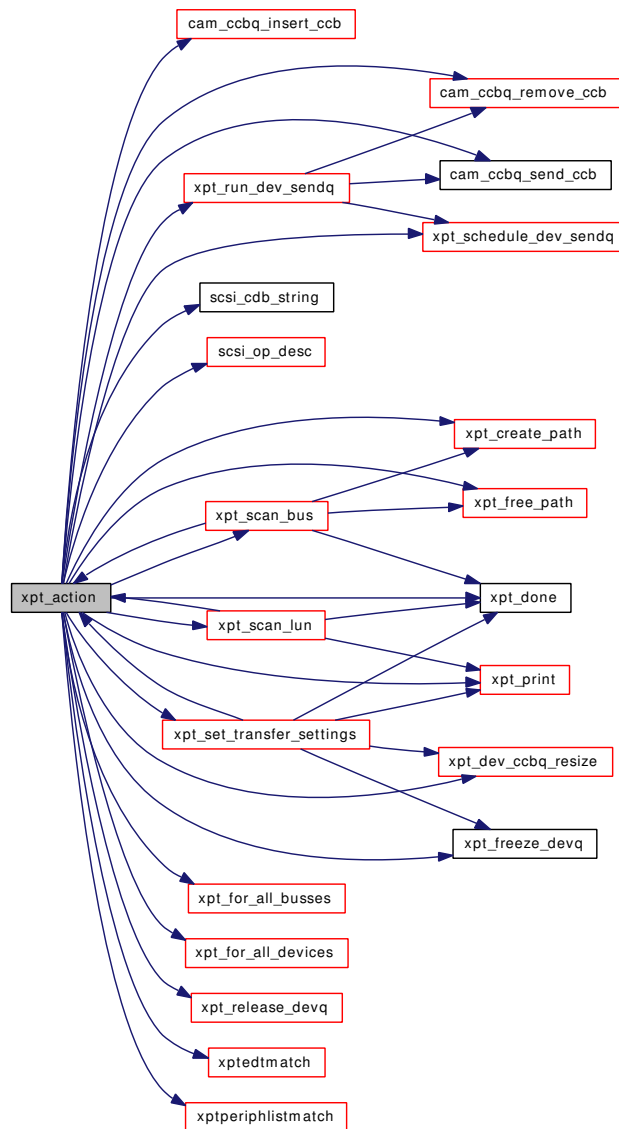
Definition at line 2963 of file `cam_xpt.c`.

References `ccb_abort::abort_ccb`, `AC_FOUND_DEVICE`, `AC_PATH_REGISTERED`, `ccb_calc_geometry::block_size`, `cam_path::bus`, `ccb::cab`, `ccb_setasync::callback`, `ccb_setasync::callback_arg`, `cam_ccbq_insert_ccb()`, `cam_ccbq_remove_ccb()`, `cam_ccbq_send_ccb()`, `CAM_CDB_POINTER`, `CAM_DEBUG`, `CAM_DEBUG_CDB`, `CAM_DEBUG_NONE`, `CAM_DEBUG_TRACE`, `CAM_DEV_MISMATCH_ERROR`, `CAM_DEV_NOT_THERE`, `CAM_DEV_POS_EDT`, `CAM_DEV_POS_NONE`, `CAM_DEV_POS_PDRV`, `CAM_DEV_POS_TYPMASK`, `CAM_DEV_QFREEZE`, `CAM_DEV_QFRZN`, `CAM_DEV_REL_ON_COMPLETE`, `CAM_DEV_REL_ON_QUEUE_EMPTY`, `CAM_DEV_REL_TIMEOUT_PENDING`, `CAM_DEV_UNCONFIGURED`, `CAM_DONEQ_INDEX`, `CAM_FUNC_NOTAVAIL`, `CAM_GDEVLIST_ERROR`, `CAM_GDEVLIST_LAST_DEVICE`, `CAM_GDEVLIST_LIST_CHANGED`, `CAM_GDEVLIST_MORE_DEVS`, `CAM_PROVIDE_FAIL`, `CAM_REQ_ABORTED`, `CAM_REQ_CMP`, `CAM_REQ_CMP_ERR`, `CAM_REQ_INPROG`, `CAM_RESRC_UNAVAIL`, `CAM_SIM_QUEUED`, `CAM_UA_ABORT`, `CAM_UNQUEUED_INDEX`, `ccb::ccb_h`, `ccb_getdev::ccb_h`, `ccb_getdevstats::ccb_h`, `ccb_getdevlist::ccb_h`, `ccb_setasync::ccb_h`, `ccb_relsim::ccb_h`, `ccb::ccg`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `ccb::cdbg`, `ccb::cdm`, `ccb::cgd`, `ccb::cgdl`, `ccb::cgds`, `ccb::cpis`, `ccb::crcn`, `ccb::crs`, `ccb::csa`, `ccb::csio`, `ccb::cts`, `ccb_calc_geometry::cylinders`,

ccb_getdevstats::dev_active, DEV_IDLEN, DEV_MATCH_BUS, DEV_MATCH_DEVICE, ccb_getdevstats::dev_openings, cam_path::device, ccb_getdevstats::devq_openings, ccb_getdevstats::devq_queued, ccb_setasynch::event_enable, ccb_hdr::flags, ccb_rescan::flags, ccb_debug::flags, ccb_hdr::func_code, ccb_getdevlist::generation, ccb_calc_geometry::heads, ccb_getdevstats::held, cam_pinfo::index, ccb_getdevlist::index, ccb_getdev::inq_data, INQ_DATA_TQ_ENABLED, ccb_pathstats::last_reset, ccb_getdevstats::last_reset, ccb_getdevstats::maxtags, ccb_getdevstats::mintags, ccb_dev_match::num_patterns, ccb_relsim::openings, ccb_hdr::path, ccb_hdr::path_id, ccb_dev_match::patterns, cam_path::periph, ccb_getdevlist::periph_name, cam_periph::periph_name, ccb_hdr::pinfo, ccb_dev_match::pos, ccb_dev_position::position_type, ccb_relsim::qfrozen_cnt, ccb_relsim::release_flags, ccb_relsim::release_timeout, RELSIM_ADJUST_OPENINGS, RELSIM_RELEASE_AFTER_CMDCMPLT, RELSIM_RELEASE_AFTER_QEMPTY, RELSIM_RELEASE_AFTER_TIMEOUT, ccb_scsiio::resid, scsi_cdb_string(), SCSI_MAX_CDBLEN, scsi_op_desc(), SCSI_REV_2, ccb_scsiio::scsi_status, SCSI_STATUS_OK, ccb_calc_geometry::secs_per_track, ccb_scsiio::sense_resid, ccb_getdev::serial_num, ccb_getdev::serial_num_len, cam_sim::sim_action, SIM_DEAD, ccb_hdr::status, ccb_getdevlist::status, cam_path::target, ccb_hdr::target_id, ccb_hdr::target_lun, dev_match_pattern::type, ccb_getdevlist::unit_number, cam_periph::unit_number, ccb_calc_geometry::volume_size, XPT_ABORT, XPT_ACCEPT_TARGET_IO, XPT_CALC_GEOMETRY, XPT_CONT_TARGET_IO, xpt_create_path(), XPT_DEBUG, xpt_dev_ccbq_resize(), XPT_DEV_MATCH, xpt_done(), XPT_EN_LUN, XPT_ENG_EXEC, XPT_ENG_INQ, XPT_FC_IS_DEV_QUEUED, XPT_FC_IS_QUEUED, xpt_for_all_busses(), xpt_for_all_devices(), xpt_free_path(), xpt_freeze_devq(), XPT_GDEV_STATS, XPT_GDEV_TYPE, XPT_GDEVLIST, XPT_GET_TRAN_SETTINGS, XPT_IMMEDIATE_NOTIFY, XPT_NOOP, XPT_NOTIFY_ACK, XPT_PATH_INQ, XPT_PATH_STATS, xpt_periph, xpt_print(), XPT_REL_SIMQ, xpt_release_devq(), XPT_RESET_BUS, XPT_RESET_DEV, xpt_run_dev_sendq(), XPT_SASYN_CB, XPT_SCAN_BUS, xpt_scan_bus(), XPT_SCAN_LUN, xpt_scan_lun(), xpt_schedule_dev_sendq(), XPT SCSI_IO, XPT_SDEV_TYPE, XPT_SET_TRAN_SETTINGS, xpt_set_transfer_settings(), XPT_TARGET_IO, XPT_TERM_IO, xptedtmatch(), and xptperiphlistmatch().

Referenced by abort_all_pending(), cam_freeze_devq(), cam_periph_bus_settle(), cam_periph_error(), cam_periph_ioctl(), cam_periph_runccb(), cam_release_devq(), camperiphdone(), camperiphfree(), camperiphscsisenseerror(), camperiphscsistatuserror(), cd6byteworkaround(), cddone(), cdoninvalidate(), cdregister(), cdstart(), chdone(), chinit(), choninvalidate(), chregister(), chstart(), cmd6workaround(), dadone(), dainit(), daoninvalidate(), daregister(), dasetgeom(), dastart(), passinit(), passoninvalidate(), passregister(), probedone(), proberequestbackoff(), proberequestdefaultnegotiation(), probeschedule(), probestart(), ptctor(), ptinit(), ptoninvalidate(), ptstart(), sainit(), saoninvalidate(), saregister(), sastart(), scsi_command_string(), scsi_sense_sbuf(), sesinit(), sesoninvalidate(), sesregister(), STAILQ_HEAD(), targbhdslun(), targbhdone(), targbhenlun(), targbhinit(), targbhstart(), targenable(), targendislun(), targioctl(), targsendccb(), targwrite(), xpt_announce_periph(), xpt_bus_register(), xpt_devise_transport(), xpt_finishconfig(), xpt_polled_action(), xpt_scan_bus(), xpt_scan_lun(), xpt_set_transfer_settings(), xpt_start_tags(), xptconfigbuscountfunc(), xptconfigfunc(), xptioctl(), xptsetasynchbusfunc(), and xptsetasynchfunc().

Here is the call graph for this function:



7.12.4.33 int32_t xpt_add_periph (struct cam_periph * periph)

Definition at line 1559 of file `cam_xpt.c`.

References `CAM_REQ_CMP`, `camq_resize()`, `cam_path::device`, `xpt_softc::generation`, `cam_periph::path`, and `xsoftc`.

Referenced by `cam_periph_alloc()`.

Here is the call graph for this function:



7.12.4.34 union `ccb*` `xpt_alloc_ccb` (void)

Definition at line 4929 of file `cam_xpt.c`.

Referenced by `passioctl()`, `xpt_scan_bus()`, `xptconfigfunc()`, and `xptioctl()`.

7.12.4.35 union `ccb*` `xpt_alloc_ccb_nowait` (void)

Definition at line 4940 of file `cam_xpt.c`.

Referenced by `xpt_get_ccb()`.

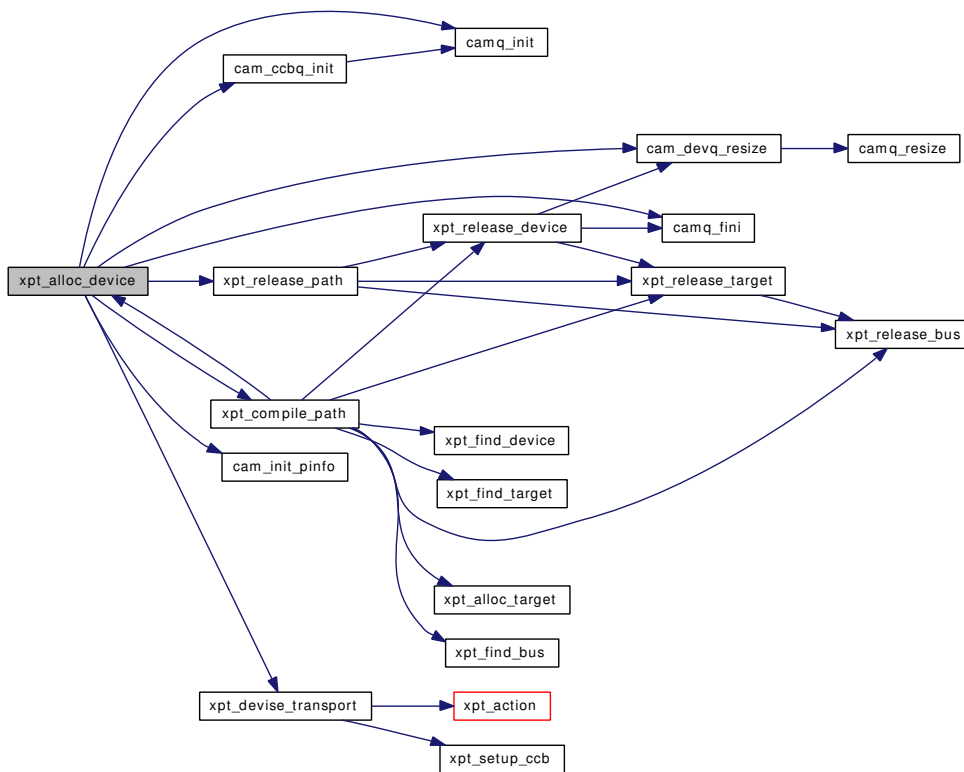
7.12.4.36 static struct `cam_ed` * `xpt_alloc_device` (struct `cam_eb` * `bus`, struct `cam_et` * `target`, `lun_id_t` `lun_id`) [static]

Definition at line 5061 of file `cam_xpt.c`.

References `cam_devq::alloc_queue`, `camq::array_size`, `cam_ccbq_init()`, `CAM_DEV_UNCONFIGURED`, `cam_devq_resize()`, `cam_init_pinfo()`, `CAM_LUN_WILDCARD`, `CAM_REQ_CMP`, `camq_fini()`, `camq_init()`, `SIM_DEAD`, `xpt_compile_path()`, `xpt_devises_transport()`, and `xpt_release_path()`.

Referenced by `xpt_compile_path()`.

Here is the call graph for this function:



7.12.4.37 `static struct cam_et * xpt_alloc_target (struct cam_eb * bus, target_id_t target_id)`
 [static]

Definition at line 5008 of file cam_xpt.c.

Referenced by xpt_compile_path().

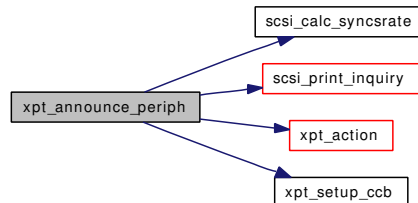
7.12.4.38 `void xpt_announce_periph (struct cam_periph * periph, char * announce_string)`

Definition at line 1629 of file cam_xpt.c.

References ccb_trans_settings_fc::bitrate, ccb_trans_settings_sas::bitrate, cam_path::bus, ccb_trans_settings_spi::bus_width, CAM_DEV_TAG_AFTER_COUNT, CAM_REQ_CMP, CAM_STATUS_MASK, ccb_trans_settings::ccb_h, ccb::ccb_h, ccb_pathinq::ccb_h, CTS_FC_VALID_PORT, CTS_FC_VALID_SPEED, CTS_FC_VALID_WWNN, CTS_FC_VALID_WWPN, CTS_SAS_VALID_SPEED, CTS_SPI_VALID_BUS_WIDTH, CTS_SPI_VALID_SYNC_OFFSET, CTS_TYPE_CURRENT_SETTINGS, cam_path::device, ccb_hdr::func_code, MSG_EXT_PPR_DT_REQ, cam_periph::path, cam_path::periph, cam_periph::periph_name, ccb_trans_settings_fc::port, ccb_trans_settings_spi::ppr_options, scsi_calc_syncsrates(), scsi_print_inquiry(), SID_CmdQue, ccb_hdr::status, ccb_trans_settings_spi::sync_offset, ccb_trans_settings_spi::sync_period, cam_path::target, ccb_trans_settings::type, cam_periph::unit_number, ccb_trans_settings_spi::valid, ccb_trans_settings_fc::valid, ccb_trans_settings_sas::valid, ccb_trans_settings_fc::wwnn, ccb_trans_settings_fc::wwpn, XPORT_FC, XPORT_SAS, XPORT_SPI, xpt_action(), XPT_GET_TRAN_SETTINGS, XPT_PATH_INQ, and xpt_setup_ccb().

Referenced by cddone(), chdone(), dadone(), passregister(), ptctor(), saregister(), sesregister(), and xptpassannouncefunc().

Here is the call graph for this function:



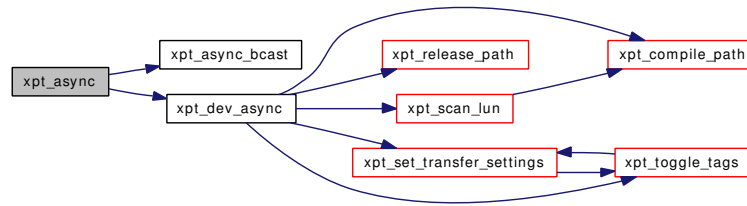
7.12.4.39 `void xpt_async (u_int32_t async_code, struct cam_path * path, void * async_arg)`

Definition at line 4564 of file cam_xpt.c.

References AC_BUS_RESET, AC_SENT_BDR, cam_path::bus, CAM_DEBUG, CAM_DEBUG_TRACE, CAM_LUN_WILDCARD, CAM_TARGET_WILDCARD, cam_path::device, cam_periph::path, cam_path::target, xpt_async_bcast(), xpt_dev_async(), and xpt_periph.

Referenced by cam_periph_error(), camperiphdone(), probedone(), xpt_bus_deregister(), and xpt_bus_register().

Here is the call graph for this function:



7.12.4.40 static void xpt_async_bcast (struct async_list * async_head, u_int32_t async_code, struct cam_path * path, void * async_arg) [static]

Definition at line 4645 of file cam_xpt.c.

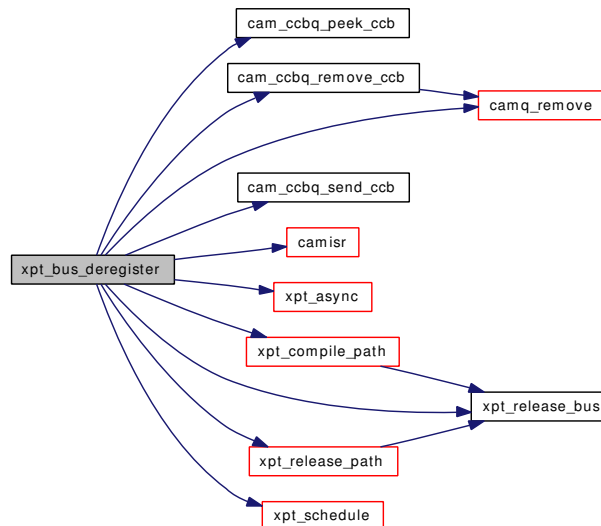
Referenced by xpt_async().

7.12.4.41 int32_t xpt_bus_deregister (path_id_t pathid)

Definition at line 4423 of file cam_xpt.c.

References AC_LOST_DEVICE, AC_PATH_DEREGISTERED, cam_devq::active_dev, cam_devq::alloc_queue, cam_path::bus, cam_ccbq_peek_ccb(), cam_ccbq_remove_ccb(), cam_ccbq_send_ccb(), cam_dead_sim, CAM_LUN_WILDCARD, CAM_REQ_CMP, CAM_TARGET_WILDCARD, camisr(), CAMQ_HEAD, camq_remove(), ccb::ccb_h, ccb_hdr::path, cam_periph::pinfo, cam_pinfo::priority, cam_devq::send_queue, cam_sim::sim_action, xpt_async(), xpt_compile_path(), xpt_release_bus(), xpt_release_path(), and xpt_schedule().

Here is the call graph for this function:



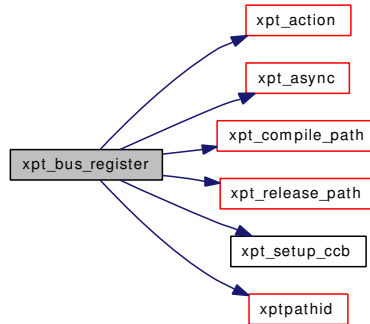
7.12.4.42 int32_t xpt_bus_register (struct cam_sim * sim, u_int32_t bus)

Definition at line 4365 of file cam_xpt.c.

References AC_PATH_REGISTERED, cam_sim::bus_id, CAM_LUN_WILDCARD, CAM_RESRC_UNAVAIL, CAM_SUCCESS, CAM_TARGET_WILDCARD, CAM_XPT_PATH_ID, ccb_pathinq::ccb_h, ccb_hdr::func_code, cam_sim::path_id, cam_sim::sim_name, cam_sim::unit_number, xpt_action(), xpt_async(), xpt_compile_path(), XPT_PATH_INQ, xpt_release_path(), xpt_setup_ccb(), and xptpathid().

Referenced by xpt_init().

Here is the call graph for this function:



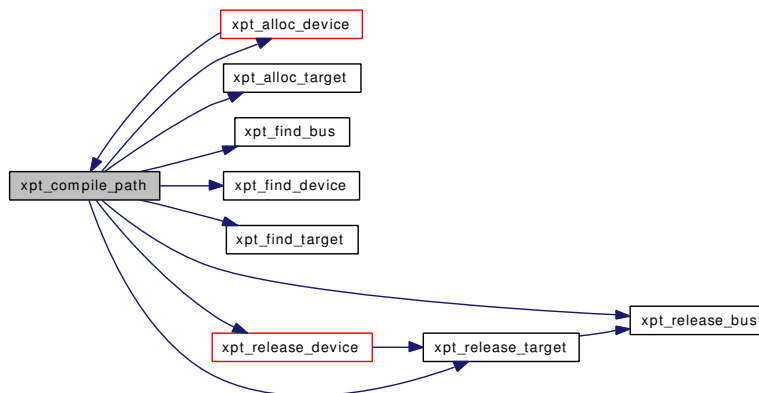
7.12.4.43 static cam_status xpt_compile_path (struct cam_path * new_path, struct cam_periph * perph, path_id_t path_id, target_id_t target_id, lun_id_t lun_id) [static]

Definition at line 4038 of file cam_xpt.c.

References cam_path::bus, CAM_DEBUG, CAM_DEBUG_TRACE, CAM_PATH_INVALID, CAM_REQ_CMP, CAM_RESRC_UNAVAIL, cam_path::device, cam_path::periph, cam_path::target, xpt_alloc_device(), xpt_alloc_target(), xpt_find_bus(), xpt_find_device(), xpt_find_target(), xpt_release_bus(), xpt_release_device(), and xpt_release_target().

Referenced by xpt_alloc_device(), xpt_bus_deregister(), xpt_bus_register(), xpt_create_path(), xpt_dev_async(), xpt_scan_lun(), xptconfigbuscountfunc(), xptsetasynbusfunc(), and xptsetasynfunc().

Here is the call graph for this function:



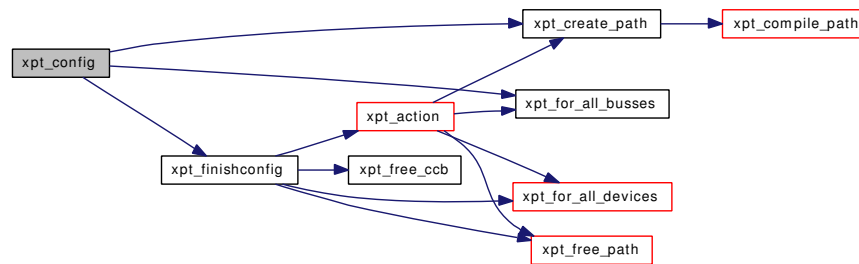
7.12.4.44 static void xpt_config (void * arg) [static]

Definition at line 6944 of file cam_xpt.c.

References CAM_DEBUG_NONE, CAM_REQ_CMP, scsi_delay, xpt_create_path(), xpt_finishconfig(), xpt_for_all_busses(), and xpt_periph.

Referenced by xpt_init().

Here is the call graph for this function:



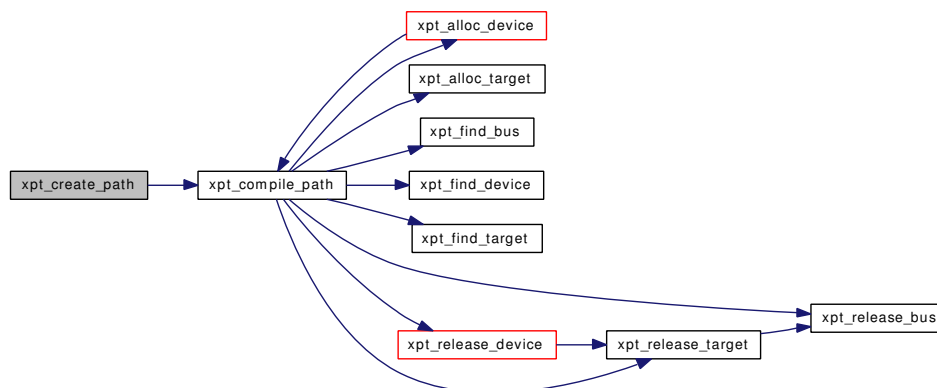
7.12.4.45 cam_status xpt_create_path (struct cam_path ** new_path_ptr, struct cam_periph * periph, path_id_t path_id, target_id_t target_id, lun_id_t lun_id)

Definition at line 4014 of file cam_xpt.c.

References CAM_REQ_CMP, CAM_RESOURCE_UNAVAIL, and xpt_compile_path().

Referenced by cam_periph_alloc(), cam_periph_error(), cdregister(), chinit(), dainit(), passinit(), ptinit(), sainit(), sesinit(), STAILQ_HEAD(), targbhasync(), targbhinit(), targioctl(), xpt_action(), xpt_config(), xpt_init(), xpt_scan_bus(), xptconfigfunc(), and xptioctl().

Here is the call graph for this function:



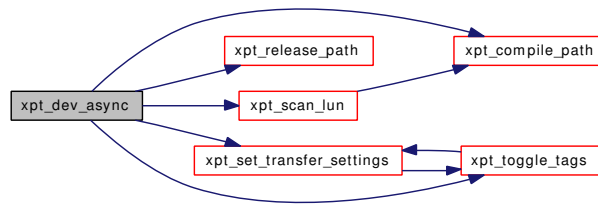
7.12.4.46 `static void xpt_dev_async (u_int32_t async_code, struct cam_cb * bus, struct cam_et * target, struct cam_ed * device, void * async_arg)` [static]

Definition at line 4672 of file `cam_xpt.c`.

References `AC_BUS_RESET`, `AC_INQ_CHANGED`, `AC_LOST_DEVICE`, `AC_SENT_BDR`, `AC_TRANSFER_NEG`, `cam_path::bus`, `CAM_DEV_UNCONFIGURED`, `CAM_EXPECT_INQ_CHANGE`, `CAM_LUN_WILDCARD`, `CAM_REQ_CMP`, `CAM_REQ_CMP_ERR`, `CAM_TARGET_WILDCARD`, `cam_path::device`, `cam_path::target`, `xpt_compile_path()`, `xpt_release_path()`, `xpt_scan_lun()`, `xpt_set_transfer_settings()`, and `xpt_toggle_tags()`.

Referenced by `xpt_async()`.

Here is the call graph for this function:



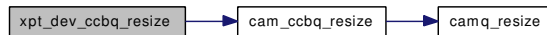
7.12.4.47 `static u_int32_t xpt_dev_ccbq_resize (struct cam_path * path, int newopenings)` [static]

Definition at line 5194 of file `cam_xpt.c`.

References `cam_ccbq_resize()`, `CAM_DEV_RESIZE_QUEUE_NEEDED`, `CAM_DEV_TAG_AFTER_COUNT`, `CAM_REQ_CMP`, `cam_path::device`, and `SID_CmdQue`.

Referenced by `xpt_action()`, `xpt_set_transfer_settings()`, and `xpt_start_tags()`.

Here is the call graph for this function:



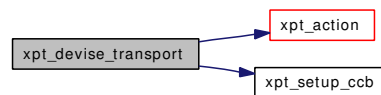
7.12.4.48 `static void xpt_devise_transport (struct cam_path * path)` [static]

Definition at line 6432 of file `cam_xpt.c`.

References `CAM_DEV_INQUIRY_DATA_VALID`, `ccb_trans_settings::ccb_h`, `ccb_pathinq::ccb_h`, `CTS_TYPE_CURRENT_SETTINGS`, `cam_path::device`, `ccb_hdr::func_code`, `PROTO_SCSI`, `ccb_trans_settings::proto_specific`, `ccb_trans_settings::protocol`, `ccb_trans_settings::protocol_version`, `SID_ANSI_REV`, `SID_SPI_MASK`, `scsi_inquiry_data::spi3data`, `cam_path::target`, `ccb_trans_settings::transport`, `ccb_trans_settings::transport_version`, `ccb_trans_settings::type`, `ccb_trans_settings::valid`, `ccb_trans_settings::xport_specific`, `XPORT_SPI`, `xpt_action()`, `XPT_PATH_INQ`, `XPT_SET_TRAN_SETTINGS`, and `xpt_setup_ccb()`.

Referenced by `probedone()`, and `xpt_alloc_device()`.

Here is the call graph for this function:



7.12.4.49 void xpt_done (union ccb * done_ccb)

Definition at line 4899 of file cam_xpt.c.

References CAM_DEBUG, CAM_DEBUG_TRACE, CAM_DONEQ_INDEX, CAM_PERIPH_BIO, ccb::ccb_h, ccb_hdr::func_code, cam_pinfo::index, ccb_hdr::path, cam_path::periph, ccb_hdr::pinfo, cam_periph::type, and XPT_FC_QUEUED.

Referenced by dead_sim_action(), probedone(), xpt_action(), xpt_scan_bus(), xpt_scan_lun(), xpt_set_transfer_settings(), and xptaction().

7.12.4.50 static struct cam_eb * xpt_find_bus (path_id_t path_id) [static]

Definition at line 5219 of file cam_xpt.c.

Referenced by xpt_compile_path(), and xpt_release_simq().

7.12.4.51 static struct cam_ed * xpt_find_device (struct cam_et * target, lun_id_t lun_id) [static]

Definition at line 5251 of file cam_xpt.c.

Referenced by xpt_compile_path().

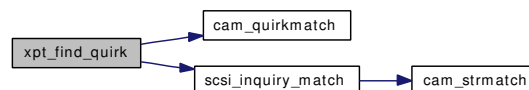
7.12.4.52 static void xpt_find_quirk (struct cam_ed * device) [static]

Definition at line 6398 of file cam_xpt.c.

References cam_quirkmatch(), and scsi_inquiry_match().

Referenced by probedone().

Here is the call graph for this function:



7.12.4.53 static struct cam_et * xpt_find_target (struct cam_eb * bus, target_id_t target_id) [static]

Definition at line 5235 of file cam_xpt.c.

Referenced by xpt_compile_path().

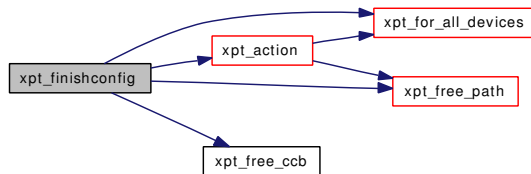
7.12.4.54 `static void xpt_finishconfig (struct cam_periph * periph, union ccb * ccb)` [static]

Definition at line 7014 of file `cam_xpt.c`.

References `CAM_DEBUG`, `CAM_DEBUG_TRACE`, `CAM_REQ_CMP`, `ccb_hdr::cbfcnp`, `ccb::ccb_h`, `ccb::crn`, `ccb_rescan::flags`, `ccb_hdr::func_code`, `periph_driver::init`, `ccb_hdr::path`, `periph_drivers`, `ccb_hdr::status`, `xpt_action()`, `xpt_for_all_devices()`, `xpt_free_ccb()`, `xpt_free_path()`, `XPT_RESET_BUS`, and `XPT_SCAN_BUS`.

Referenced by `xpt_config()`, and `xptconfigfunc()`.

Here is the call graph for this function:



7.12.4.55 `static int xpt_for_all_busses (xpt_busfunc_t * tr_func, void * arg)` [static]

Definition at line 2844 of file `cam_xpt.c`.

References `xpt_traverse_config::depth`, `xpt_traverse_config::tr_arg`, `xpt_traverse_config::tr_func`, `XPT_DEPTH_BUS`, and `xptbustraverse()`.

Referenced by `xpt_action()`, and `xpt_config()`.

Here is the call graph for this function:



7.12.4.56 `static int xpt_for_all_devices (xpt_devicefunc_t * tr_func, void * arg)` [static]

Definition at line 2876 of file `cam_xpt.c`.

References `xpt_traverse_config::depth`, `xpt_traverse_config::tr_arg`, `xpt_traverse_config::tr_func`, `XPT_DEPTH_DEVICE`, and `xptbustraverse()`.

Referenced by `xpt_action()`, and `xpt_finishconfig()`.

Here is the call graph for this function:



7.12.4.57 `void xpt_free_ccb (union ccb * free_ccb)`

Definition at line 4951 of file `cam_xpt.c`.

Referenced by `passioctl()`, `TAILQ_HEAD()`, `targbhdone()`, `xpt_finishconfig()`, `xpt_release_ccb()`, `xpt_rescan()`, `xpt_scan_bus()`, `xptconfigfunc()`, and `xptioctl()`.

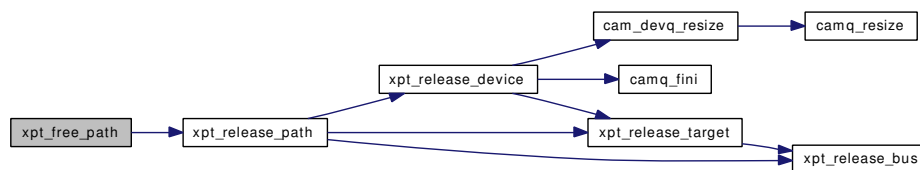
7.12.4.58 `void xpt_free_path (struct cam_path * path)`

Definition at line 4130 of file `cam_xpt.c`.

References `CAM_DEBUG`, `CAM_DEBUG_TRACE`, and `xpt_release_path()`.

Referenced by `cam_periph_alloc()`, `cam_periph_error()`, `camperiphfree()`, `cdregister()`, `chinit()`, `dainit()`, `passinit()`, `ptinit()`, `sainit()`, `sesinit()`, `STAILQ_HEAD()`, `TAILQ_HEAD()`, `targbhasync()`, `targbhinit()`, `targioctl()`, `xpt_action()`, `xpt_finishconfig()`, `xpt_init()`, `xpt_rescan()`, `xpt_scan_bus()`, and `xptioctl()`.

Here is the call graph for this function:



7.12.4.59 `u_int32_t xpt_freeze_devq (struct cam_path * path, u_int count)`

Definition at line 4733 of file `cam_xpt.c`.

References `CAM_REQ_INPROG`, `CAM_QUEUE_REQ`, `cam_path::device`, `ccb_hdr::path`, and `ccb_hdr::status`.

Referenced by `xpt_action()`, `xpt_set_transfer_settings()`, and `xpt_start_tags()`.

7.12.4.60 `u_int32_t xpt_freeze_simq (struct cam_sim * sim, u_int count)`

Definition at line 4764 of file `cam_xpt.c`.

References `cam_devq::active_dev`, `CAM_REQ_INPROG`, `CAM_QUEUE_REQ`, `cam_sim::devq`, `camq::qfrozen_cnt`, `cam_devq::send_queue`, and `ccb_hdr::status`.

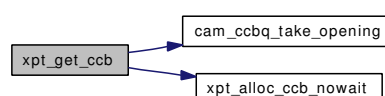
7.12.4.61 `static union ccb * xpt_get_ccb (struct cam_ed * device)` `[static]`

Definition at line 4968 of file `cam_xpt.c`.

References `cam_ccbq_take_opening()`, `ccb::ccb_h`, `ccb_hdr::timeout_ch`, and `xpt_alloc_ccb_nowait()`.

Referenced by `xpt_run_dev_allocq()`, and `xpt_schedule()`.

Here is the call graph for this function:



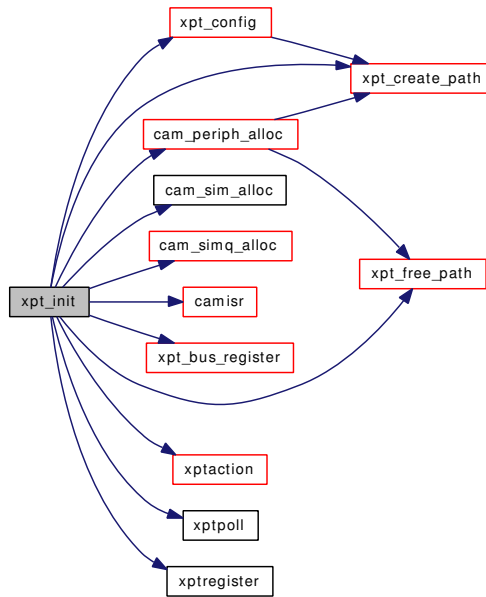
7.12.4.62 `static void xpt_init (void *) [static]`

Definition at line 1459 of file `cam_xpt.c`.

References `CAM_LUN_WILDCARD`, `cam_periph_alloc()`, `CAM_PERIPH_BIO`, `CAM_REQ_CMP`, `cam_sim_alloc()`, `cam_simq_alloc()`, `CAM_SUCCESS`, `CAM_TARGET_WILDCARD`, `CAM_XPT_PATH_ID`, `camisr()`, `cam_sim::softc`, `xpt_bus_register()`, `xpt_config()`, `xpt_create_path()`, `xpt_free_path()`, `xpt_periph`, `xptaction()`, `xptpoll()`, and `xptregister()`.

Referenced by `cam_module_event_handler()`.

Here is the call graph for this function:



7.12.4.63 `void xpt_merge_ccb (union ccb * master_ccb, union ccb * slave_ccb)`

Definition at line 3973 of file `cam_xpt.c`.

References `ccb::ccb_h`, `ccb_hdr::flags`, `ccb_hdr::func_code`, `ccb_hdr::retry_count`, and `ccb_hdr::timeout`.

Referenced by `passsendccb()`, and `xptioctl()`.

7.12.4.64 `int xpt_path_comp (struct cam_path * path1, struct cam_path * path2)`

Definition at line 4145 of file `cam_xpt.c`.

References `cam_path::bus`, `CAM_BUS_WILDCARD`, `CAM_LUN_WILDCARD`, `CAM_TARGET_WILDCARD`, `cam_path::device`, and `cam_path::target`.

Referenced by `cam_periph_find()`, and `xpt_rescan()`.

7.12.4.65 `lun_id_t xpt_path_lun_id (struct cam_path * path)`

Definition at line 4283 of file `cam_xpt.c`.

References CAM_LUN_WILDCARD, and cam_path::device.

Referenced by cam_periph_alloc().

7.12.4.66 **path_id_t** xpt_path_path_id (struct cam_path * path)

Definition at line 4264 of file cam_xpt.c.

References cam_path::bus.

Referenced by cam_periph_alloc(), cam_periph_error(), and targbhasync().

7.12.4.67 **struct cam_periph*** xpt_path_periph (struct cam_path * path)

Definition at line 4302 of file cam_xpt.c.

References cam_periph::path, and cam_path::periph.

Referenced by cam_periph_error(), camperiphscsisenseerror(), cd6byteworkaround(), cderror(), cdruncceb(), cherror(), cmd6workaround(), daerror(), passerror(), pterror(), saerror(), and seserror().

7.12.4.68 **struct cam_sim*** xpt_path_sim (struct cam_path * path)

Definition at line 4294 of file cam_xpt.c.

References cam_path::bus.

7.12.4.69 **int** xpt_path_string (struct cam_path * path, char * str, size_t str_len)

Definition at line 4224 of file cam_xpt.c.

References cam_path::bus, cam_path::device, cam_path::periph, cam_periph::periph_name, cam_path::target, and cam_periph::unit_number.

Referenced by cam_error_string(), and scsi_sense_sbuf().

7.12.4.70 **target_id_t** xpt_path_target_id (struct cam_path * path)

Definition at line 4272 of file cam_xpt.c.

References CAM_TARGET_WILDCARD, and cam_path::target.

Referenced by cam_periph_alloc(), and cam_periph_error().

7.12.4.71 **static void** xpt_periph_init () [static]

Definition at line 978 of file cam_xpt.c.

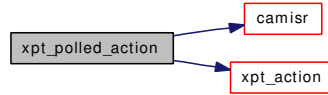
7.12.4.72 **void** xpt_polled_action (union ccb * start_ccb)

Definition at line 3596 of file cam_xpt.c.

References cam_path::bus, CAM_CMD_TIMEOUT, CAM_REQ_INPROG, CAM_RESRC_UNAVAIL, CAM_STATUS_MASK, camisr(), ccb::ccb_h, cam_path::device, cam_sim::devq, ccb_hdr::path, cam_devq::send_openings, cam_sim::sim_poll, ccb_hdr::status, ccb_hdr::timeout, and xpt_action().

Referenced by `dadump()`, and `dashutdown()`.

Here is the call graph for this function:



7.12.4.73 void `xpt_print` (struct `cam_path` * `path`, const char * `fmt`, ...)

Definition at line 4214 of file `cam_xpt.c`.

References `xpt_print_path()`.

Referenced by `abort_all_pending()`, `cam_periph_error()`, `camperiphdone()`, `camperiphnextunit()`, `camperiphscsisstatuserror()`, `cd6byteworkaround()`, `cdcleanup()`, `cddone()`, `cdgetmode()`, `cdoninvalidate()`, `cdreportkey()`, `cdstart()`, `chcleanup()`, `chdone()`, `chgetelemstatus()`, `chgetparams()`, `choninvalidate()`, `cmd6workaround()`, `dacleanup()`, `daclose()`, `dadone()`, `dadump()`, `daoninvalidate()`, `dashutdown()`, `passcleanup()`, `passioctl()`, `passoninvalidate()`, `passopen()`, `probedone()`, `proberequestbackoff()`, `probestart()`, `ptdone()`, `ptdtor()`, `ptoninvalidate()`, `sacleanup()`, `saclose()`, `saerror()`, `saiioctl()`, `samount()`, `saoninvalidate()`, `saregister()`, `sasetparams()`, `sastart()`, `sastrategy()`, `sescleanup()`, `sesoninvalidate()`, `targbhenlun()`, `targendislun()`, `targreturnccb()`, `targstart()`, `xpt_action()`, `xpt_rescan()`, `xpt_scan_lun()`, and `xpt_set_transfer_settings()`.

Here is the call graph for this function:



7.12.4.74 void `xpt_print_path` (struct `cam_path` * `path`)

Definition at line 4181 of file `cam_xpt.c`.

References `cam_path::bus`, `cam_path::device`, `cam_path::periph`, `cam_periph::periph_name`, `cam_path::target`, and `cam_periph::unit_number`.

Referenced by `cam_periph_error()`, `camperiphscsisenseerror()`, `sagetparams()`, `sasetparams()`, `sastrategy()`, and `xpt_print()`.

7.12.4.75 static void `xpt_release_bus` (struct `cam_eb` * `bus`) [static]

Definition at line 4992 of file `cam_xpt.c`.

Referenced by `xpt_bus_deregister()`, `xpt_compile_path()`, `xpt_release_path()`, `xpt_release_simq()`, and `xpt_release_target()`.

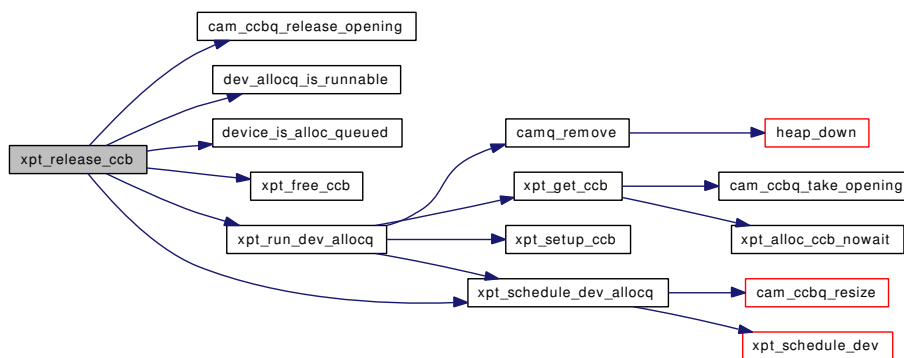
7.12.4.76 void `xpt_release_ccb` (union `ccb` * `free_ccb`)

Definition at line 4316 of file `cam_xpt.c`.

References `cam_path::bus`, `cam_ccbq_release_opening()`, `CAM_DEBUG_PRINT`, `CAM_DEBUG_XPT`, `ccb::ccb_h`, `dev_allocq_is_runnable()`, `cam_path::device`, `device_is_alloc_queued()`, `ccb_hdr::path`, `xpt_free_ccb()`, `xpt_run_dev_allocq()`, and `xpt_schedule_dev_allocq()`.

Referenced by `cam_periph_ioctl()`, `cddone()`, `cdgetmode()`, `cdpause()`, `cdplay()`, `cdplaymsf()`, `cdplaytracks()`, `cdprevent()`, `cdreaddvdstructure()`, `cdreadsubchannel()`, `cdreadtoc()`, `cdreportkey()`, `cdsendkey()`, `cdsetmode()`, `cdsetspeed()`, `cdsize()`, `cdstartunit()`, `cdstopunit()`, `chdone()`, `chexchange()`, `chgetelemstatus()`, `chgetparams()`, `chielem()`, `chmove()`, `chposition()`, `daclose()`, `dadone()`, `dagetcapacity()`, `daprevent()`, `dastart()`, `passdone()`, `passioctl()`, `probedone()`, `ptdone()`, `ptstart()`, `sadone()`, `saerase()`, `sagetparams()`, `saloadunload()`, `samount()`, `saprevent()`, `sardpos()`, `sareservereleaseunit()`, `saretension()`, `sarewind()`, `sasetparams()`, `sasetpos()`, `saspace()`, `sastart()`, `sawritefilemarks()`, `ses_runcmd()`, `targbhdone()`, `targbhstart()`, `targfreeccb()`, and `targstart()`.

Here is the call graph for this function:



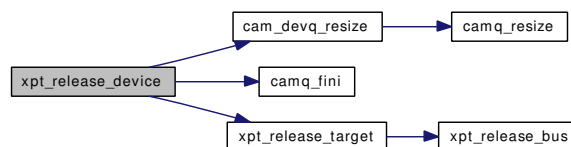
7.12.4.77 static void xpt_release_device (struct `camEb` * *bus*, struct `camEt` * *target*, struct `camEd` * *device*) [static]

Definition at line 5158 of file `cam_xpt.c`.

References `cam_devq::alloc_queue`, `camq::array_size`, `CAM_DEV_REL_TIMEOUT_PENDING`, `CAM_DEV_UNCONFIGURED`, `cam_devq_resize()`, `CAM_UNQUEUEUED_INDEX`, `camq_fini()`, `SIM_DEAD`, and `xpt_release_target()`.

Referenced by `xpt_compile_path()`, and `xpt_release_path()`.

Here is the call graph for this function:



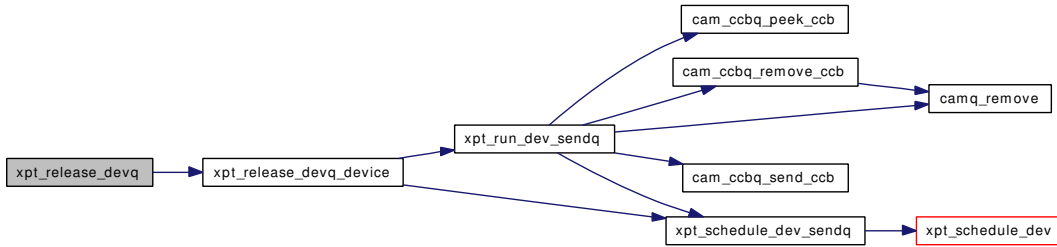
7.12.4.78 void xpt_release_devq (struct `camPath` * *path*, u_int *count*, int *run_queue*)

Definition at line 4791 of file `cam_xpt.c`.

References `cam_path::device`, and `xpt_release_devq_device()`.

Referenced by `camisr()`, `probedone()`, and `xpt_action()`.

Here is the call graph for this function:



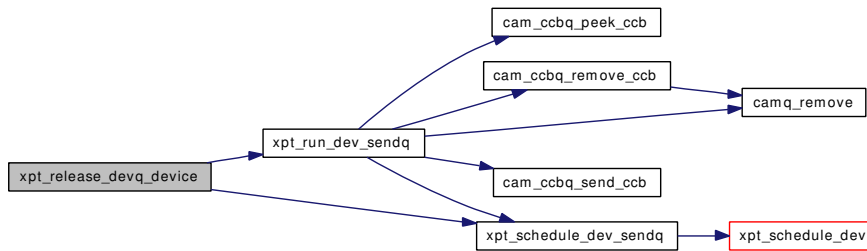
7.12.4.79 `static void xpt_release_devq_device (struct cam_ed * dev, u_int count, int run_queue)` `[static]`

Definition at line 4799 of file `cam_xpt.c`.

References `CAM_DEV_REL_ON_COMPLETE`, `CAM_DEV_REL_TIMEOUT_PENDING`, `xpt_run_dev_sendq()`, and `xpt_schedule_dev_sendq()`.

Referenced by `xpt_release_devq()`, and `xpt_release_devq_timeout()`.

Here is the call graph for this function:

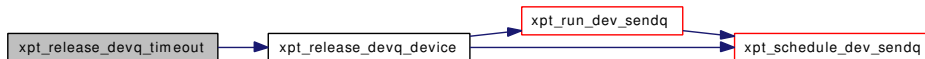


7.12.4.80 `static void xpt_release_devq_timeout (void * arg)` `[static]`

Definition at line 4781 of file `cam_xpt.c`.

References `xpt_release_devq_device()`.

Here is the call graph for this function:



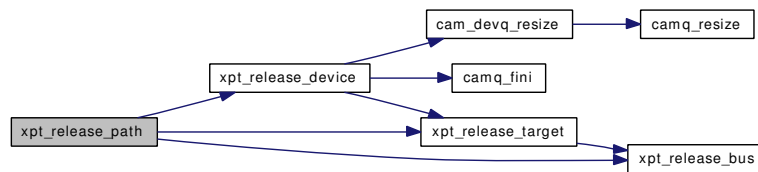
7.12.4.81 `static void xpt_release_path (struct cam_path * path)` `[static]`

Definition at line 4112 of file `cam_xpt.c`.

References `cam_path::bus`, `CAM_DEBUG`, `CAM_DEBUG_TRACE`, `cam_path::device`, `cam_path::target`, `xpt_release_bus()`, `xpt_release_device()`, and `xpt_release_target()`.

Referenced by `xpt_alloc_device()`, `xpt_bus_deregister()`, `xpt_bus_register()`, `xpt_dev_async()`, `xpt_free_path()`, `xptconfigbuscountfunc()`, `xptscandone()`, `xptsetasynbusfunc()`, and `xptsetasynfunc()`.

Here is the call graph for this function:



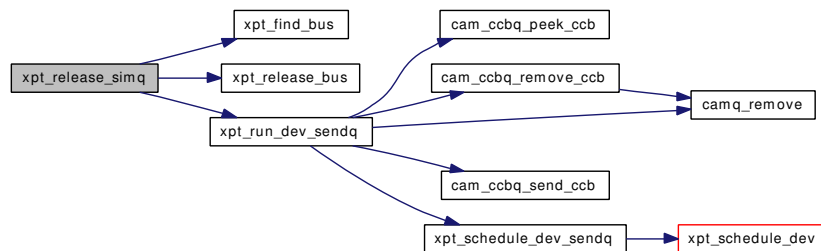
7.12.4.82 void xpt_release_simq (struct `cam_sim` * *sim*, int *run_queue*)

Definition at line 4848 of file `cam_xpt.c`.

References `cam_sim::c_handle`, `CAM_SIM_REL_TIMEOUT_PENDING`, `cam_sim::devq`, `cam_sim::flags`, `cam_sim::path_id`, `camq::qfrozen_cnt`, `cam_devq::send_queue`, `xpt_find_bus()`, `xpt_release_bus()`, and `xpt_run_dev_sendq()`.

Referenced by `camisr()`, and `xpt_release_simq_timeout()`.

Here is the call graph for this function:

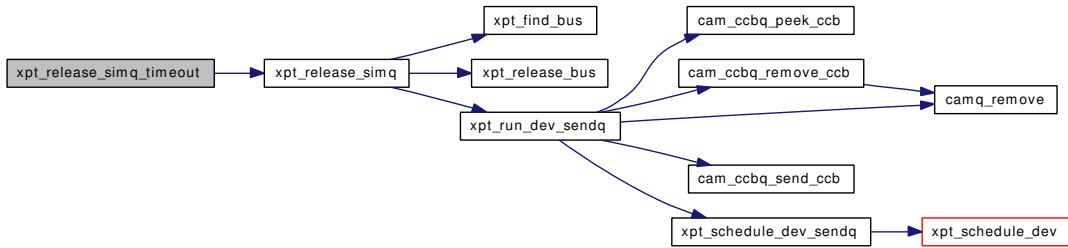


7.12.4.83 static void xpt_release_simq_timeout (void * *arg*) [static]

Definition at line 4890 of file `cam_xpt.c`.

References `xpt_release_simq()`.

Here is the call graph for this function:



7.12.4.84 static void xpt_release_target (struct cam_eb * bus, struct cam_et * target) [static]

Definition at line 5044 of file cam_xpt.c.

References xpt_release_bus().

Referenced by xpt_compile_path(), xpt_release_device(), and xpt_release_path().

Here is the call graph for this function:



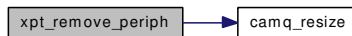
7.12.4.85 void xpt_remove_periph (struct cam_periph * periph)

Definition at line 1598 of file cam_xpt.c.

References camq_resize(), cam_path::device, xpt_softc::generation, cam_periph::path, and xsoftc.

Referenced by cam_periph_alloc(), and camperiphfree().

Here is the call graph for this function:

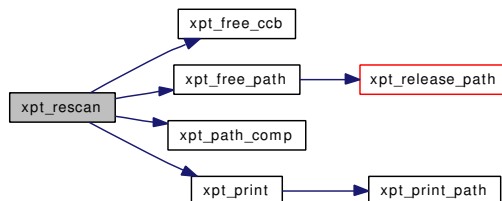


7.12.4.86 void xpt_rescan (union ccb * ccb)

Definition at line 1438 of file cam_xpt.c.

References ccb::ccb_h, ccb_hdr::path, ccb_hdr::sim_links, xpt_free_ccb(), xpt_free_path(), xpt_path_comp(), and xpt_print().

Here is the call graph for this function:



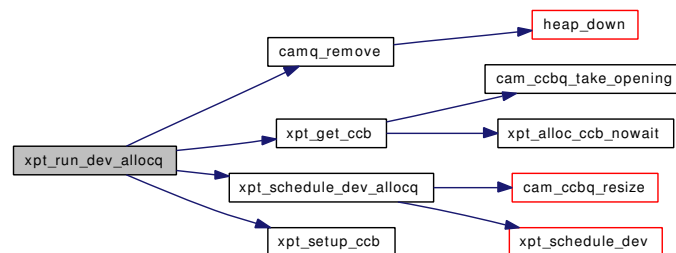
7.12.4.87 `static void xpt_run_dev_allocq (struct camEb * bus) [static]`

Definition at line 3758 of file cam_xpt.c.

References `cam_devq::alloc_active`, `cam_devq::alloc_openings`, `cam_devq::alloc_queue`, `CAM_DEBUG_PRINT`, `CAM_DEBUG_XPT`, `CAMQ_HEAD`, `camq_remove()`, `ccb::ccb_h`, `camq::entries`, `cam_periph::path`, `cam_periph::periph_start`, `cam_periph::pinfo`, `cam_pinfo::priority`, `camq::qfrozen_cnt`, `xpt_get_ccb()`, `xpt_schedule_dev_allocq()`, and `xpt_setup_ccb()`.

Referenced by `xpt_release_ccb()`, and `xpt_schedule()`.

Here is the call graph for this function:

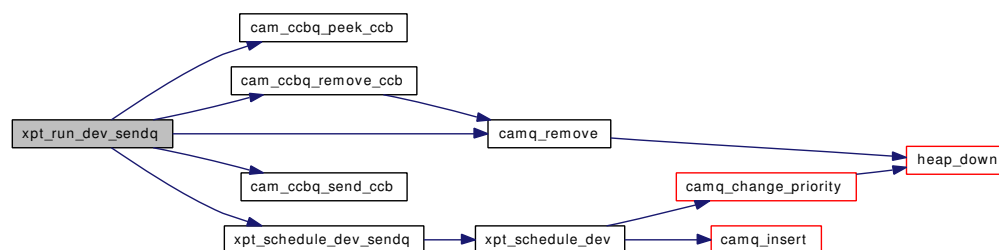
**7.12.4.88** `static void xpt_run_dev_sendq (struct camEb * bus) [static]`

Definition at line 3836 of file cam_xpt.c.

References `cam_devq::active_dev`, `cam_path::bus`, `cam_ccbq_peek_ccb()`, `cam_ccbq_remove_ccb()`, `cam_ccbq_send_ccb()`, `CAM_DEBUG_PRINT`, `CAM_DEBUG_XPT`, `CAM_DEV_QFREEZE`, `CAM_HIGH_POWER`, `CAM_TAG_ACTION_NONE`, `CAM_TAG_ACTION_VALID`, `CAMQ_HEAD`, `camq_remove()`, `ccb::ccb_h`, `ccb::csio`, `cam_sim::devq`, `camq::entries`, `ccb_hdr::flags`, `ccb_hdr::func_code`, `ccb_hdr::path`, `camq::qfrozen_cnt`, `cam_devq::send_active`, `cam_devq::send_openings`, `cam_devq::send_queue`, `SID_CmdQue`, `cam_sim::sim_action`, `ccb_scsiio::tag_action`, `xpt_schedule_dev_sendq()`, and `XPT_SCSI_IO`.

Referenced by `camisr()`, `xpt_action()`, `xpt_release_devq_device()`, and `xpt_release_simq()`.

Here is the call graph for this function:

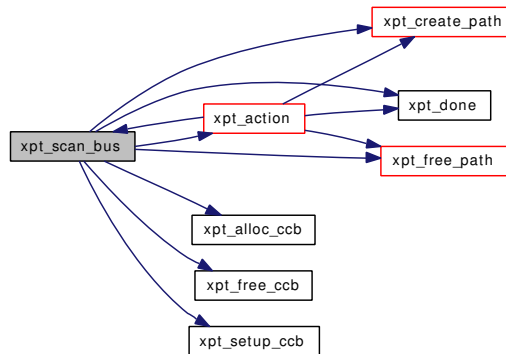
**7.12.4.89** `static void xpt_scan_bus (struct camPeriph * periph, union ccb * ccb) [static]`

Definition at line 5278 of file cam_xpt.c.

References CAM_DEBUG, CAM_DEBUG_TRACE, CAM_QUIRK_NOLUNS, CAM_REQ_CMP, CAM_SCSI2_MAXLUN, CAN_SRCH_HI_DENSE, CAN_SRCH_HI_SPARSE, ccb_hdr::cbfcnp, ccb::ccb_h, xpt_scan_bus_info::counter, xpt_scan_bus_info::cpi, ccb::cpi, ccb::crcn, cam_path::device, ccb_rescan::flags, ccb_hdr::func_code, ccb_pathinq::hba_misc, ccb_pathinq::initiator_id, ccb_pathinq::max_lun, ccb_pathinq::max_target, ccb_hdr::path, ccb_hdr::path_id, PIM_NOINITIATOR, PIM_SEQSCAN, ccb_hdr::pinfo, cam_pinfo::priority, xpt_scan_bus_info::request_ccb, ccb_hdr::status, cam_path::target, ccb_hdr::target_id, ccb_hdr::target_lun, xpt_action(), xpt_alloc_ccb(), xpt_create_path(), xpt_done(), xpt_free_ccb(), xpt_free_path(), XPT_GDEV_TYPE, XPT_PATH_INQ, xpt_periph, XPT_SCAN_BUS, XPT_SCAN_LUN, and xpt_setup_ccb().

Referenced by xpt_action().

Here is the call graph for this function:



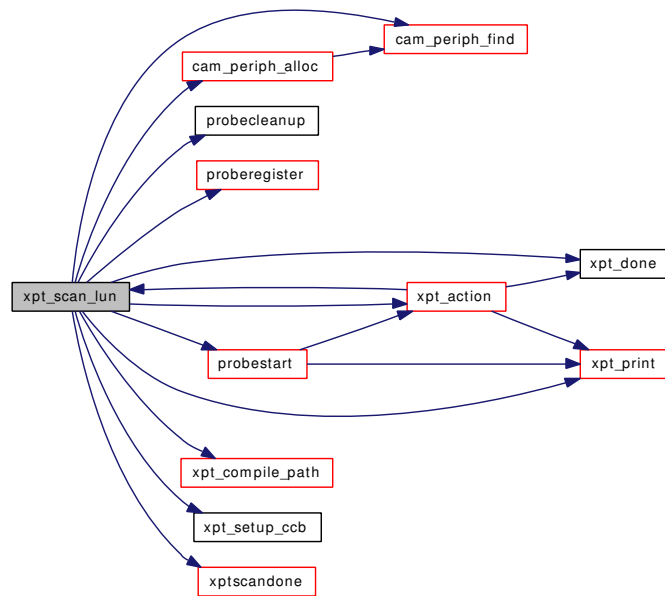
7.12.4.90 `static void xpt_scan_lun (struct cam_periph * periph, struct cam_path * path, cam_flags flags, union ccb * ccb)` [static]

Definition at line 5552 of file `cam_xpt.c`.

References `cam_path::bus`, CAM_DEBUG, CAM_DEBUG_TRACE, `cam_periph_alloc()`, CAM_PERIPH_BIO, `cam_periph_find()`, CAM_REQ_CMP, `ccb_hdr::cbfcnp`, `ccb_pathinq::ccb_h`, `ccb::ccb_h`, `ccb::crcn`, `cam_path::device`, `ccb_rescan::flags`, `ccb_hdr::func_code`, `cam_periph::path`, `ccb_hdr::path`, PIM_NOINITIATOR, `probecleanup()`, `proberegister()`, `probestart()`, `cam_periph::softc`, `ccb_hdr::status`, `cam_path::target`, `xpt_action()`, `xpt_compile_path()`, `xpt_done()`, XPT_PATH_INQ, `xpt_periph`, `xpt_print()`, XPT_SCAN_LUN, `xpt_setup_ccb()`, and `xptscandone()`.

Referenced by `xpt_action()`, and `xpt_dev_async()`.

Here is the call graph for this function:



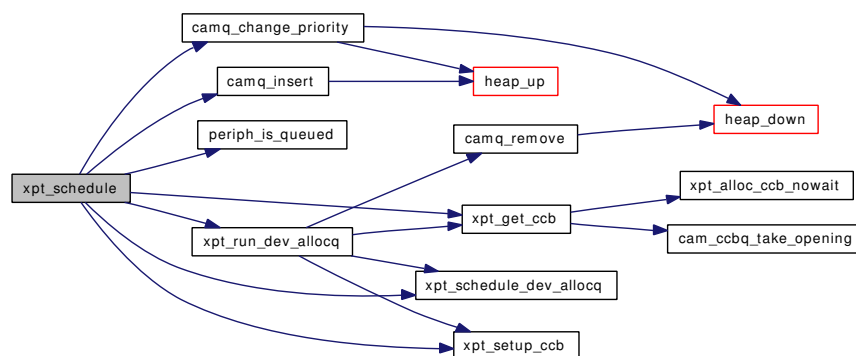
7.12.4.91 void xpt_schedule (struct cam_periph * perph, u_int32_t new_priority)

Definition at line 3660 of file cam_xpt.c.

References cam_path::bus, CAM_DEBUG, CAM_DEBUG_SUBTRACE, CAM_DEBUG_TRACE, camq_change_priority(), camq_insert(), ccb::ccb_h, cam_path::device, cam_pinfo::generation, cam_pinfo::index, cam_periph::path, periph_is_queued(), cam_periph::periph_start, cam_periph::pinfo, cam_pinfo::priority, SIM_DEAD, xpt_get_ccb(), xpt_run_dev_allocq(), xpt_schedule_dev_allocq(), and xpt_setup_ccb().

Referenced by cam_periph_getccb(), cdregister(), cdrunchangerqueue(), cdschedule(), cdstart(), cdstrategy(), chregister(), dadone(), daregister(), dastart(), dastrategy(), probedone(), probeschedule(), ptstart(), ptstrategy(), sastart(), sastrategy(), targbhdone(), targbhstart(), targstart(), targwrite(), and xpt_bus_deregister().

Here is the call graph for this function:



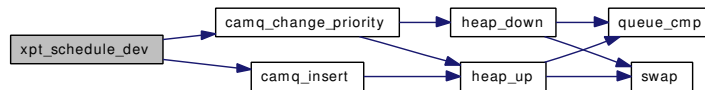
7.12.4.92 `static int xpt_schedule_dev (struct camq * queue, cam_pinfo * dev_pinfo, u_int32_t new_priority) [static]`

Definition at line 3720 of file cam_xpt.c.

References CAM_DEBUG_PRINT, CAM_DEBUG_XPT, CAM_UNQUEUED_INDEX, camq_change_priority(), camq_insert(), camq::generation, cam_pinfo::generation, cam_pinfo::index, and cam_pinfo::priority.

Referenced by xpt_schedule_dev_allocq(), and xpt_schedule_dev_sendq().

Here is the call graph for this function:



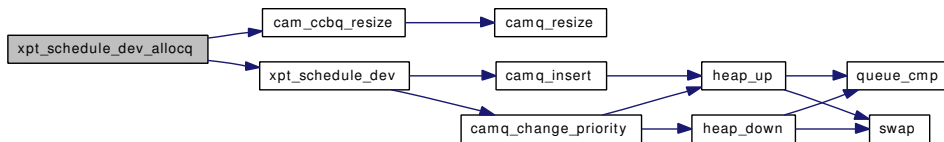
7.12.4.93 `static __inline int xpt_schedule_dev_allocq (struct cam_eb * bus, struct cam_ed * dev) [static]`

Definition at line 899 of file cam_xpt.c.

References cam_ccbq_resize(), CAM_DEV_RESIZE_QUEUE_NEEDED, CAMQ_GET_HEAD, and xpt_schedule_dev().

Referenced by xpt_release_ccb(), xpt_run_dev_allocq(), and xpt_schedule().

Here is the call graph for this function:



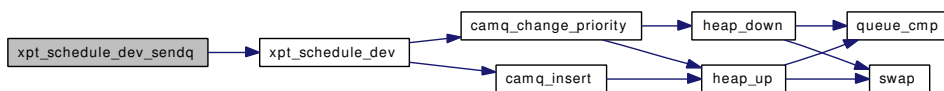
7.12.4.94 `static __inline int xpt_schedule_dev_sendq (struct cam_eb * bus, struct cam_ed * dev) [static]`

Definition at line 926 of file cam_xpt.c.

References CAMQ_GET_HEAD, and xpt_schedule_dev().

Referenced by camisr(), xpt_action(), xpt_release_devq_device(), and xpt_run_dev_sendq().

Here is the call graph for this function:



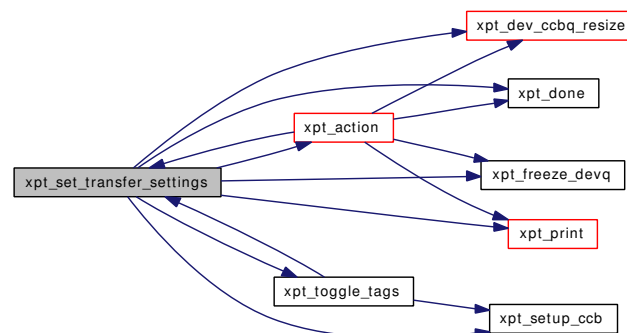
7.12.4.95 static void xpt_set_transfer_settings (struct **ccb_trans_settings** * *cts*, struct **cam_ed** * *device*, int *async_update*) [static]

Definition at line 6512 of file cam_xpt.c.

References `cam_path::bus`, `ccb_trans_settings_spi::bus_width`, `CAM_DEV_INQUIRY_DATA_VALID`, `CAM_DEV_TAG_AFTER_COUNT`, `CAM_PATH_INVALID`, `CAM_REQ_CMP`, `CAM_STATUS_MASK`, `CAM_TAG_DELAY_COUNT`, `ccb_relsim::ccb_h`, `ccb::ccb_h`, `ccb_pathinq::ccb_h`, `ccb_trans_settings::ccb_h`, `ccb::cpi`, `ccb::cts`, `CTS_SCSI_FLAGS_TAG_ENB`, `CTS_SCSI_VALID_TQ`, `CTS_SPI_FLAGS_DISC_ENB`, `CTS_SPI_VALID_BUS_WIDTH`, `CTS_SPI_VALID_DISC`, `CTS_SPI_VALID_PPR_OPTIONS`, `CTS_SPI_VALID_SYNC_OFFSET`, `CTS_SPI_VALID_SYNC_RATE`, `CTS_TYPE_CURRENT_SETTINGS`, `CTS_TYPE_USER_SETTINGS`, `scsi_inquiry_data::device`, `scsi_inquiry_data::flags`, `ccb_trans_settings_spi::flags`, `ccb_trans_settings_scsi::flags`, `ccb_hdr::func_code`, `ccb_pathinq::hba_inquiry`, `INQ_DATA_TQ_ENABLED`, `cam_sim::max_dev_openings`, `MSG_EXT_PPR_DT_REQ`, `MSG_EXT_PPR_IU_REQ`, `MSG_EXT_PPR_QAS_REQ`, `MSG_EXT_WDTR_BUS_16_BIT`, `MSG_EXT_WDTR_BUS_32_BIT`, `MSG_EXT_WDTR_BUS_8_BIT`, `ccb_relsim::openings`, `ccb_hdr::path`, `PI_SDTR_ABLE`, `PI_TAG_ABLE`, `PI_WIDE_16`, `PI_WIDE_32`, `ccb_trans_settings_spi::ppr_options`, `PROTO_SCSI`, `ccb_trans_settings::proto_specific`, `PROTO_UNKNOWN`, `PROTO_UNSPECIFIED`, `PROTO_VERSION_UNKNOWN`, `PROTO_VERSION_UNSPECIFIED`, `ccb_trans_settings::protocol`, `ccb_trans_settings::protocol_version`, `ccb_relsim::qfrozen_cnt`, `ccb_relsim::release_flags`, `ccb_relsim::release_timeout`, `RELSIM_RELEASE_AFTER_QEMPTY`, `SCP_QUEUE_DQUE`, `ccb_trans_settings::scsi`, `SID_CmdQue`, `SID_SPI_CLOCK_DT`, `SID_SPI_IUS`, `SID_SPI_QAS`, `SID_Sync`, `SID_WBus16`, `SID_WBus32`, `cam_sim::sim_action`, `ccb_trans_settings::spi`, `scsi_inquiry_data::spi3data`, `ccb_hdr::status`, `ccb_trans_settings_spi::sync_offset`, `ccb_trans_settings_spi::sync_period`, `ccb_trans_settings::transport`, `ccb_trans_settings::transport_version`, `ccb_trans_settings::type`, `ccb_trans_settings_spi::valid`, `ccb_trans_settings_scsi::valid`, `ccb_trans_settings::xport_specific`, `XPORT_SPI`, `XPORT_UNKNOWN`, `XPORT_UNSPECIFIED`, `XPORT_VERSION_UNKNOWN`, `XPORT_VERSION_UNSPECIFIED`, `xpt_action()`, `xpt_dev_ccbq_resize()`, `xpt_done()`, `xpt_freeze_devq()`, `XPT_GET_TRAN_SETTINGS`, `XPT_PATH_INQ`, `xpt_print()`, `XPT_REL_SIMQ`, `xpt_setup_ccb()`, and `xpt_toggle_tags()`.

Referenced by `xpt_action()`, `xpt_dev_async()`, and `xpt_toggle_tags()`.

Here is the call graph for this function:



7.12.4.96 void xpt_setup_ccb (struct **ccb_hdr** * *ccb_h*, struct **cam_path** * *path*, u_int32_t *priority*)

Definition at line 3990 of file cam_xpt.c.

References `cam_path::bus`, `CAM_DEBUG`, `CAM_DEBUG_TRACE`, `CAM_TARGET_WILDCARD`, `CAM_UNQUEUED_INDEX`, `cam_path::device`, `ccb_hdr::flags`, `cam_pinfo::generation`, `cam_`

pinfo::index, ccb_hdr::path, ccb_hdr::path_id, ccb_hdr::pinfo, cam_pinfo::priority, cam_path::target, ccb_hdr::target_id, and ccb_hdr::target_lun.

Referenced by abort_all_pending(), cam_freeze_devq(), cam_periph_bus_settle(), cam_periph_ioctl(), cam_release_devq(), camperiphdone(), camperiphfree(), camperiphscsisenseerror(), camperiphscsisatausererror(), cddone(), cdoninvalidate(), cdregister(), chinit(), choninvalidate(), chregister(), dadone(), dadump(), dainit(), daoninvalidate(), daregister(), dasetgeom(), dashutdown(), passinit(), passioctl(), passoninvalidate(), passregister(), proberequestbackoff(), proberequestdefaultnegotiation(), probeschedule(), ptctor(), ptinit(), ptoninvalidate(), sainit(), saoninvalidate(), saregister(), scsi_command_string(), scsi_sense_sbuf(), sesinit(), sesoninvalidate(), sesregister(), STAILQ_HEAD(), TAILQ_HEAD(), targbhdslun(), targbhenlun(), targbhinit(), targenable(), targendislun(), targgetccb(), targioctl(), targusermerge(), xpt_announce_periph(), xpt_bus_register(), xpt_devise_transport(), xpt_run_dev_allocq(), xpt_scan_bus(), xpt_scan_lun(), xpt_schedule(), xpt_set_transfer_settings(), xpt_start_tags(), xpt_toggle_tags(), xptconfigbuscountfunc(), xptconfigfunc(), xptioctl(), xptsetasynbusfunc(), and xptsetasynfunc().

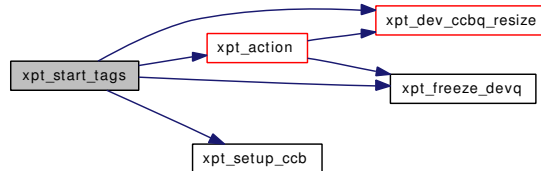
7.12.4.97 static void xpt_start_tags (struct cam_path * path) [static]

Definition at line 6833 of file cam_xpt.c.

References cam_path::bus, CAM_DEV_TAG_AFTER_COUNT, ccb_relsim::ccb_h, cam_path::device, ccb_hdr::func_code, cam_sim::max_tagged_dev_openings, ccb_relsim::openings, ccb_relsim::qfrozen_cnt, ccb_relsim::release_flags, ccb_relsim::release_timeout, RELSIM_RELEASE_AFTER_QEMPTY, SID_CmdQue, xpt_action(), xpt_dev_ccbq_resize(), xpt_freeze_devq(), XPT_REL_SIMQ, and xpt_setup_ccb().

Referenced by camisr().

Here is the call graph for this function:



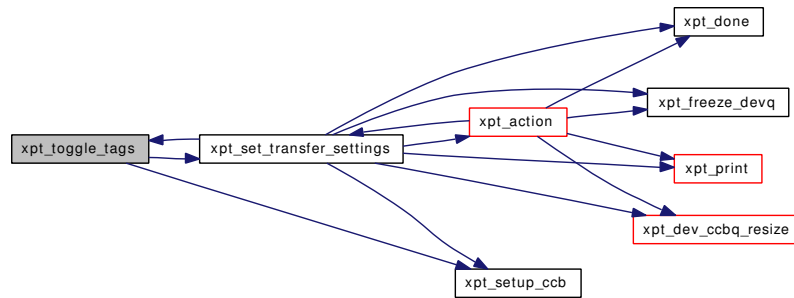
7.12.4.98 static void xpt_toggle_tags (struct cam_path * path) [static]

Definition at line 6800 of file cam_xpt.c.

References CAM_DEV_TAG_AFTER_COUNT, ccb_trans_settings::ccb_h, CTS SCSI_FLAGS_TAG_ENB, CTS SCSI_VALID_TQ, cam_path::device, ccb_trans_settings_scsi::flags, PROTO_SCSI, ccb_trans_settings::proto_specific, PROTO_VERSION_UNSPECIFIED, ccb_trans_settings::protocol, ccb_trans_settings::protocol_version, ccb_trans_settings::scsi, SID_CmdQue, SID_Sync, SID_WBus16, SID_WBus32, ccb_trans_settings::transport, ccb_trans_settings::transport_version, ccb_trans_settings_scsi::valid, XPORT_UNSPECIFIED, XPORT_VERSION_UNSPECIFIED, xpt_set_transfer_settings(), and xpt_setup_ccb().

Referenced by xpt_dev_async(), and xpt_set_transfer_settings().

Here is the call graph for this function:



7.12.4.99 static void xptaction (struct [cam_sim](#) * *sim*, union [ccb](#) * *work_ccb*) [static]

Definition at line 7065 of file `cam_xpt.c`.

References `ccb_pathinq::base_transfer_speed`, `cam_sim::bus_id`, `ccb_pathinq::bus_id`, `CAM_DEBUG`, `CAM_DEBUG_TRACE`, `CAM_REQ_CMP`, `CAM_REQ_INVALID`, `ccb_pathinq::ccb_h`, `ccb::ccb_h`, `ccb::cpi`, `DEV_IDLEN`, `ccb_pathinq::dev_name`, `ccb_hdr::func_code`, `ccb_pathinq::hba_eng_cnt`, `HBA_IDLEN`, `ccb_pathinq::hba_inquiry`, `ccb_pathinq::hba_misc`, `ccb_pathinq::hba_vid`, `ccb_pathinq::initiator_id`, `ccb_pathinq::max_lun`, `ccb_pathinq::max_target`, `ccb_hdr::path`, `PROTO_UNSPECIFIED`, `PROTO_VERSION_UNSPECIFIED`, `ccb_pathinq::protocol`, `ccb_pathinq::protocol_version`, `SIM_IDLEN`, `cam_sim::sim_name`, `ccb_pathinq::sim_vid`, `ccb_hdr::status`, `ccb_pathinq::target_sprt`, `ccb_pathinq::transport`, `ccb_pathinq::transport_version`, `cam_sim::unit_number`, `ccb_pathinq::unit_number`, `ccb_pathinq::version_num`, `XPORT_UNSPECIFIED`, `XPORT_VERSION_UNSPECIFIED`, `xpt_done()`, and `XPT_PATH_INQ`.

Referenced by `xpt_init()`.

Here is the call graph for this function:



7.12.4.100 static [dev_match_ret](#) xptbusmatch (struct [dev_match_pattern](#) * *patterns*, u_int *num_patterns*, struct [cam_eb](#) * *bus*) [static]

Definition at line 1766 of file `cam_xpt.c`.

References `BUS_MATCH_ANY`, `BUS_MATCH_BUS_ID`, `BUS_MATCH_NAME`, `BUS_MATCH_NONE`, `BUS_MATCH_PATH`, `BUS_MATCH_UNIT`, `match_pattern::bus_pattern`, `DEV_IDLEN`, `DEV_MATCH_BUS`, `DM_RET_ACTION_MASK`, `DM_RET_COPY`, `DM_RET_DESCEND`, `DM_RET_ERROR`, `DM_RET_NONE`, and `dev_match_pattern::pattern`.

Referenced by `xptedtbusfunc()`.

7.12.4.101 static int xptbustraverse (struct [cam_eb](#) * *start_bus*, [xpt_busfunc_t](#) * *tr_func*, void * *arg*) [static]

Definition at line 2629 of file `cam_xpt.c`.

Referenced by `xpt_for_all_busses()`, `xpt_for_all_devices()`, and `xptedtmatch()`.

7.12.4.102 `static int xptclose (struct cdev * dev, int flag, int fmt, struct thread * td)` [static]

Definition at line 1034 of file cam_xpt.c.

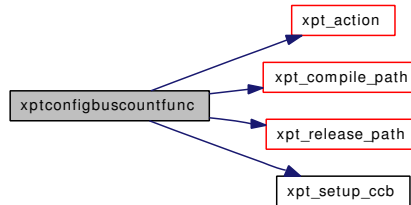
References `xpt_softc::flags`, `XPT_FLAG_OPEN`, and `xsoftc`.

7.12.4.103 `static int xptconfigbuscountfunc (struct cam_bcb * bus, void * arg)` [static]

Definition at line 6865 of file cam_xpt.c.

References `CAM_LUN_WILDCARD`, `CAM_TARGET_WILDCARD`, `CAM_XPT_PATH_ID`, `ccb_pathinq::ccb_h`, `ccb_hdr::func_code`, `PI_SDTR_ABLE`, `PI_WIDE_16`, `PI_WIDE_32`, `PIM_NOBUSRESET`, `xpt_action()`, `xpt_compile_path()`, `XPT_PATH_INQ`, `xpt_release_path()`, and `xpt_setup_ccb()`.

Here is the call graph for this function:

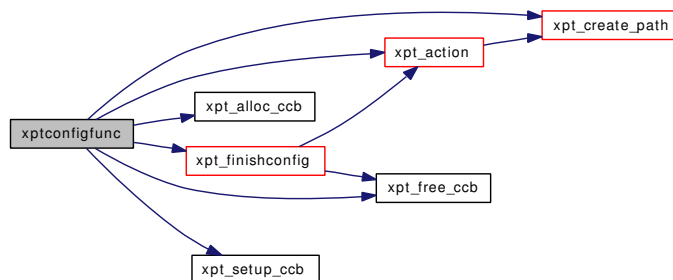


7.12.4.104 `static int xptconfigfunc (struct cam_bcb * bus, void * arg)` [static]

Definition at line 6890 of file cam_xpt.c.

References `CAM_DEBUG`, `CAM_DEBUG_SUBTRACE`, `CAM_LUN_WILDCARD`, `CAM_REQ_CMP`, `CAM_TARGET_WILDCARD`, `CAM_XPT_PATH_ID`, `ccb_hdr::cbfnp`, `ccb::ccb_h`, `ccb::cpi`, `ccb_hdr::func_code`, `ccb_pathinq::hba_inquiry`, `ccb_pathinq::hba_misc`, `PI_SDTR_ABLE`, `PI_WIDE_16`, `PI_WIDE_32`, `PIM_NOBUSRESET`, `ccb_hdr::status`, `xpt_action()`, `xpt_alloc_ccb()`, `xpt_create_path()`, `xpt_finishconfig()`, `xpt_free_ccb()`, `XPT_PATH_INQ`, `xpt_periph`, `XPT_RESET_BUS`, and `xpt_setup_ccb()`.

Here is the call graph for this function:



7.12.4.105 `static int xptdefbusfunc (struct cam_bcb * bus, void * arg)` [static]

Definition at line 2772 of file cam_xpt.c.

References `xpt_traverse_config::depth`, `xpt_traverse_config::tr_arg`, `xpt_traverse_config::tr_func`, `XPT_DEPTH_BUS`, and `xpttargettraverse()`.

Here is the call graph for this function:



7.12.4.106 `static int xptdefdevicefunc (struct cam_ed * device, void * arg)` [static]

Definition at line 2806 of file `cam_xpt.c`.

References `xpt_traverse_config::depth`, `xpt_traverse_config::tr_arg`, `xpt_traverse_config::tr_func`, `XPT_DEPTH_DEVICE`, and `xptperiphtraverse()`.

Here is the call graph for this function:



7.12.4.107 `static int xptdefperiphfunc (struct cam_periph * periph, void * arg)` [static]

Definition at line 2823 of file `cam_xpt.c`.

References `xpt_traverse_config::tr_arg`, and `xpt_traverse_config::tr_func`.

7.12.4.108 `static int xptdeftargetfunc (struct cam_et * target, void * arg)` [static]

Definition at line 2789 of file `cam_xpt.c`.

References `xpt_traverse_config::depth`, `xpt_traverse_config::tr_arg`, `xpt_traverse_config::tr_func`, `XPT_DEPTH_TARGET`, and `xptdevicetraverse()`.

Here is the call graph for this function:



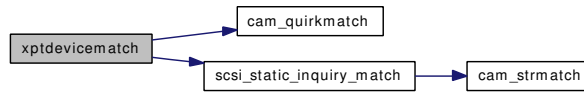
7.12.4.109 `static dev_match_ret xptdevicematch (struct dev_match_pattern * patterns, u_int num_patterns, struct cam_ed * device)` [static]

Definition at line 1878 of file `cam_xpt.c`.

References `cam_quirkmatch()`, `DEV_MATCH_ANY`, `DEV_MATCH_DEVICE`, `DEV_MATCH_INQUIRY`, `DEV_MATCH_LUN`, `DEV_MATCH_NONE`, `DEV_MATCH_PATH`, `DEV_MATCH_PERIPH`, `DEV_MATCH_TARGET`, `match_pattern::device_pattern`, `DM_RET_COPY`, `DM_RET_DESCEND`, `DM_RET_ERROR`, `DM_RET_NONE`, `dev_match_pattern::pattern`, and `scsi_static_inquiry_match()`.

Referenced by `xptdefdevicefunc()`.

Here is the call graph for this function:



7.12.4.110 `static int xptdevicetraverse (struct cam_et * target, struct cam_ed * start_device, xpt_devicefunc_t * tr_func, void * arg)` [static]

Definition at line 2673 of file cam_xpt.c.

Referenced by xptdeftargetfunc(), and xptedttargetfunc().

7.12.4.111 `static void xptdone (struct cam_periph * periph, union ccb * done_ccb)` [static]

Definition at line 990 of file cam_xpt.c.

References ccb_hdr::cbfcnp, and ccb::ccb_h.

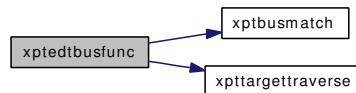
Referenced by TAILQ_HEAD(), and xptioctl().

7.12.4.112 `static int xptedtbusfunc (struct cam_eb * bus, void * arg)` [static]

Definition at line 2102 of file cam_xpt.c.

References ccb_dm_cookie::bus, bus_match_result::bus_id, match_result::bus_result, CAM_BUS_GENERATION, CAM_DEV_MATCH_ERROR, CAM_DEV_MATCH_LIST_CHANGED, CAM_DEV_MATCH_MORE, CAM_DEV_POS_BUS, CAM_DEV_POS_EDT, CAM_DEV_POS_TARGET, CAM_TARGET_GENERATION, ccb_dev_position::cookie, DEV_IDLEN, DEV_MATCH_BUS, bus_match_result::dev_name, DM_RET_COPY, DM_RET_DESCEND, DM_RET_ERROR, DM_RET_STOP, ccb_dev_position::generations, ccb_dev_match::match_buf_len, ccb_dev_match::matches, ccb_dev_match::num_matches, ccb_dev_match::num_patterns, bus_match_result::path_id, ccb_dev_match::patterns, ccb_dev_match::pos, ccb_dev_position::position_type, dev_match_result::result, ccb_dev_match::status, ccb_dm_cookie::target, dev_match_result::type, bus_match_result::unit_number, xptbusmatch(), and xpttargettraverse().

Here is the call graph for this function:



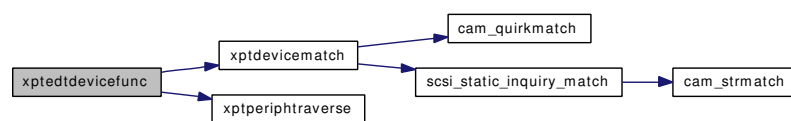
7.12.4.113 `static int xptedtdevicefunc (struct cam_ed * device, void * arg)` [static]

Definition at line 2234 of file cam_xpt.c.

References ccb_dm_cookie::bus, CAM_BUS_GENERATION, CAM_DEV_GENERATION, CAM_DEV_MATCH_ERROR, CAM_DEV_MATCH_LIST_CHANGED, CAM_DEV_MATCH_MORE,

CAM_DEV_POS_BUS, CAM_DEV_POS_DEVICE, CAM_DEV_POS_EDT, CAM_DEV_POS_PERIPH, CAM_DEV_POS_TARGET, CAM_DEV_UNCONFIGURED, CAM_PERIPH_GENERATION, CAM_TARGET_GENERATION, ccb_dev_position::cookie, DEV_MATCH_DEVICE, DEV_RESULT_NOFLAG, DEV_RESULT_UNCONFIGURED, ccb_dm_cookie::device, match_result::device_result, DM_RET_COPY, DM_RET_DESCEND, DM_RET_ERROR, DM_RET_STOP, device_match_result::flags, ccb_dev_position::generations, device_match_result::inq_data, ccb_dev_match::match_buf_len, ccb_dev_match::matches, ccb_dev_match::num_matches, ccb_dev_match::num_patterns, device_match_result::path_id, ccb_dev_match::patterns, ccb_dm_cookie::periph, ccb_dev_match::pos, ccb_dev_position::position_type, dev_match_result::result, ccb_dev_match::status, ccb_dm_cookie::target, device_match_result::target_id, device_match_result::target_lun, dev_match_result::type, xptdevicematch(), and xptperiphtraverse().

Here is the call graph for this function:



7.12.4.114 static int xptedtmatch (struct [ccb_dev_match](#) * *cdm*) [static]

Definition at line 2425 of file cam_xpt.c.

References ccb_dm_cookie::bus, CAM_BUS_GENERATION, CAM_DEV_MATCH_LAST, CAM_DEV_MATCH_LIST_CHANGED, CAM_DEV_POS_BUS, ccb_dev_position::cookie, ccb_dev_position::generations, ccb_dev_match::num_matches, ccb_dev_match::pos, ccb_dev_position::position_type, ccb_dev_match::status, and xptbustraverse().

Referenced by xpt_action().

Here is the call graph for this function:



7.12.4.115 static int xptedtperiphfunc (struct [cam_periph](#) * *periph*, void * *arg*) [static]

Definition at line 2355 of file cam_xpt.c.

References cam_path::bus, ccb_dm_cookie::bus, CAM_BUS_GENERATION, CAM_DEV_GENERATION, CAM_DEV_MATCH_ERROR, CAM_DEV_MATCH_MORE, CAM_DEV_POS_BUS, CAM_DEV_POS_DEVICE, CAM_DEV_POS_EDT, CAM_DEV_POS_PERIPH, CAM_DEV_POS_TARGET, CAM_PERIPH_GENERATION, CAM_TARGET_GENERATION, ccb_dev_position::cookie, DEV_IDLEN, DEV_MATCH_PERIPH, cam_path::device, ccb_dm_cookie::device, DM_RET_COPY, DM_RET_ERROR, ccb_dev_position::generations, ccb_dev_match::match_buf_len, ccb_dev_match::matches, ccb_dev_match::num_matches, ccb_dev_match::num_patterns, cam_periph::path, periph_match_result::path_id, ccb_dev_match::patterns, ccb_dm_cookie::periph, cam_periph::periph_name, periph_match_result::periph_name, match_result::periph_result, ccb_dev_match::pos, ccb_dev_position::position_type, dev_match_result::result, ccb_dev_match::status, cam_path::target, ccb_dm_cookie::target, periph_match_result::target_id, periph_match_result::target_lun, dev_match_result::type, cam_periph::unit_number, periph_match_result::unit_number, and xptperiphmatch().

Here is the call graph for this function:



7.12.4.116 `static int xptedttargetfunc (struct cam_et * target, void * arg)` [static]

Definition at line 2198 of file `cam_xpt.c`.

References `ccb_dm_cookie::bus`, `CAM_DEV_GENERATION`, `CAM_DEV_MATCH_LIST_CHANGED`, `CAM_DEV_POS_BUS`, `CAM_DEV_POS_DEVICE`, `CAM_DEV_POS_TARGET`, `ccb_dev_position::cookie`, `ccb_dm_cookie::device`, `ccb_dev_position::generations`, `ccb_dev_match::pos`, `ccb_dev_position::position_type`, `ccb_dev_match::status`, `ccb_dm_cookie::target`, and `xptdevicetraverse()`.

Here is the call graph for this function:

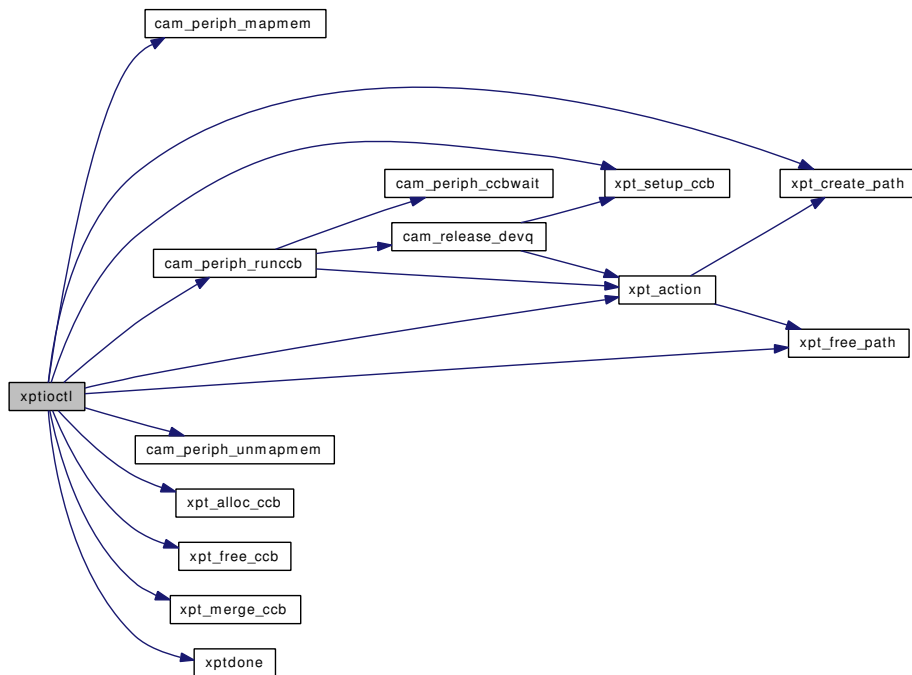


7.12.4.117 `static int xptioctl (struct cdev * dev, u_long cmd, caddr_t addr, int flag, struct thread * td)` [static]

Definition at line 1057 of file `cam_xpt.c`.

References `CAM_DATA_PHYS`, `CAM_GDEVLIST_ERROR`, `CAM_GDEVLIST_LAST_DEVICE`, `CAM_GDEVLIST_MORE_DEVS`, `CAM_LUN_WILDCARD`, `cam_periph_mapmem()`, `cam_periph_runccb()`, `cam_periph_unmapmem()`, `CAM_REQ_CMP`, `CAM_REQ_CMP_ERR`, `CAM_TARGET_WILDCARD`, `CAMGETPASSTHRU`, `CAMIOCOMMAND`, `ccb_hdr::cbfcnp`, `ccb::ccb_h`, `ccb::cgdl`, `ccb_hdr::flags`, `ccb_hdr::func_code`, `xpt_softc::generation`, `cam_periph::path`, `ccb_hdr::path`, `ccb_hdr::path_id`, `periph_drivers`, `ccb_getdevlist::periph_name`, `ccb_hdr::pinfo`, `cam_pinfo::priority`, `ccb_getdevlist::status`, `ccb_hdr::status`, `ccb_hdr::target_id`, `ccb_hdr::target_lun`, `cam_periph::unit_number`, `ccb_getdevlist::unit_number`, `xpt_action()`, `xpt_alloc_ccb()`, `xpt_create_path()`, `XPT_DEBUG`, `XPT_DEV_MATCH`, `XPT_ENG_INQ`, `xpt_free_ccb()`, `xpt_free_path()`, `xpt_merge_ccb()`, `XPT_PATH_INQ`, `xpt_periph`, `XPT_RESET_BUS`, `XPT_SCAN_BUS`, `XPT_SCAN_LUN`, `xpt_setup_ccb()`, `xptdone()`, and `xsoftc`.

Here is the call graph for this function:



7.12.4.118 static `path_id_t` `xptnextfreepathid` (void) [static]

Definition at line 4492 of file `cam_xpt.c`.

Referenced by `xptpathid()`.

7.12.4.119 static int `xptopen` (struct `cdev` * *dev*, int *flags*, int *fmt*, struct `thread` * *td*) [static]

Definition at line 997 of file `cam_xpt.c`.

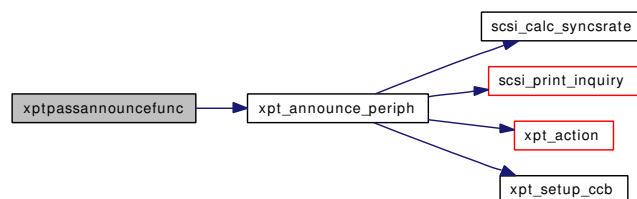
References `xpt_softc::flags`, `XPT_FLAG_OPEN`, and `xsoftc`.

7.12.4.120 static int `xptpassannouncefunc` (struct `cam_ed` * *device*, void * *arg*) [static]

Definition at line 6997 of file `cam_xpt.c`.

References `cam_periph::periph_name`, and `xpt_announce_periph()`.

Here is the call graph for this function:



7.12.4.121 `static path_id_t xptpathid (const char * sim_name, int sim_unit, int sim_bus)`
`[static]`

Definition at line 4521 of file `cam_xpt.c`.

References `CAM_XPT_PATH_ID`, and `xptnextfreepathid()`.

Referenced by `xpt_bus_register()`.

Here is the call graph for this function:



7.12.4.122 `static int xtpdperiphtraverse (struct periph_driver ** pdrv, struct cam_periph * start_periph, xpt_periphfunc_t * tr_func, void * arg)` `[static]`

Definition at line 2749 of file `cam_xpt.c`.

Referenced by `xptplistpdrvfunc()`.

7.12.4.123 `static int xtpdrvtraverse (struct periph_driver ** start_pdrv, xpt_pdrvfunc_t * tr_func, void * arg)` `[static]`

Definition at line 2721 of file `cam_xpt.c`.

References `periph_drivers`.

Referenced by `xtpperiphlistmatch()`.

7.12.4.124 `static int xtpperiphlistmatch (struct ccb_dev_match * cdm)` `[static]`

Definition at line 2591 of file `cam_xpt.c`.

References `CAM_DEV_MATCH_LAST`, `CAM_DEV_POS_PDPTR`, `ccb_dev_position::cookie`, `ccb_dev_match::num_matches`, `ccb_dm_cookie::pdrv`, `ccb_dev_match::pos`, `ccb_dev_position::position_type`, `ccb_dev_match::status`, and `xtpdrvtraverse()`.

Referenced by `xpt_action()`.

Here is the call graph for this function:



7.12.4.125 `static dev_match_ret xtpperiphmatch (struct dev_match_pattern * patterns, u_int num_patterns, struct cam_periph * periph)` `[static]`

Definition at line 1994 of file `cam_xpt.c`.

References `cam_path::bus`, `DEV_IDLEN`, `DEV_MATCH_PERIPH`, `cam_path::device`, `DM_RET_COPY`, `DM_RET_ERROR`, `DM_RET_STOP`, `cam_periph::path`, `dev_match_pattern::pattern`, `PERIPH_MATCH_ANY`, `PERIPH_MATCH_LUN`, `PERIPH_MATCH_NAME`, `PERIPH_MATCH_NONE`,

PERIPH_MATCH_PATH, PERIPH_MATCH_TARGET, PERIPH_MATCH_UNIT, cam_periph::periph_name, match_pattern::periph_pattern, cam_path::target, and cam_periph::unit_number.

Referenced by xptedtperiphfunc(), and xtplistperiphfunc().

7.12.4.126 `static int xptperiphtraverse (struct cam_ed * device, struct cam_periph * start_periph, xpt_periphfunc_t * tr_func, void * arg)` [static]

Definition at line 2697 of file cam_xpt.c.

Referenced by xptdefdevicefunc(), and xptedtdevicefunc().

7.12.4.127 `static int xtplistpdrvfunc (struct periph_driver ** pdrv, void * arg)` [static]

Definition at line 2462 of file cam_xpt.c.

References CAM_DEV_MATCH_LIST_CHANGED, CAM_DEV_POS_PDPTR, CAM_DEV_POS_PERIPH, CAM_PERIPH_GENERATION, ccb_dev_position::cookie, ccb_dev_position::generations, ccb_dm_cookie::pdrv, ccb_dm_cookie::periph, ccb_dev_match::pos, ccb_dev_position::position_type, ccb_dev_match::status, and xptpdperiphtraverse().

Here is the call graph for this function:



7.12.4.128 `static int xtplistperiphfunc (struct cam_periph * periph, void * arg)` [static]

Definition at line 2490 of file cam_xpt.c.

References cam_path::bus, CAM_DEV_MATCH_ERROR, CAM_DEV_MATCH_MORE, CAM_DEV_POS_PDPTR, CAM_DEV_POS_PDRV, CAM_DEV_POS_PERIPH, CAM_PERIPH_GENERATION, ccb_dev_position::cookie, DEV_IDLEN, DEV_MATCH_PERIPH, cam_path::device, DM_RET_COPY, DM_RET_ERROR, ccb_dev_position::generations, ccb_dev_match::match_buf_len, ccb_dev_match::matches, ccb_dev_match::num_matches, ccb_dev_match::num_patterns, cam_periph::path, periph_match_result::path_id, ccb_dev_match::patterns, ccb_dm_cookie::pdrv, ccb_dm_cookie::periph, periph_drivers, periph_match_result::periph_name, cam_periph::periph_name, match_result::periph_result, ccb_dev_match::pos, ccb_dev_position::position_type, dev_match_result::result, ccb_dev_match::status, cam_path::target, periph_match_result::target_id, periph_match_result::target_lun, dev_match_result::type, cam_periph::unit_number, periph_match_result::unit_number, and xptperiphmatch().

Here is the call graph for this function:



7.12.4.129 `static void xtpoll (struct cam_sim * sim)` [static]

Definition at line 7110 of file cam_xpt.c.

Referenced by xpt_init().

7.12.4.130 `static cam_status xptregister (struct cam_periph * periph, void * arg) [static]`

Definition at line 1544 of file `cam_xpt.c`.

References `CAM_REQ_CMP`, `CAM_REQ_CMP_ERR`, `cam_periph::softc`, and `xpt_periph`.

Referenced by `xpt_init()`.

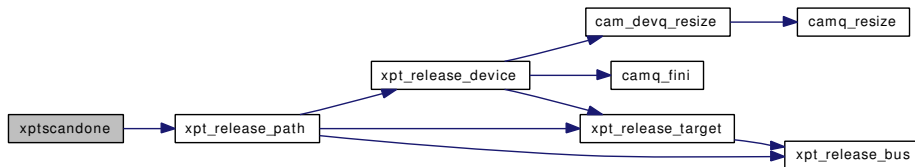
7.12.4.131 `static void xptscandone (struct cam_periph * periph, union ccb * done_ccb) [static]`

Definition at line 5645 of file `cam_xpt.c`.

References `ccb::ccb_h`, `ccb_hdr::path`, and `xpt_release_path()`.

Referenced by `xpt_scan_lun()`.

Here is the call graph for this function:

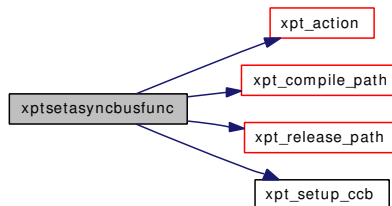


7.12.4.132 `static int xptsetasynbusfunc (struct cam_eb * bus, void * arg) [static]`

Definition at line 2939 of file `cam_xpt.c`.

References `AC_PATH_REGISTERED`, `CAM_LUN_WILDCARD`, `CAM_TARGET_WILDCARD`, `ccb_pathinq::ccb_h`, `ccb_hdr::func_code`, `xpt_action()`, `xpt_compile_path()`, `XPT_PATH_INQ`, `xpt_release_path()`, and `xpt_setup_ccb()`.

Here is the call graph for this function:

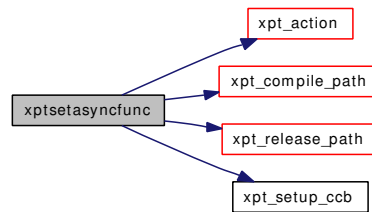


7.12.4.133 `static int xptsetasynfunc (struct cam_ed * device, void * arg) [static]`

Definition at line 2905 of file `cam_xpt.c`.

References `AC_FOUND_DEVICE`, `CAM_DEV_UNCONFIGURED`, `ccb_getdev::ccb_h`, `ccb_hdr::func_code`, `xpt_action()`, `xpt_compile_path()`, `XPT_GDEV_TYPE`, `xpt_release_path()`, and `xpt_setup_ccb()`.

Here is the call graph for this function:



7.12.4.134 `static int xpttargettraverse (struct cam_eb * bus, struct cam_et * start_target, xpt_targetfunc_t * tr_func, void * arg)` [static]

Definition at line 2650 of file `cam_xpt.c`.

Referenced by `xptdefbusfunc()`, and `xptedtbusfunc()`.

7.12.5 Variable Documentation

7.12.5.1 `int busses_to_config` [static]

Definition at line 6861 of file `cam_xpt.c`.

7.12.5.2 `int busses_to_reset` [static]

Definition at line 6862 of file `cam_xpt.c`.

7.12.5.3 `struct cam_sim cam_dead_sim` [static]

Initial value:

```

{
    .sim_action =    dead_sim_action,
    .sim_poll =     dead_sim_poll,
    .sim_name =     "dead_sim",
}

```

Definition at line 700 of file `cam_xpt.c`.

Referenced by `xpt_bus_deregister()`.

7.12.5.4 `int cam_srch_hi = 0` [static]

Definition at line 215 of file `cam_xpt.c`.

7.12.5.5 `const char microp[] = "MICROP"` [static]

Definition at line 254 of file `cam_xpt.c`.

7.12.5.6 struct `periph_driver probe_driver` [static]**Initial value:**

```
{
    probe_periph_init, "probe",
}
```

Definition at line 671 of file `cam_xpt.c`.

7.12.5.7 const char `quantum`[] = "QUANTUM" [static]

Definition at line 249 of file `cam_xpt.c`.

7.12.5.8 const char `samsung`[] = "SAMSUNG" [static]

Definition at line 252 of file `cam_xpt.c`.

7.12.5.9 const char `seagate`[] = "SEAGATE" [static]

Definition at line 253 of file `cam_xpt.c`.

7.12.5.10 const char `sony`[] = "SONY" [static]

Definition at line 250 of file `cam_xpt.c`.

7.12.5.11 const char `west_digital`[] = "WDIGTL" [static]

Definition at line 251 of file `cam_xpt.c`.

7.12.5.12 struct `cdevsw xpt_cdevsw` [static]**Initial value:**

```
{
    .d_version =    D_VERSION,
    .d_flags =     D_NEEDGIANT,
    .d_open =      xptopen,
    .d_close =     xptclose,
    .d_ioctl =     xptioctl,
    .d_name =      "xpt",
}
```

Definition at line 685 of file `cam_xpt.c`.

7.12.5.13 struct `intr_config_hook* xpt_config_hook` [static]

Definition at line 694 of file `cam_xpt.c`.

7.12.5.14 `struct xpt_quirk_entry xpt_quirk_table[]` [static]

Definition at line 256 of file cam_xpt.c.

7.12.5.15 `const int xpt_quirk_table_size` [static]

Initial value:

```
sizeof(xpt_quirk_table) / sizeof(*xpt_quirk_table)
```

Definition at line 610 of file cam_xpt.c.

7.12.5.16 `timeout_t xpt_release_devq_timeout` [static]

Definition at line 783 of file cam_xpt.c.

7.12.5.17 `timeout_t xpt_release_simq_timeout` [static]

Definition at line 784 of file cam_xpt.c.

7.12.5.18 `d_close_t xptclose` [static]

Definition at line 682 of file cam_xpt.c.

7.12.5.19 `xpt_busfunc_t xptconfigbuscountfunc` [static]

Definition at line 808 of file cam_xpt.c.

7.12.5.20 `xpt_busfunc_t xptconfigfunc` [static]

Definition at line 809 of file cam_xpt.c.

7.12.5.21 `xpt_busfunc_t xptdefbusfunc` [static]

Definition at line 854 of file cam_xpt.c.

7.12.5.22 `xpt_devicefunc_t xptdefdevicefunc` [static]

Definition at line 856 of file cam_xpt.c.

7.12.5.23 `xpt_periphfunc_t xptdefperiphfunc` [static]

Definition at line 857 of file cam_xpt.c.

7.12.5.24 `xpt_targetfunc_t xptdeftargetfunc` [static]

Definition at line 855 of file cam_xpt.c.

7.12.5.25 [xpt_busfunc_t](#) `xptedtbusfunc` [static]

Definition at line 829 of file cam_xpt.c.

7.12.5.26 [xpt_devicefunc_t](#) `xptedtdevicefunc` [static]

Definition at line 831 of file cam_xpt.c.

7.12.5.27 [xpt_periphfunc_t](#) `xptedtperiphfunc` [static]

Definition at line 832 of file cam_xpt.c.

7.12.5.28 [xpt_targetfunc_t](#) `xptedttargetfunc` [static]

Definition at line 830 of file cam_xpt.c.

7.12.5.29 [d_ioctl_t](#) `xptioctl` [static]

Definition at line 683 of file cam_xpt.c.

7.12.5.30 [d_open_t](#) `xptopen` [static]

Definition at line 681 of file cam_xpt.c.

7.12.5.31 [xpt_devicefunc_t](#) `xtpassannouncefunc` [static]

Definition at line 811 of file cam_xpt.c.

7.12.5.32 [xpt_drvfunc_t](#) `xptplistdrvfunc` [static]

Definition at line 833 of file cam_xpt.c.

7.12.5.33 [xpt_periphfunc_t](#) `xptplistperiphfunc` [static]

Definition at line 834 of file cam_xpt.c.

7.12.5.34 [xpt_busfunc_t](#) `xptsetasynbusfunc` [static]

Definition at line 870 of file cam_xpt.c.

7.12.5.35 [xpt_devicefunc_t](#) `xptsetasynfunc` [static]

Definition at line 869 of file cam_xpt.c.

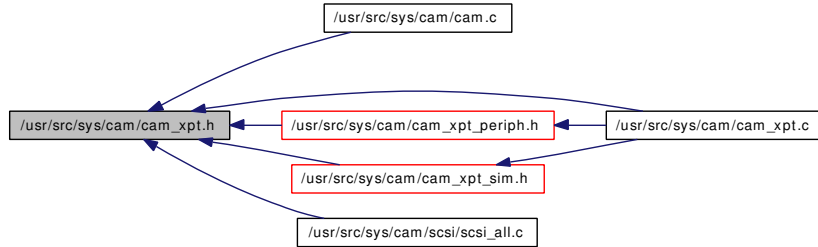
7.12.5.36 struct `xpt_softc` `xsoftc` [static]

Definition at line 643 of file `cam_xpt.c`.

Referenced by `xpt_add_periph()`, `xpt_remove_periph()`, `xptclose()`, `xptioctl()`, and `xptopen()`.

7.13 /usr/src/sys/cam/cam_xpt.h File Reference

This graph shows which files directly or indirectly include this file:



Defines

- `#define _CAM_CAM_XPT_H 1`

Functions

- `void xpt_action (union ccb *new_ccb)`
- `void xpt_setup_ccb (struct ccb_hdr *ccb_h, struct cam_path *path, u_int32_t priority)`
- `void xpt_merge_ccb (union ccb *master_ccb, union ccb *slave_ccb)`
- `cam_status xpt_create_path (struct cam_path **new_path_ptr, struct cam_periph *perph, path_id_t path_id, target_id_t target_id, lun_id_t lun_id)`
- `void xpt_free_path (struct cam_path *path)`
- `int xpt_path_comp (struct cam_path *path1, struct cam_path *path2)`
- `void xpt_print_path (struct cam_path *path)`
- `void xpt_print (struct cam_path *path, const char *fmt,...)`
- `int xpt_path_string (struct cam_path *path, char *str, size_t str_len)`
- `path_id_t xpt_path_path_id (struct cam_path *path)`
- `target_id_t xpt_path_target_id (struct cam_path *path)`
- `lun_id_t xpt_path_lun_id (struct cam_path *path)`
- `cam_sim * xpt_path_sim (struct cam_path *path)`
- `cam_periph * xpt_path_periph (struct cam_path *path)`
- `void xpt_async (u_int32_t async_code, struct cam_path *path, void *async_arg)`
- `void xpt_rescan (union ccb *ccb)`

7.13.1 Define Documentation

7.13.1.1 #define _CAM_CAM_XPT_H 1

Definition at line 33 of file `cam_xpt.h`.

7.13.2 Function Documentation

7.13.2.1 void xpt_action (union [ccb](#) * new_ccb)

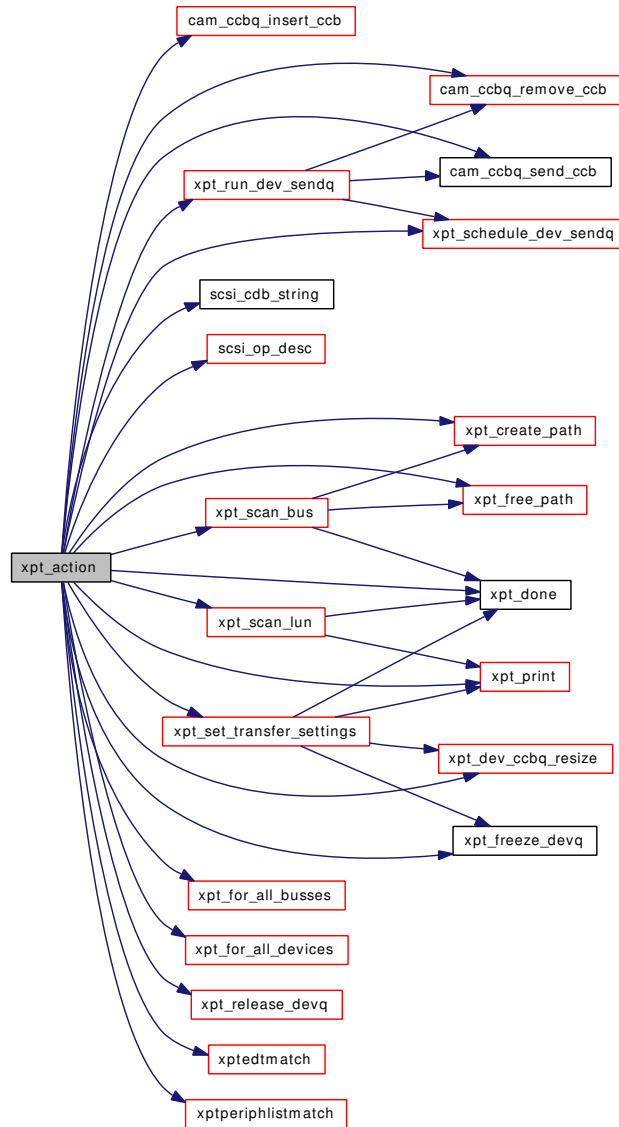
Definition at line 2963 of file `cam_xpt.c`.

References `ccb_abort::abort_ccb`, `AC_FOUND_DEVICE`, `AC_PATH_REGISTERED`, `ccb_calc_geometry::block_size`, `cam_path::bus`, `ccb::cab`, `ccb_setasync::callback`, `ccb_setasync::callback_arg`, `cam_ccbq_insert_ccb()`, `cam_ccbq_remove_ccb()`, `cam_ccbq_send_ccb()`, `CAM_CDB_POINTER`, `CAM_DEBUG`, `CAM_DEBUG_CDB`, `CAM_DEBUG_NONE`, `CAM_DEBUG_TRACE`, `CAM_DEV_MATCH_ERROR`, `CAM_DEV_NOT_THERE`, `CAM_DEV_POS_EDT`, `CAM_DEV_POS_NONE`, `CAM_DEV_POS_PDRV`, `CAM_DEV_POS_TPEMASK`, `CAM_DEV_QFREEZE`, `CAM_DEV_QFRZN`, `CAM_DEV_REL_ON_COMPLETE`, `CAM_DEV_REL_ON_QUEUE_EMPTY`, `CAM_DEV_REL_TIMEOUT_PENDING`, `CAM_DEV_UNCONFIGURED`, `CAM_DONEQ_INDEX`, `CAM_FUNC_NOTAVAIL`, `CAM_GDEVLIST_ERROR`, `CAM_GDEVLIST_LAST_DEVICE`, `CAM_GDEVLIST_LIST_CHANGED`, `CAM_GDEVLIST_MORE_DEVS`, `CAM_PROVIDE_FAIL`, `CAM_REQ_ABORTED`, `CAM_REQ_CMP`, `CAM_REQ_CMP_ERR`, `CAM_REQ_INPROG`, `CAM_RESRC_UNAVAIL`, `CAM_SIM_QUEUED`, `CAM_UA_ABORT`, `CAM_UNQUEUED_INDEX`, `ccb_relsim::ccb_h`, `ccb_setasync::ccb_h`, `ccb_getdevlist::ccb_h`, `ccb_getdevstats::ccb_h`, `ccb_getdev::ccb_h`, `ccb::ccb_h`, `ccb::csg`, `ccb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `ccb::cdbg`, `ccb::cdm`, `ccb::cgd`, `ccb::cgdl`, `ccb::cgds`, `ccb::cpis`, `ccb::crn`, `ccb::crs`, `ccb::csa`, `ccb::csio`, `ccb::cts`, `ccb_calc_geometry::cylinders`, `ccb_getdevstats::dev_active`, `DEV_IDLEN`, `DEV_MATCH_BUS`, `DEV_MATCH_DEVICE`, `ccb_getdevstats::dev_openings`, `cam_path::device`, `ccb_getdevstats::devq_openings`, `ccb_getdevstats::devq_queued`, `ccb_setasync::event_enable`, `ccb_debug::flags`, `ccb_rescan::flags`, `ccb_hdr::flags`, `ccb_hdr::func_code`, `ccb_getdevlist::generation`, `ccb_calc_geometry::heads`, `ccb_getdevstats::held`, `ccb_getdevlist::index`, `cam_pinfo::index`, `ccb_getdev::inq_data`, `INQ_DATA_TQ_ENABLED`, `ccb_getdevstats::last_reset`, `ccb_pathstats::last_reset`, `ccb_getdevstats::maxtags`, `ccb_getdevstats::mintags`, `ccb_dev_match::num_patterns`, `ccb_relsim::openings`, `ccb_hdr::path`, `ccb_hdr::path_id`, `ccb_dev_match::patterns`, `cam_path::periph`, `cam_periph::periph_name`, `ccb_getdevlist::periph_name`, `ccb_hdr::pinfo`, `ccb_dev_match::pos`, `ccb_dev_position::position_type`, `ccb_relsim::qfrozen_cnt`, `ccb_relsim::release_flags`, `ccb_relsim::release_timeout`, `RELSIM_ADJUST_OPENINGS`, `RELSIM_RELEASE_AFTER_CMDCMPLT`, `RELSIM_RELEASE_AFTER_QEMPTY`, `RELSIM_RELEASE_AFTER_TIMEOUT`, `ccb_scsiio::resid`, `scsi_cdb_string()`, `SCSI_MAX_CDBLEN`, `scsi_op_desc()`, `SCSI_REV_2`, `ccb_scsiio::scsi_status`, `SCSI_STATUS_OK`, `ccb_calc_geometry::secs_per_track`, `ccb_scsiio::sense_resid`, `ccb_getdev::serial_num`, `ccb_getdev::serial_num_len`, `cam_sim::sim_action`, `SIM_DEAD`, `ccb_getdevlist::status`, `ccb_hdr::status`, `cam_path::target`, `ccb_hdr::target_id`, `ccb_hdr::target_lun`, `dev_match_pattern::type`, `cam_periph::unit_number`, `ccb_getdevlist::unit_number`, `ccb_calc_geometry::volume_size`, `XPT_ABORT`, `XPT_ACCEPT_TARGET_IO`, `XPT_CALC_GEOMETRY`, `XPT_CONT_TARGET_IO`, `xpt_create_path()`, `XPT_DEBUG`, `xpt_dev_ccbq_resize()`, `XPT_DEV_MATCH`, `xpt_done()`, `XPT_EN_LUN`, `XPT_ENG_EXEC`, `XPT_ENG_INQ`, `XPT_FC_IS_DEV_QUEUED`, `XPT_FC_IS_QUEUED`, `xpt_for_all_busses()`, `xpt_for_all_devices()`, `xpt_free_path()`, `xpt_freeze_devq()`, `XPT_GDEV_STATS`, `XPT_GDEV_TYPE`, `XPT_GDEVLIST`, `XPT_GET_TRAN_SETTINGS`, `XPT_IMMEDIATE_NOTIFY`, `XPT_NOOP`, `XPT_NOTIFY_ACK`, `XPT_PATH_INQ`, `XPT_PATH_STATS`, `xpt_periph`, `xpt_print()`, `XPT_REL_SIMQ`, `xpt_release_devq()`, `XPT_RESET_BUS`, `XPT_RESET_DEV`, `xpt_run_dev_sendq()`, `XPT_SASYNC_CB`, `xpt_scan_bus()`, `XPT_SCAN_BUS`, `xpt_scan_lun()`, `XPT_SCAN_LUN`, `xpt_schedule_dev_sendq()`, `XPT_SCSI_IO`, `XPT_SDEV_TYPE`, `XPT_SET_TRAN_SETTINGS`, `xpt_set_transfer_settings()`, `XPT_TARGET_IO`, `XPT_TERM_IO`, `xptedtmatch()`, and `xptperiphlistmatch()`.

Referenced by `abort_all_pending()`, `cam_freeze_devq()`, `cam_periph_bus_settle()`, `cam_periph_error()`, `cam_periph_ioctl()`, `cam_periph_runccb()`, `cam_release_devq()`, `camperiphdone()`, `camperiphfree()`, `camperiphscsisenseerror()`, `camperiphscsistatuserror()`, `cd6byteworkaround()`, `cddone()`, `cdoninvalidate()`, `cdregister()`, `cdstart()`, `chdone()`, `chinit()`, `choninvalidate()`, `chregister()`, `chstart()`, `cmd6workaround()`, `dadone()`, `dainit()`, `daoninvalidate()`, `daregister()`, `dasetgeom()`, `dastart()`, `passinit()`, `passoninvalidate()`, `passregister()`, `probedone()`, `proberequestbackoff()`, `proberequestdefaultnegotiation()`, `probeschedule()`, `probstart()`, `ptctor()`, `ptinit()`, `ptoninvalidate()`, `ptstart()`, `sainit()`, `saoninvalidate()`, `saregister()`, `sastart()`, `scsi_command_string()`, `scsi_sense_sbuf()`, `sesinit()`, `sesoninvalidate()`, `sesregister()`, `STAILQ_HEAD()`, `targbhdislun()`, `targbhdone()`, `targbhendlun()`, `targbhinit()`, `targbhstart()`, `targenable()`, `targendislun()`, `targioctl()`, `targsendccb()`, `targwrite()`, `xpt_announce_periph()`, `xpt_bus_register()`, `xpt_devise_transport()`, `xpt_finishconfig()`, `xpt_polled_action()`, `xpt_scan_bus()`, `xpt_scan_lun()`, `xpt_set_transfer_settings()`, `xpt_start_tags()`, `xptconfigbuscountfunc()`, `xptconfigfunc()`, `xptioctl()`, `xptsetasynccbusfunc()`, and `xptsetasync-`

func()).

Here is the call graph for this function:



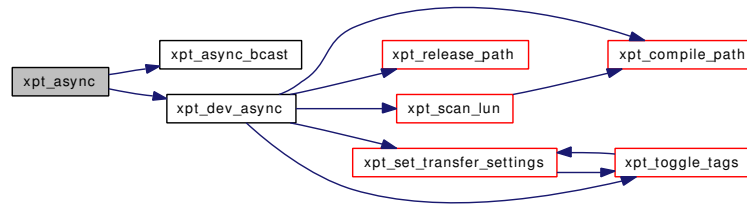
7.13.2.2 void xpt_async (u_int32_t async_code, struct cam_path * path, void * async_arg)

Definition at line 4564 of file cam_xpt.c.

References AC_BUS_RESET, AC_SENT_BDR, cam_path::bus, CAM_DEBUG, CAM_DEBUG_TRACE, CAM_LUN_WILDCARD, CAM_TARGET_WILDCARD, cam_path::device, cam_periph::path, cam_path::target, xpt_async_bcast(), xpt_dev_async(), and xpt_periph.

Referenced by cam_periph_error(), camperiphdone(), probedone(), xpt_bus_deregister(), and xpt_bus_register().

Here is the call graph for this function:



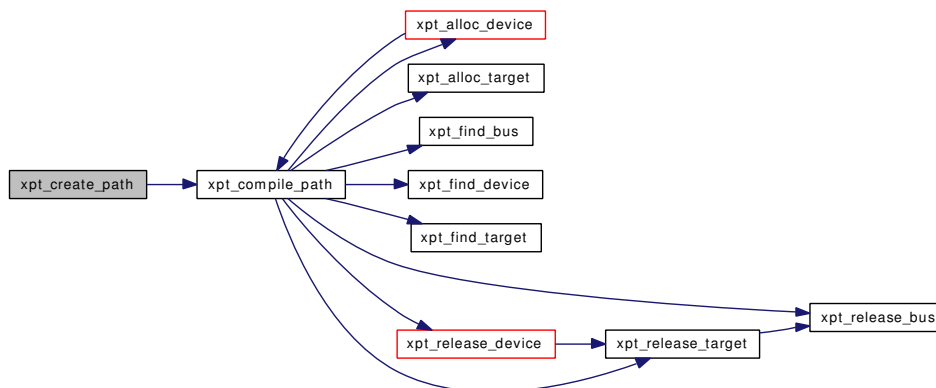
7.13.2.3 `cam_status xpt_create_path(struct cam_path ** new_path_ptr, struct cam_periph * periph, path_id_t path_id, target_id_t target_id, lun_id_t lun_id)`

Definition at line 4014 of file cam_xpt.c.

References CAM_REQ_CMP, CAM_RESRC_UNAVAIL, and xpt_compile_path().

Referenced by cam_periph_alloc(), cam_periph_error(), cdregister(), chinit(), dainit(), passinit(), ptinit(), sainit(), sesinit(), STAILQ_HEAD(), targbhasync(), targbhinit(), targioctl(), xpt_action(), xpt_config(), xpt_init(), xpt_scan_bus(), xptconfigfunc(), and xptioctl().

Here is the call graph for this function:



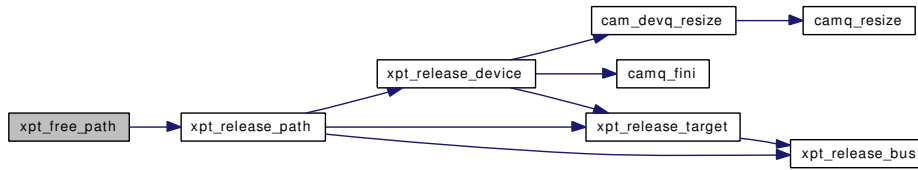
7.13.2.4 `void xpt_free_path(struct cam_path * path)`

Definition at line 4130 of file cam_xpt.c.

References CAM_DEBUG, CAM_DEBUG_TRACE, and xpt_release_path().

Referenced by cam_periph_alloc(), cam_periph_error(), camperiphfree(), cdregister(), chinit(), dainit(), passinit(), ptinit(), sainit(), sesinit(), STAILQ_HEAD(), TAILQ_HEAD(), targbhasync(), targbhinit(), targioctl(), xpt_action(), xpt_finishconfig(), xpt_init(), xpt_rescan(), xpt_scan_bus(), and xptioctl().

Here is the call graph for this function:



7.13.2.5 void xpt_merge_ccb (union ccb * master_ccb, union ccb * slave_ccb)

Definition at line 3973 of file cam_xpt.c.

References ccb::ccb_h, ccb_hdr::flags, ccb_hdr::func_code, ccb_hdr::retry_count, and ccb_hdr::timeout.

Referenced by passsendccb(), and xptioctl().

7.13.2.6 int xpt_path_comp (struct cam_path * path1, struct cam_path * path2)

Definition at line 4145 of file cam_xpt.c.

References cam_path::bus, CAM_BUS_WILDCARD, CAM_LUN_WILDCARD, CAM_TARGET_WILDCARD, cam_path::device, and cam_path::target.

Referenced by cam_periph_find(), and xpt_rescan().

7.13.2.7 lun_id_t xpt_path_lun_id (struct cam_path * path)

Definition at line 4283 of file cam_xpt.c.

References CAM_LUN_WILDCARD, and cam_path::device.

Referenced by cam_periph_alloc().

7.13.2.8 path_id_t xpt_path_path_id (struct cam_path * path)

Definition at line 4264 of file cam_xpt.c.

References cam_path::bus.

Referenced by cam_periph_alloc(), cam_periph_error(), and targbhasync().

7.13.2.9 struct cam_periph* xpt_path_periph (struct cam_path * path)

Definition at line 4302 of file cam_xpt.c.

References cam_periph::path, and cam_path::periph.

Referenced by cam_periph_error(), camperiphscsisenseerror(), cd6byteworkaround(), cderror(), cdrunccb(), cherror(), cmd6workaround(), daerror(), passerror(), pterror(), saerror(), and seserror().

7.13.2.10 struct cam_sim* xpt_path_sim (struct cam_path * path)

Definition at line 4294 of file cam_xpt.c.

References `cam_path::bus`.

7.13.2.11 `int xpt_path_string (struct cam_path * path, char * str, size_t str_len)`

Definition at line 4224 of file `cam_xpt.c`.

References `cam_path::bus`, `cam_path::device`, `cam_path::periph`, `cam_periph::periph_name`, `cam_path::target`, and `cam_periph::unit_number`.

Referenced by `cam_error_string()`, and `scsi_sense_sbuf()`.

7.13.2.12 `target_id_t xpt_path_target_id (struct cam_path * path)`

Definition at line 4272 of file `cam_xpt.c`.

References `CAM_TARGET_WILDCARD`, and `cam_path::target`.

Referenced by `cam_periph_alloc()`, and `cam_periph_error()`.

7.13.2.13 `void xpt_print (struct cam_path * path, const char * fmt, ...)`

Definition at line 4214 of file `cam_xpt.c`.

References `xpt_print_path()`.

Referenced by `abort_all_pending()`, `cam_periph_error()`, `camperiphdone()`, `camperiphnextunit()`, `camperiphscsisstatuserror()`, `cd6byteworkaround()`, `cdcleanup()`, `cddone()`, `cdgetmode()`, `cdoninvalidate()`, `cdreportkey()`, `cdstart()`, `chcleanup()`, `chdone()`, `chgetelemstatus()`, `chgetparams()`, `choninvalidate()`, `cmd6workaround()`, `dacleanup()`, `daclose()`, `dadone()`, `dadump()`, `daoninvalidate()`, `dashutdown()`, `passcleanup()`, `passioctl()`, `passoninvalidate()`, `passopen()`, `probedone()`, `proberequestbackoff()`, `probestart()`, `ptdone()`, `ptdtor()`, `ptoninvalidate()`, `sacleanup()`, `saclose()`, `saerror()`, `saiioctl()`, `samount()`, `saoninvalidate()`, `saregister()`, `sasetparams()`, `sastart()`, `sastrategy()`, `sescleanup()`, `sesoninvalidate()`, `targbhenlun()`, `targendislnun()`, `targreturnccb()`, `targstart()`, `xpt_action()`, `xpt_rescan()`, `xpt_scan_lun()`, and `xpt_set_transfer_settings()`.

Here is the call graph for this function:



7.13.2.14 `void xpt_print_path (struct cam_path * path)`

Definition at line 4181 of file `cam_xpt.c`.

References `cam_path::bus`, `cam_path::device`, `cam_path::periph`, `cam_periph::periph_name`, `cam_path::target`, and `cam_periph::unit_number`.

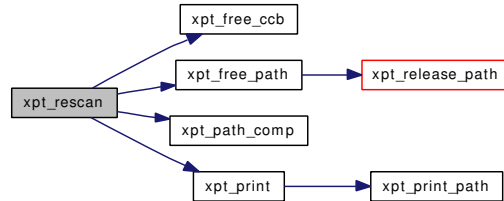
Referenced by `cam_periph_error()`, `camperiphscsisenseerror()`, `sagetparams()`, `sasetparams()`, `sastrategy()`, and `xpt_print()`.

7.13.2.15 `void xpt_rescan (union ccb * ccb)`

Definition at line 1438 of file `cam_xpt.c`.

References `ccb::ccb_h`, `ccb_hdr::path`, `ccb_hdr::sim_links`, `xpt_free_ccb()`, `xpt_free_path()`, `xpt_path_comp()`, and `xpt_print()`.

Here is the call graph for this function:



7.13.2.16 void xpt_setup_ccb (struct `ccb_hdr` * `ccb_h`, struct `cam_path` * `path`, `u_int32_t` `priority`)

Definition at line 3990 of file `cam_xpt.c`.

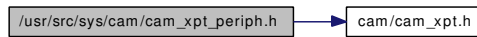
References `cam_path::bus`, `CAM_DEBUG`, `CAM_DEBUG_TRACE`, `CAM_TARGET_WILDCARD`, `CAM_UNQUEUED_INDEX`, `cam_path::device`, `ccb_hdr::flags`, `cam_pinfo::generation`, `cam_pinfo::index`, `ccb_hdr::path`, `ccb_hdr::path_id`, `ccb_hdr::pinfo`, `cam_pinfo::priority`, `cam_path::target`, `ccb_hdr::target_id`, and `ccb_hdr::target_lun`.

Referenced by `abort_all_pending()`, `cam_freeze_devq()`, `cam_periph_bus_settle()`, `cam_periph_ioctl()`, `cam_release_devq()`, `camperiphdone()`, `camperiphfree()`, `camperiphscsisenseerror()`, `camperiphscsisatuserror()`, `cddone()`, `cdoninvalidate()`, `cdregister()`, `chinit()`, `choninvalidate()`, `chregister()`, `dadone()`, `dadump()`, `dainit()`, `daoninvalidate()`, `daregister()`, `dasetgeom()`, `dashutdown()`, `passinit()`, `passioctl()`, `passoninvalidate()`, `passregister()`, `proberequestbackoff()`, `proberequestdefaultnegotiation()`, `probeschedule()`, `ptctor()`, `ptinit()`, `ptoninvalidate()`, `sainit()`, `saoninvalidate()`, `saregister()`, `scsi_command_string()`, `scsi_sense_sbuf()`, `sesinit()`, `sesoninvalidate()`, `sesregister()`, `STAILQ_HEAD()`, `TAILQ_HEAD()`, `targbhdslun()`, `targbhenlun()`, `targbhinit()`, `targenable()`, `targendislun()`, `targgetccb()`, `targioctl()`, `targusermerge()`, `xpt_announce_periph()`, `xpt_bus_register()`, `xpt_devise_transport()`, `xpt_run_dev_allocq()`, `xpt_scan_bus()`, `xpt_scan_lun()`, `xpt_schedule()`, `xpt_set_transfer_settings()`, `xpt_start_tags()`, `xpt_toggle_tags()`, `xptconfigbuscountfunc()`, `xptconfigfunc()`, `xptioctl()`, `xptsetasynbusfunc()`, and `xptsetasynfunc()`.

7.14 /usr/src/sys/cam/cam_xpt_periph.h File Reference

```
#include <cam/cam_xpt.h>
```

Include dependency graph for cam_xpt_periph.h:



This graph shows which files directly or indirectly include this file:



Defines

- `#define _CAM_CAM_XPT_PERIPH_H 1`

Functions

- `void xpt_polled_action (union ccb *ccb)`
- `ccb * xpt_alloc_ccb (void)`
- `ccb * xpt_alloc_ccb_nowait (void)`
- `void xpt_free_ccb (union ccb *free_ccb)`
- `void xpt_release_ccb (union ccb *released_ccb)`
- `void xpt_schedule (struct cam_periph *perph, u_int32_t new_priority)`
- `int32_t xpt_add_periph (struct cam_periph *periph)`
- `void xpt_remove_periph (struct cam_periph *periph)`
- `void xpt_announce_periph (struct cam_periph *periph, char *announce_string)`

7.14.1 Define Documentation

7.14.1.1 #define _CAM_CAM_XPT_PERIPH_H 1

Definition at line 34 of file cam_xpt_periph.h.

7.14.2 Function Documentation

7.14.2.1 int32_t xpt_add_periph (struct cam_periph * periph)

Definition at line 1559 of file cam_xpt.c.

References CAM_REQ_CMP, camq_resize(), cam_path::device, xpt_softc::generation, cam_periph::path, and xsoftc.

Referenced by cam_periph_alloc().

Here is the call graph for this function:



7.14.2.2 union ccb* xpt_alloc_ccb (void)

Definition at line 4929 of file cam_xpt.c.

Referenced by passioctl(), xpt_scan_bus(), xptconfigfunc(), and xptioctl().

7.14.2.3 union ccb* xpt_alloc_ccb_nowait (void)

Definition at line 4940 of file cam_xpt.c.

Referenced by xpt_get_ccb().

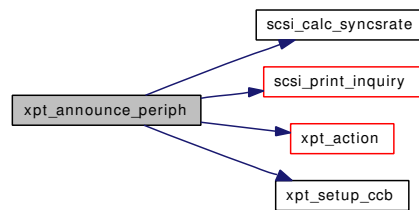
7.14.2.4 void xpt_announce_periph (struct cam_periph * periph, char * announce_string)

Definition at line 1629 of file cam_xpt.c.

References ccb_trans_settings_sas::bitrate, ccb_trans_settings_fc::bitrate, cam_path::bus, ccb_trans_settings_spi::bus_width, CAM_DEV_TAG_AFTER_COUNT, CAM_REQ_CMP, CAM_STATUS_MASK, ccb_pathinq::ccb_h, ccb::ccb_h, ccb_trans_settings::ccb_h, CTS_FC_VALID_PORT, CTS_FC_VALID_SPEED, CTS_FC_VALID_WWNN, CTS_FC_VALID_WWPN, CTS_SAS_VALID_SPEED, CTS_SPI_VALID_BUS_WIDTH, CTS_SPI_VALID_SYNC_OFFSET, CTS_TYPE_CURRENT_SETTINGS, cam_path::device, ccb_hdr::func_code, MSG_EXT_PPR_DT_REQ, cam_periph::path, cam_path::periph, cam_periph::periph_name, ccb_trans_settings_fc::port, ccb_trans_settings_spi::ppr_options, scsi_calc_syncsrate(), scsi_print_inquiry(), SID_CmdQue, ccb_hdr::status, ccb_trans_settings_spi::sync_offset, ccb_trans_settings_spi::sync_period, cam_path::target, ccb_trans_settings::type, cam_periph::unit_number, ccb_trans_settings_sas::valid, ccb_trans_settings_fc::valid, ccb_trans_settings_spi::valid, ccb_trans_settings_fc::wwnn, ccb_trans_settings_fc::wwpn, XPORT_FC, XPORT_SAS, XPORT_SPI, xpt_action(), XPT_GET_TRAN_SETTINGS, XPT_PATH_INQ, and xpt_setup_ccb().

Referenced by cddone(), chdone(), dadone(), passregister(), ptctor(), saregister(), sesregister(), and xptpassannouncefunc().

Here is the call graph for this function:



7.14.2.5 void xpt_free_ccb (union ccb * free_ccb)

Definition at line 4951 of file cam_xpt.c.

Referenced by passioctl(), TAILQ_HEAD(), targbhdone(), xpt_finishconfig(), xpt_release_ccb(), xpt_rescan(), xpt_scan_bus(), xptconfigfunc(), and xptioctl().

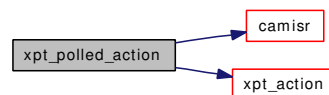
7.14.2.6 void xpt_polled_action (union ccb * ccb)

Definition at line 3596 of file cam_xpt.c.

References cam_path::bus, CAM_CMD_TIMEOUT, CAM_REQ_INPROG, CAM_RESRC_UNAVAIL, CAM_STATUS_MASK, camisr(), ccb::ccb_h, cam_path::device, cam_sim::devq, ccb_hdr::path, cam_devq::send_openings, cam_sim::sim_poll, ccb_hdr::status, ccb_hdr::timeout, and xpt_action().

Referenced by dadump(), and dashutdown().

Here is the call graph for this function:



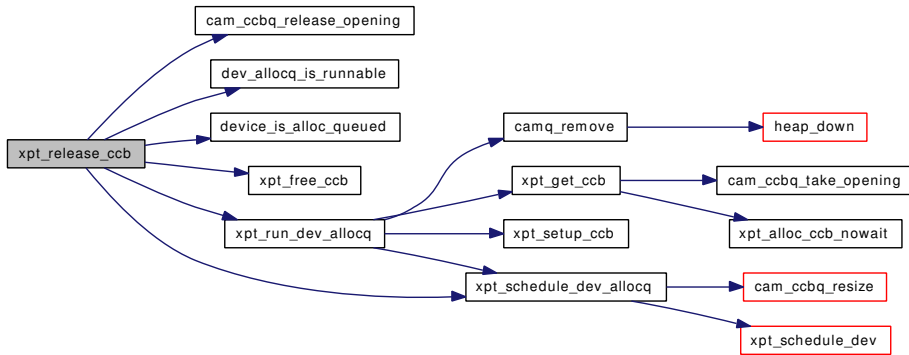
7.14.2.7 void xpt_release_ccb (union ccb * released_ccb)

Definition at line 4316 of file cam_xpt.c.

References cam_path::bus, cam_ccbq_release_opening(), CAM_DEBUG_PRINT, CAM_DEBUG_XPT, ccb::ccb_h, dev_allocq_is_runnable(), cam_path::device, device_is_alloc_queued(), ccb_hdr::path, xpt_free_ccb(), xpt_run_dev_allocq(), and xpt_schedule_dev_allocq().

Referenced by cam_periph_ioctl(), cddone(), cdgetmode(), cdpause(), cdplay(), cdplaymsf(), cdplaytracks(), cdprevent(), cdreadvdstructure(), cdreadsubchannel(), cdreadtoc(), cdreportkey(), cdsendkey(), cdsetmode(), cdsetspeed(), cdsizes(), cdstart(), cdstartunit(), cdstopunit(), chdone(), chexchange(), chgetelemstatus(), chgetparams(), chielem(), chmove(), chposition(), daclose(), dadone(), dagetcapacity(), daprevent(), dastart(), passdone(), passioctl(), probedone(), ptdone(), ptstart(), sadone(), saerase(), sagetparams(), saloadunload(), samount(), saprevent(), sardpos(), sarservereleaseunit(), saretension(), sarewind(), sasetparams(), sasetpos(), saspace(), sastart(), sawritefilemarks(), ses_runcmd(), targbhdone(), targbhstart(), targfreecb(), and targstart().

Here is the call graph for this function:



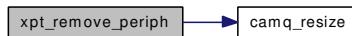
7.14.2.8 void xpt_remove_periph (struct cam_periph * periph)

Definition at line 1598 of file cam_xpt.c.

References camq_resize(), cam_path::device, xpt_softc::generation, cam_periph::path, and xsoftc.

Referenced by cam_periph_alloc(), and camperiphfree().

Here is the call graph for this function:



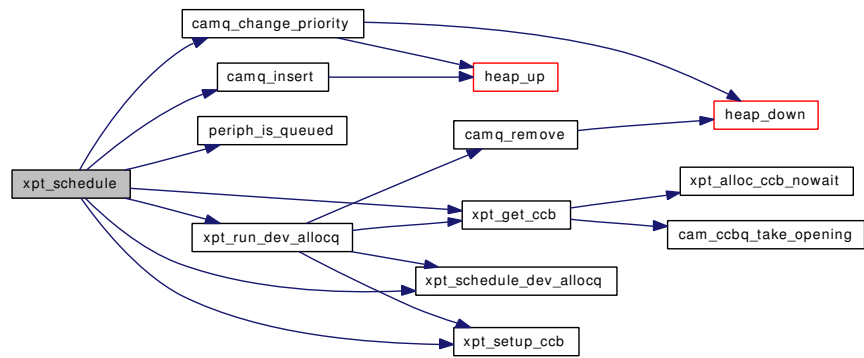
7.14.2.9 void xpt_schedule (struct cam_periph * periph, u_int32_t new_priority)

Definition at line 3660 of file cam_xpt.c.

References cam_path::bus, CAM_DEBUG, CAM_DEBUG_SUBTRACE, CAM_DEBUG_TRACE, camq_change_priority(), camq_insert(), ccb::ccb_h, cam_path::device, cam_pinfo::generation, cam_pinfo::index, cam_periph::path, periph_is_queued(), cam_periph::periph_start, cam_periph::pinfo, cam_pinfo::priority, SIM_DEAD, xpt_get_ccb(), xpt_run_dev_allocq(), xpt_schedule_dev_allocq(), and xpt_setup_ccb().

Referenced by cam_periph_getccb(), cdregister(), cdrunchangerqueue(), cdschedule(), cdstart(), cdstrategy(), chregister(), dadone(), daregister(), dastart(), dastrategy(), probedone(), probeschedule(), ptstart(), ptstrategy(), sastart(), sastrategy(), targbhdone(), targbhstart(), targstart(), targwrite(), and xpt_bus_deregister().

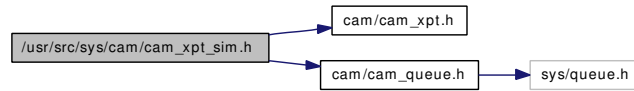
Here is the call graph for this function:



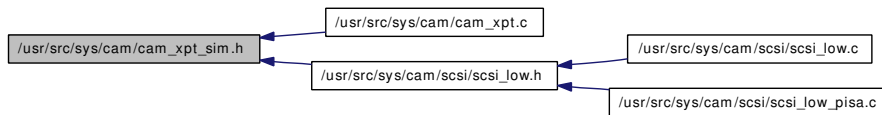
7.15 /usr/src/sys/cam/cam_xpt_sim.h File Reference

```
#include <cam/cam_xpt.h>
#include <cam/cam_queue.h>
```

Include dependency graph for cam_xpt_sim.h:



This graph shows which files directly or indirectly include this file:



Defines

- #define `_CAM_CAM_XPT_SIM_H` 1

Functions

- `int32_t xpt_bus_register` (struct `cam_sim` *sim, `u_int32_t` bus)
- `int32_t xpt_bus_deregister` (`path_id_t` path_id)
- `u_int32_t xpt_freeze_simq` (struct `cam_sim` *sim, `u_int` count)
- `void xpt_release_simq` (struct `cam_sim` *sim, `int` run_queue)
- `u_int32_t xpt_freeze_devq` (struct `cam_path` *path, `u_int` count)
- `void xpt_release_devq` (struct `cam_path` *path, `u_int` count, `int` run_queue)
- `void xpt_done` (union `ccb` *done_ccb)

7.15.1 Define Documentation

7.15.1.1 #define _CAM_CAM_XPT_SIM_H 1

Definition at line 33 of file `cam_xpt_sim.h`.

7.15.2 Function Documentation

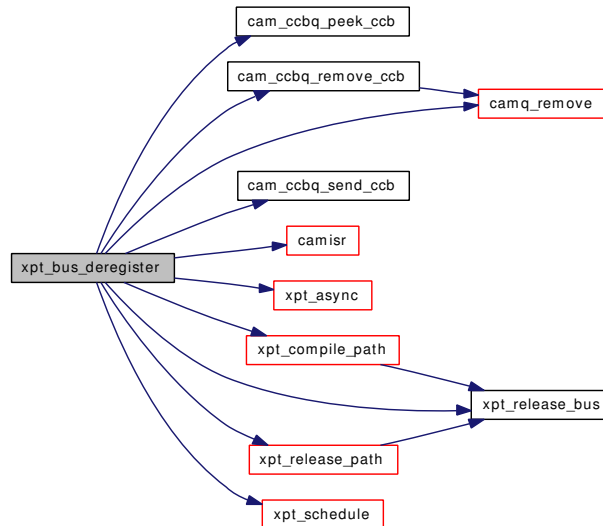
7.15.2.1 `int32_t xpt_bus_deregister` (`path_id_t` path_id)

Definition at line 4423 of file `cam_xpt.c`.

References `AC_LOST_DEVICE`, `AC_PATH_DEREGISTERED`, `cam_devq::active_dev`, `cam_devq::alloc_queue`, `cam_path::bus`, `cam_ccbq_peek_ccb()`, `cam_ccbq_remove_ccb()`, `cam_ccbq_send_ccb()`, `cam_dead_sim`, `CAM_LUN_WILDCARD`, `CAM_REQ_CMP`, `CAM_TARGET_WILDCARD`,

camisr(), CAMQ_HEAD, camq_remove(), ccb::ccb_h, ccb_hdr::path, cam_periph::pinfo, cam_pinfo::priority, cam_devq::send_queue, cam_sim::sim_action, xpt_async(), xpt_compile_path(), xpt_release_bus(), xpt_release_path(), and xpt_schedule().

Here is the call graph for this function:



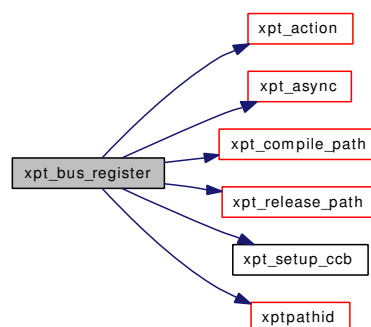
7.15.2.2 int32_t xpt_bus_register (struct cam_sim * sim, u_int32_t bus)

Definition at line 4365 of file `cam_xpt.c`.

References `AC_PATH_REGISTERED`, `cam_sim::bus_id`, `CAM_LUN_WILDCARD`, `CAM_RESRC_UNAVAIL`, `CAM_SUCCESS`, `CAM_TARGET_WILDCARD`, `CAM_XPT_PATH_ID`, `ccb_pathinq::ccb_h`, `ccb_hdr::func_code`, `cam_sim::path_id`, `cam_sim::sim_name`, `cam_sim::unit_number`, `xpt_action()`, `xpt_async()`, `xpt_compile_path()`, `XPT_PATH_INQ`, `xpt_release_path()`, `xpt_setup_ccb()`, and `xptpathid()`.

Referenced by `xpt_init()`.

Here is the call graph for this function:



7.15.2.3 void xpt_done (union ccb * done_ccb)

Definition at line 4899 of file `cam_xpt.c`.

References CAM_DEBUG, CAM_DEBUG_TRACE, CAM_DONEQ_INDEX, CAM_PERIPH_BIO, ccb::ccb_h, ccb_hdr::func_code, cam_pinfo::index, ccb_hdr::path, cam_path::periph, ccb_hdr::pinfo, cam_periph::type, and XPT_FC_QUEUED.

Referenced by dead_sim_action(), probedone(), xpt_action(), xpt_scan_bus(), xpt_scan_lun(), xpt_set_transfer_settings(), and xptaction().

7.15.2.4 u_int32_t xpt_freeze_devq (struct cam_path * path, u_int count)

Definition at line 4733 of file cam_xpt.c.

References CAM_REQ_INPROG, CAM_REQUEUE_REQ, cam_path::device, ccb_hdr::path, and ccb_hdr::status.

Referenced by xpt_action(), xpt_set_transfer_settings(), and xpt_start_tags().

7.15.2.5 u_int32_t xpt_freeze_simq (struct cam_sim * sim, u_int count)

Definition at line 4764 of file cam_xpt.c.

References cam_devq::active_dev, CAM_REQ_INPROG, CAM_REQUEUE_REQ, cam_sim::devq, camq::qfrozen_cnt, cam_devq::send_queue, and ccb_hdr::status.

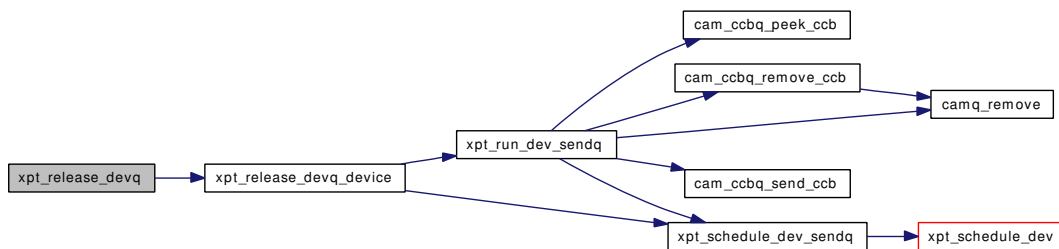
7.15.2.6 void xpt_release_devq (struct cam_path * path, u_int count, int run_queue)

Definition at line 4791 of file cam_xpt.c.

References cam_path::device, and xpt_release_devq_device().

Referenced by camisr(), probedone(), and xpt_action().

Here is the call graph for this function:



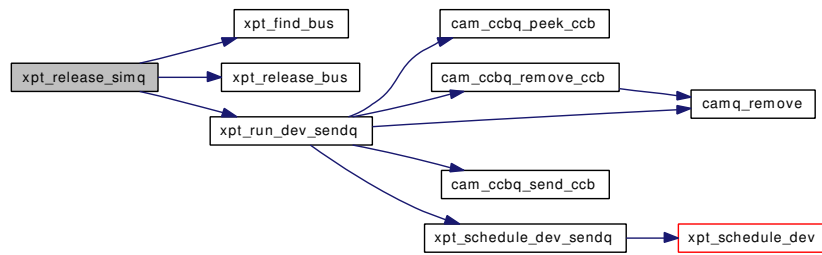
7.15.2.7 void xpt_release_simq (struct cam_sim * sim, int run_queue)

Definition at line 4848 of file cam_xpt.c.

References cam_sim::c_handle, CAM_SIM_REL_TIMEOUT_PENDING, cam_sim::devq, cam_sim::flags, cam_sim::path_id, camq::qfrozen_cnt, cam_devq::send_queue, xpt_find_bus(), xpt_release_bus(), and xpt_run_dev_sendq().

Referenced by camisr(), and xpt_release_simq_timeout().

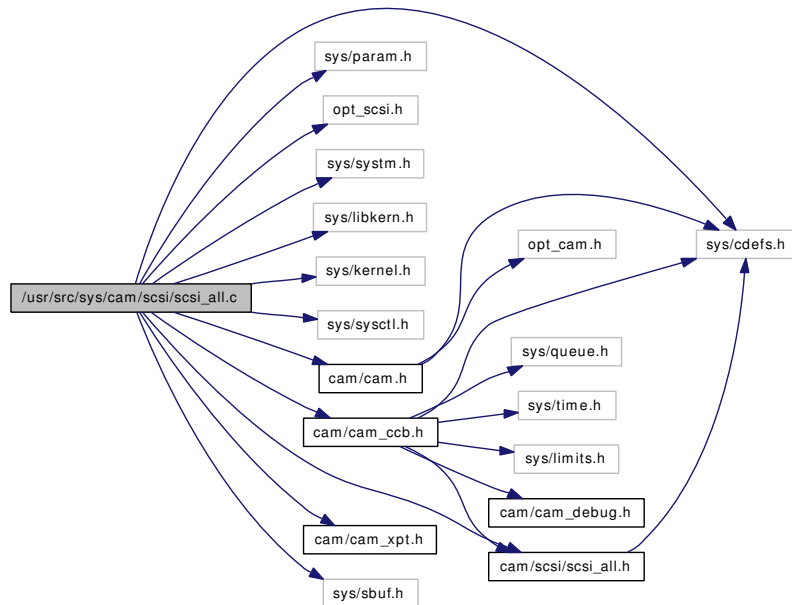
Here is the call graph for this function:



7.16 /usr/src/sys/cam/scsi/scsi_all.c File Reference

```
#include <sys/cdefs.h>
#include <sys/param.h>
#include <opt_scsi.h>
#include <sys/system.h>
#include <sys/libkern.h>
#include <sys/kernel.h>
#include <sys/sysctl.h>
#include <cam/cam.h>
#include <cam/cam_ccb.h>
#include <cam/cam_xpt.h>
#include <cam/scsi/scsi_all.h>
#include <sys/sbuf.h>
```

Include dependency graph for scsi_all.c:



Data Structures

- struct [asc_key](#)

Defines

- #define [SCSI_DELAY](#) 2000

- #define [SCSI_MIN_DELAY](#) 100
- #define [D](#) 0x001
- #define [T](#) 0x002
- #define [L](#) 0x004
- #define [P](#) 0x008
- #define [W](#) 0x010
- #define [R](#) 0x020
- #define [S](#) 0x040
- #define [O](#) 0x080
- #define [M](#) 0x100
- #define [C](#) 0x200
- #define [A](#) 0x400
- #define [E](#) 0x800
- #define [ALL](#) 0xFF
- #define [SST](#)(asc, ascq, action, desc) asc, ascq, action, desc

Functions

- [__FBSDID](#) ("FreeBSD: src/sys/cam/scsi/scsi_all.c,v 1.49 2006/08/21 13:24:50 ken Exp \$")
- static int [ascentrycomp](#) (const void *key, const void *member)
- static int [senseentrycomp](#) (const void *key, const void *member)
- static void [fetchtableentries](#) (int sense_key, int asc, int ascq, struct [scsi_inquiry_data](#) *, const struct [sense_key_table_entry](#) **, const struct [asc_table_entry](#) **)
- static void [init_scsi_delay](#) (void)
- static int [sysctl_scsi_delay](#) (SYSCTL_HANDLER_ARGS)
- static int [set_scsi_delay](#) (int delay)
- const char * [scsi_op_desc](#) (u_int16_t opcode, struct [scsi_inquiry_data](#) *inq_data)
- void [scsi_sense_desc](#) (int sense_key, int asc, int ascq, struct [scsi_inquiry_data](#) *inq_data, const char **sense_key_desc, const char **asc_desc)
- [scsi_sense_action](#) [scsi_error_action](#) (struct [ccb_scsiio](#) *csio, struct [scsi_inquiry_data](#) *inq_data, u_int32_t sense_flags)
- char * [scsi_cdb_string](#) (u_int8_t *cdb_ptr, char *cdb_string, size_t len)
- const char * [scsi_status_string](#) (struct [ccb_scsiio](#) *csio)
- int [scsi_command_string](#) (struct [ccb_scsiio](#) *csio, struct sbuf *sb)
- int [scsi_sense_sbuf](#) (struct [ccb_scsiio](#) *csio, struct sbuf *sb, [scsi_sense_string_flags](#) flags)
- char * [scsi_sense_string](#) (struct [ccb_scsiio](#) *csio, char *str, int str_len)
- void [scsi_sense_print](#) (struct [ccb_scsiio](#) *csio)
- void [scsi_print_inquiry](#) (struct [scsi_inquiry_data](#) *inq_data)
- u_int [scsi_calc_syncrate](#) (u_int [period_factor](#))
- u_int [scsi_calc_syncparam](#) (u_int [period](#))
- void [scsi_test_unit_ready](#) (struct [ccb_scsiio](#) *csio, u_int32_t retries, void(*cbfcn)(struct [cam_periph](#) *, union [ccb](#) *), u_int8_t tag_action, u_int8_t sense_len, u_int32_t timeout)
- void [scsi_request_sense](#) (struct [ccb_scsiio](#) *csio, u_int32_t retries, void(*cbfcn)(struct [cam_periph](#) *, union [ccb](#) *), void *data_ptr, u_int8_t dxfer_len, u_int8_t tag_action, u_int8_t sense_len, u_int32_t timeout)
- void [scsi_inquiry](#) (struct [ccb_scsiio](#) *csio, u_int32_t retries, void(*cbfcn)(struct [cam_periph](#) *, union [ccb](#) *), u_int8_t tag_action, u_int8_t *inq_buf, u_int32_t inq_len, int evpd, u_int8_t page_code, u_int8_t sense_len, u_int32_t timeout)
- void [scsi_mode_sense](#) (struct [ccb_scsiio](#) *csio, u_int32_t retries, void(*cbfcn)(struct [cam_periph](#) *, union [ccb](#) *), u_int8_t tag_action, int dbd, u_int8_t page_code, u_int8_t page, u_int8_t *param_buf, u_int32_t param_len, u_int8_t sense_len, u_int32_t timeout)

- void `scsi_mode_sense_len` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, int dbd, u_int8_t page_code, u_int8_t page, u_int8_t *param_buf, u_int32_t param_len, int minimum_cmd_size, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_mode_select` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, int scsi_page_fmt, int save_pages, u_int8_t *param_buf, u_int32_t param_len, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_mode_select_len` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, int scsi_page_fmt, int save_pages, u_int8_t *param_buf, u_int32_t param_len, int minimum_cmd_size, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_log_sense` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, u_int8_t page_code, u_int8_t page, int save_pages, int ppc, u_int32_t paramptr, u_int8_t *param_buf, u_int32_t param_len, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_log_select` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, u_int8_t page_code, int save_pages, int pc_reset, u_int8_t *param_buf, u_int32_t param_len, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_prevent` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, u_int8_t action, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_read_capacity` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, struct `scsi_read_capacity_data` *rcap_buf, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_read_capacity_16` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, u_int64_t lba, int reladr, int pmi, struct `scsi_read_capacity_data_long` *rcap_buf, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_report_luns` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, u_int8_t select_report, struct `scsi_report_luns_data` *rpl_buf, u_int32_t alloc_len, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_synchronize_cache` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, u_int32_t begin_lba, u_int16_t lb_count, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_read_write` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, int readop, u_int8_t byte2, int minimum_cmd_size, u_int64_t lba, u_int32_t block_count, u_int8_t *data_ptr, u_int32_t dxfer_len, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_start_stop` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, int start, int load_eject, int immediate, u_int8_t sense_len, u_int32_t timeout)
- int `scsi_inquiry_match` (caddr_t inqbuffer, caddr_t table_entry)
- int `scsi_static_inquiry_match` (caddr_t inqbuffer, caddr_t table_entry)
- `SYSINIT` (`scsi_delay`, `SI_SUB_TUNABLES`, `SI_ORDER_ANY`, `init_scsi_delay`, `NULL`)
- `SYSCTL_PROC` (`_kern_cam`, `OID_AUTO`, `scsi_delay`, `CTLTYPE_INT|CTLFLAG_RW`, `0`, `0`, `sysctl_scsi_delay`, "I", "Delay to allow devices to settle after a SCSI bus reset (ms)")

Variables

- int `scsi_delay`
- static struct `op_table_entry` `plexor_cd_ops` []
- static struct `scsi_op_quirk_entry` `scsi_op_quirk_table` []
- static struct `op_table_entry` `scsi_op_codes` []
- `sense_key_table_entry` `sense_key_table` []
- const int `sense_key_table_size`
- static struct `asc_table_entry` `quantum_fireball_entries` []

- static struct `asc_table_entry` `sony_mo_entries` []
- static struct `scsi_sense_quirk_entry` `sense_quirk_table` []
- const int `sense_quirk_table_size`
- static struct `asc_table_entry` `asc_table` []
- const int `asc_table_size` = sizeof(`asc_table`)/sizeof(`asc_table`[0])
- struct {
 - u_int `period_factor`
 - u_int `period`
- } `scsi_syncrates` []

7.16.1 Define Documentation

7.16.1.1 #define A 0x400

Definition at line 116 of file `scsi_all.c`.

7.16.1.2 #define ALL 0xFFFF

Definition at line 119 of file `scsi_all.c`.

7.16.1.3 #define C 0x200

Definition at line 115 of file `scsi_all.c`.

7.16.1.4 #define D 0x001

Definition at line 106 of file `scsi_all.c`.

7.16.1.5 #define E 0x800

Definition at line 117 of file `scsi_all.c`.

7.16.1.6 #define L 0x004

Definition at line 108 of file `scsi_all.c`.

Referenced by `cam_calc_geometry()`.

7.16.1.7 #define M 0x100

Definition at line 114 of file `scsi_all.c`.

7.16.1.8 #define O 0x080

Definition at line 113 of file `scsi_all.c`.

7.16.1.9 #define P 0x008

Definition at line 109 of file scsi_all.c.

7.16.1.10 #define R 0x020

Definition at line 111 of file scsi_all.c.

7.16.1.11 #define S 0x040

Definition at line 112 of file scsi_all.c.

7.16.1.12 #define SCSI_DELAY 2000

Definition at line 72 of file scsi_all.c.

Referenced by `init_scsi_delay()`.

7.16.1.13 #define SCSI_MIN_DELAY 100

Definition at line 81 of file scsi_all.c.

Referenced by `set_scsi_delay()`.

7.16.1.14 #define SST(asc, ascq, action, desc) asc, ascq, action, desc

Definition at line 733 of file scsi_all.c.

7.16.1.15 #define T 0x002

Definition at line 107 of file scsi_all.c.

7.16.1.16 #define W 0x010

Definition at line 110 of file scsi_all.c.

7.16.2 Function Documentation**7.16.2.1 __FBSID ("FreeBSD: src/sys/cam/scsi/scsi_all. c, v 1.49 2006/08/21 13:24:50 ken Exp \$")****7.16.2.2 static int ascentrycomp (const void * *key*, const void * *member*) [static]**

Definition at line 1491 of file scsi_all.c.

References `asc_table_entry::action`, `asc_table_entry::asc`, `asc_table_entry::ascq`, and `SSQ_RANGE`.

Referenced by `fetchtableentries()`.

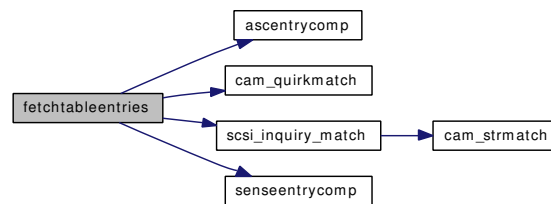
7.16.2.3 static void fetchtableentries (int *sense_key*, int *asc*, int *ascq*, struct *scsi_inquiry_data* *, const struct *sense_key_table_entry* **, const struct *asc_table_entry* **) [static]

Definition at line 1537 of file scsi_all.c.

References *asc_key::asc*, *scsi_sense_quirk_entry::asc_info*, *ascentrycomp()*, *asc_key::ascq*, *cam_quirkmatch()*, *scsi_sense_quirk_entry::num_ascs*, *scsi_sense_quirk_entry::num_sense_keys*, *scsi_inquiry_match()*, *scsi_sense_quirk_entry::sense_key_info*, and *senseentrycomp()*.

Referenced by *scsi_error_action()*, and *scsi_sense_desc()*.

Here is the call graph for this function:



7.16.2.4 static void init_scsi_delay (void) [static]

Definition at line 3008 of file scsi_all.c.

References *SCSI_DELAY*, and *set_scsi_delay()*.

Here is the call graph for this function:



7.16.2.5 u_int scsi_calc_syncparam (u_int *period*)

Definition at line 2346 of file scsi_all.c.

References *period_factor*, and *scsi_syncrates*.

7.16.2.6 u_int scsi_calc_syncrate (u_int *period_factor*)

Definition at line 2310 of file scsi_all.c.

References *period*, and *scsi_syncrates*.

Referenced by *xpt_announce_periph()*.

7.16.2.7 char* scsi_cdb_string (u_int8_t * *cdb_ptr*, char * *cdb_string*, size_t *len*)

Definition at line 1734 of file scsi_all.c.

Referenced by *scsi_command_string()*, and *xpt_action()*.

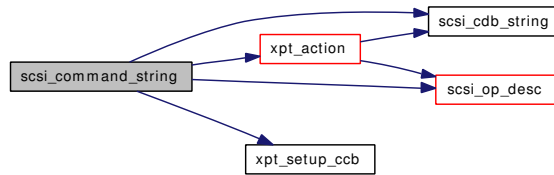
7.16.2.8 `int scsi_command_string (struct ccb_scsiio * csio, struct sbuf * sb)`

Definition at line 1826 of file `scsi_all.c`.

References `CAM_CDB_POINTER`, `CAM_DEV_NOT_THERE`, `ccb_getdev::ccb_h`, `ccb_scsiio::ccb_h`, `ccb::ccb_h`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `cdb_t::cdb_ptr`, `ccb_hdr::flags`, `ccb_hdr::func_code`, `ccb_getdev::inq_data`, `ccb_hdr::path`, `scsi_cdb_string()`, `SCSI_MAX_CDBLEN`, `scsi_op_desc()`, `ccb_hdr::status`, `T_DIRECT`, `xpt_action()`, `XPT_GDEV_TYPE`, and `xpt_setup_ccb()`.

Referenced by `cam_error_string()`, and `scsi_sense_sbuf()`.

Here is the call graph for this function:



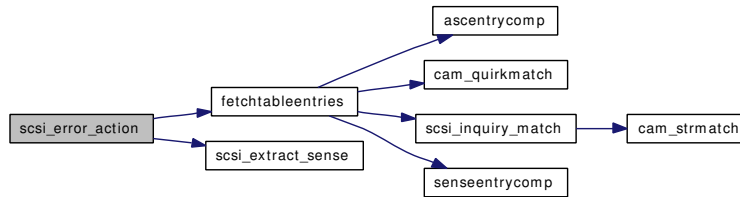
7.16.2.9 `scsi_sense_action scsi_error_action (struct ccb_scsiio * csio, struct scsi_inquiry_data * inq_data, u_int32_t sense_flags)`

Definition at line 1649 of file `scsi_all.c`.

References `sense_key_table_entry::action`, `asc_table_entry::action`, `fetchtableentries()`, `scsi_extract_sense()`, `ccb_scsiio::sense_data`, `sense_key_table_entry::sense_key`, `SF_NO_PRINT`, `SF_PRINT_ALWAYS`, `SF_QUIET_IR`, `SF_RETRY_UA`, `SS_ERRMASK`, `SS_FAIL`, `SS_MASK`, `SS_NOP`, `SS_RETRY`, `SSD_DEFERRED_ERROR`, `SSD_KEY_ILLEGAL_REQUEST`, `SSD_KEY_RECOVERED_ERROR`, `SSD_KEY_UNIT_ATTENTION`, `SSQ_DECREMENT_COUNT`, `SSQ_MASK`, and `SSQ_PRINT_SENSE`.

Referenced by `camperiphdone()`, and `camperiphscsisenseerror()`.

Here is the call graph for this function:



7.16.2.10 `void scsi_inquiry (struct ccb_scsiio * csio, u_int32_t retries, void(*) (struct cam_periph *, union ccb *) cbfcnp, u_int8_t tag_action, u_int8_t * inq_buf, u_int32_t inq_len, int evpd, u_int8_t page_code, u_int8_t sense_len, u_int32_t timeout)`

Definition at line 2422 of file `scsi_all.c`.

References `CAM_DIR_IN`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `INQUIRY`, and `SI_EVPD`.

Referenced by probestart().

Here is the call graph for this function:



7.16.2.11 int scsi_inquiry_match (caddr_t inqbuffer, caddr_t table_entry)

Definition at line 2957 of file scsi_all.c.

References cam_strmatch(), scsi_inquiry_pattern::media_type, scsi_inquiry_pattern::product, scsi_inquiry_pattern::revision, SID_IS_REMOVABLE, SID_TYPE, SIP_MEDIA_FIXED, SIP_MEDIA_REMOVABLE, T_ANY, scsi_inquiry_pattern::type, and scsi_inquiry_pattern::vendor.

Referenced by cdregister(), daregister(), fetchtableentries(), saregister(), scsi_op_desc(), and xpt_find_quirk().

Here is the call graph for this function:

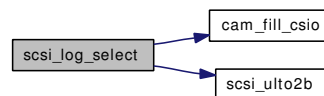


7.16.2.12 void scsi_log_select (struct ccb_scsiio * csio, u_int32_t retries, void(*) (struct cam_periph *, union ccb *) cbfnp, u_int8_t tag_action, u_int8_t page_code, int save_pages, int pc_reset, u_int8_t * param_buf, u_int32_t param_len, u_int8_t sense_len, u_int32_t timeout)

Definition at line 2631 of file scsi_all.c.

References CAM_DIR_OUT, cam_fill_csio(), cdb_t::cdb_bytes, ccb_scsiio::cdb_io, LOG_SELECT, scsi_ulito2b(), SLS_PAGE_CODE, SLS_PCR, and SLS_SP.

Here is the call graph for this function:

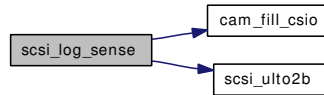


7.16.2.13 void scsi_log_sense (struct ccb_scsiio * csio, u_int32_t retries, void(*) (struct cam_periph *, union ccb *) cbfnp, u_int8_t tag_action, u_int8_t page_code, u_int8_t page, int save_pages, int ppc, u_int32_t paramptr, u_int8_t * param_buf, u_int32_t param_len, u_int8_t sense_len, u_int32_t timeout)

Definition at line 2596 of file scsi_all.c.

References CAM_DIR_IN, cam_fill_csio(), cdb_t::cdb_bytes, ccb_scsiio::cdb_io, LOG_SENSE, scsi_ulito2b(), SLS_PPC, and SLS_SP.

Here is the call graph for this function:



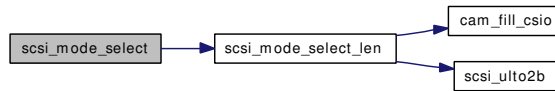
7.16.2.14 void `scsi_mode_select` (struct `ccb_scsiio` * *csio*, `u_int32_t` *retries*, void(*) (struct `cam_periph` *, union `ccb` *) *cbfcnp*, `u_int8_t` *tag_action*, `int` *scsi_page_fmt*, `int` *save_pages*, `u_int8_t` * *param_buf*, `u_int32_t` *param_len*, `u_int8_t` *sense_len*, `u_int32_t` *timeout*)

Definition at line 2526 of file `scsi_all.c`.

References `scsi_mode_select_len()`.

Referenced by `sasetparams()`.

Here is the call graph for this function:



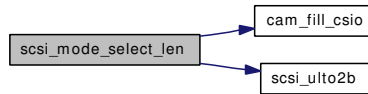
7.16.2.15 void `scsi_mode_select_len` (struct `ccb_scsiio` * *csio*, `u_int32_t` *retries*, void(*) (struct `cam_periph` *, union `ccb` *) *cbfcnp*, `u_int8_t` *tag_action*, `int` *scsi_page_fmt*, `int` *save_pages*, `u_int8_t` * *param_buf*, `u_int32_t` *param_len*, `int` *minimum_cmd_size*, `u_int8_t` *sense_len*, `u_int32_t` *timeout*)

Definition at line 2538 of file `scsi_all.c`.

References `CAM_DIR_OUT`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `MODE_SELECT_10`, `MODE_SELECT_6`, `scsi_ulto2b()`, `SMS_PF`, and `SMS_SP`.

Referenced by `cdsetmode()`, and `scsi_mode_select()`.

Here is the call graph for this function:



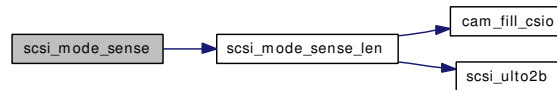
7.16.2.16 void `scsi_mode_sense` (struct `ccb_scsiio` * *csio*, `u_int32_t` *retries*, void(*) (struct `cam_periph` *, union `ccb` *) *cbfcnp*, `u_int8_t` *tag_action*, `int` *dbd*, `u_int8_t` *page_code*, `u_int8_t` *page*, `u_int8_t` * *param_buf*, `u_int32_t` *param_len*, `u_int8_t` *sense_len*, `u_int32_t` *timeout*)

Definition at line 2459 of file `scsi_all.c`.

References `scsi_mode_sense_len()`.

Referenced by `chgetparams()`, `chstart()`, `probstart()`, and `sagetparams()`.

Here is the call graph for this function:



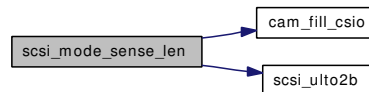
7.16.2.17 void `scsi_mode_sense_len` (struct `ccb_scsiio` * *csio*, `u_int32_t` *retries*, void(*) (struct `cam_periph` *, union `ccb` *) *cbfcnp*, `u_int8_t` *tag_action*, `int` *dbd*, `u_int8_t` *page_code*, `u_int8_t` *page*, `u_int8_t` * *param_buf*, `u_int32_t` *param_len*, `int` *minimum_cmd_size*, `u_int8_t` *sense_len*, `u_int32_t` *timeout*)

Definition at line 2472 of file `scsi_all.c`.

References `CAM_DIR_IN`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `MODE_SENSE_10`, `MODE_SENSE_6`, `scsi_ulito2b()`, and `SMS_DBD`.

Referenced by `cdgetmode()`, and `scsi_mode_sense()`.

Here is the call graph for this function:



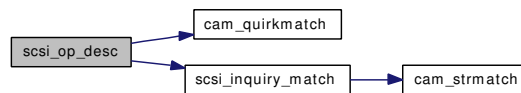
7.16.2.18 const char* `scsi_op_desc` (`u_int16_t` *opcode*, struct `scsi_inquiry_data` * *inq_data*)

Definition at line 659 of file `scsi_all.c`.

References `cam_quirkmatch()`, `op_table_entry::opcode`, `op_table_entry::opmask`, `scsi_inquiry_match()`, `scsi_op_codes`, `scsi_op_quirk_table`, `SID_TYPE`, `T_DIRECT`, and `T_RBC`.

Referenced by `scsi_command_string()`, and `xpt_action()`.

Here is the call graph for this function:



7.16.2.19 void `scsi_prevent` (struct `ccb_scsiio` * *csio*, `u_int32_t` *retries*, void(*) (struct `cam_periph` *, union `ccb` *) *cbfcnp*, `u_int8_t` *tag_action*, `u_int8_t` *action*, `u_int8_t` *sense_len*, `u_int32_t` *timeout*)

Definition at line 2667 of file `scsi_all.c`.

References `CAM_DIR_NONE`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, and `PREVENT_ALLOW`.

Referenced by `cdprevent()`, `daprevent()`, and `saprevent()`.

Here is the call graph for this function:



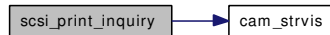
7.16.2.20 void `scsi_print_inquiry` (struct `scsi_inquiry_data` * `inq_data`)

Definition at line 2190 of file `scsi_all.c`.

References `cam_strvis()`, `scsi_inquiry_data::product`, `scsi_inquiry_data::revision`, `SCSI_REV_CCS`, `SID_ANSI_REV`, `SID_IS_REMOVABLE`, `SID_QUAL`, `SID_QUAL_BAD_LU`, `SID_QUAL_IS_VENDOR_UNIQUE`, `SID_QUAL_LU_CONNECTED`, `SID_QUAL_LU_OFFLINE`, `SID_QUAL_RSVD`, `SID_TYPE`, `T_CDROM`, `T_CHANGER`, `T_COMM`, `T_DIRECT`, `T_ENCLOSURE`, `T_NODEVICE`, `T_OCRW`, `T_OPTICAL`, `T_PRINTER`, `T_PROCESSOR`, `T_RBC`, `T_SCANNER`, `T_SEQUENTIAL`, `T_STORARRAY`, `T_WORM`, and `scsi_inquiry_data::vendor`.

Referenced by `xpt_announce_periph()`.

Here is the call graph for this function:



7.16.2.21 void `scsi_read_capacity` (struct `ccb_scsiio` * `csio`, `u_int32_t` `retries`, void(*) (struct `cam_periph` *, union `ccb` *) `cbfcnp`, `u_int8_t` `tag_action`, struct `scsi_read_capacity_data` * `rcap_buf`, `u_int8_t` `sense_len`, `u_int32_t` `timeout`)

Definition at line 2693 of file `scsi_all.c`.

References `CAM_DIR_IN`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, and `READ_CAPACITY`.

Referenced by `cdsize()`, `cdstart()`, `dagetcapacity()`, and `dastart()`.

Here is the call graph for this function:



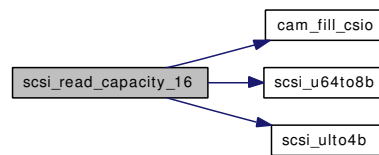
7.16.2.22 void `scsi_read_capacity_16` (struct `ccb_scsiio` * `csio`, `uint32_t` `retries`, void(*) (struct `cam_periph` *, union `ccb` *) `cbfcnp`, `uint8_t` `tag_action`, `uint64_t` `lba`, `int` `reladr`, `int` `pmi`, struct `scsi_read_capacity_data_long` * `rcap_buf`, `uint8_t` `sense_len`, `uint32_t` `timeout`)

Definition at line 2718 of file `scsi_all.c`.

References `CAM_DIR_IN`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `scsi_u64to8b()`, `scsi_ulto4b()`, `SERVICE_ACTION_IN`, `SRC16_PMI`, `SRC16_RELADR`, and `SRC16_SERVICE_ACTION`.

Referenced by `dagetcapacity()`, and `dastart()`.

Here is the call graph for this function:



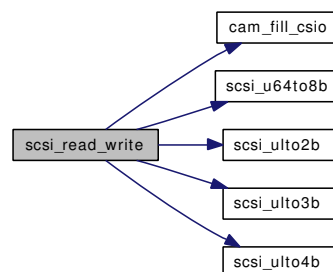
7.16.2.23 void `scsi_read_write` (struct `ccb_scsiio` * *csio*, u_int32_t *retries*, void(*) (struct `cam_periph` *, union `ccb` *) *cbfcnp*, u_int8_t *tag_action*, int *readop*, u_int8_t *byte2*, int *minimum_cmd_size*, u_int64_t *lba*, u_int32_t *block_count*, u_int8_t * *data_ptr*, u_int32_t *dxfer_len*, u_int8_t *sense_len*, u_int32_t *timeout*)

Definition at line 2808 of file `scsi_all.c`.

References `scsi_rw_6::addr`, `scsi_rw_10::addr`, `scsi_rw_12::addr`, `scsi_rw_16::addr`, `scsi_rw_10::byte2`, `scsi_rw_12::byte2`, `scsi_rw_16::byte2`, `CAM_DEBUG`, `CAM_DEBUG_SUBTRACE`, `CAM_DIR_IN`, `CAM_DIR_OUT`, `cam_fill_csio()`, `ccb_scsiio::ccb_h`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `scsi_rw_6::control`, `scsi_rw_10::control`, `scsi_rw_12::control`, `scsi_rw_16::control`, `scsi_rw_6::length`, `scsi_rw_10::length`, `scsi_rw_12::length`, `scsi_rw_16::length`, `scsi_rw_6::opcode`, `scsi_rw_10::opcode`, `scsi_rw_12::opcode`, `scsi_rw_16::opcode`, `ccb_hdr::path`, `READ_10`, `READ_12`, `READ_16`, `READ_6`, `scsi_rw_10::reserved`, `scsi_rw_12::reserved`, `scsi_rw_16::reserved`, `scsi_u64to8b()`, `scsi_ulto2b()`, `scsi_ulto3b()`, `scsi_ulto4b()`, `WRITE_10`, `WRITE_12`, `WRITE_16`, and `WRITE_6`.

Referenced by `cdstart()`, `dadump()`, and `dastart()`.

Here is the call graph for this function:

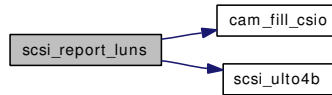


7.16.2.24 void `scsi_report_luns` (struct `ccb_scsiio` * *csio*, u_int32_t *retries*, void(*) (struct `cam_periph` *, union `ccb` *) *cbfcnp*, u_int8_t *tag_action*, u_int8_t *select_report*, struct `scsi_report_luns_data` * *rpl_buf*, u_int32_t *alloc_len*, u_int8_t *sense_len*, u_int32_t *timeout*)

Definition at line 2750 of file `scsi_all.c`.

References `CAM_DIR_IN`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `REPORT_LUNS`, and `scsi_ulto4b()`.

Here is the call graph for this function:



7.16.2.25 void `scsi_request_sense` (struct `ccb_scsiio` * *csio*, u_int32_t *retries*, void(*) (struct `cam_periph` *, union `ccb` *) *cbfcn*, void * *data_ptr*, u_int8_t *dxfer_len*, u_int8_t *tag_action*, u_int8_t *sense_len*, u_int32_t *timeout*)

Definition at line 2397 of file `scsi_all.c`.

References `CAM_DIR_IN`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, and `REQUEST_SENSE`.

Referenced by `camperiphscsisenseerror()`.

Here is the call graph for this function:



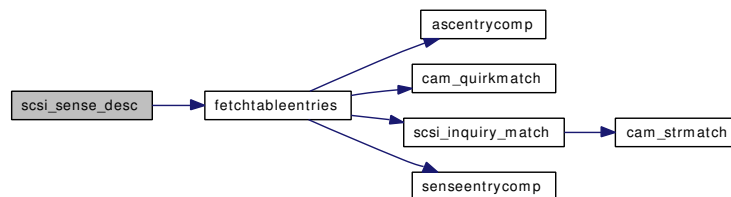
7.16.2.26 void `scsi_sense_desc` (int *sense_key*, int *asc*, int *ascq*, struct `scsi_inquiry_data` * *inq_data*, const char ** *sense_key_desc*, const char ** *asc_desc*)

Definition at line 1619 of file `scsi_all.c`.

References `sense_key_table_entry::desc`, `asc_table_entry::desc`, and `fetchtableentries()`.

Referenced by `cddone()`, `dadone()`, and `scsi_sense_sbuf()`.

Here is the call graph for this function:



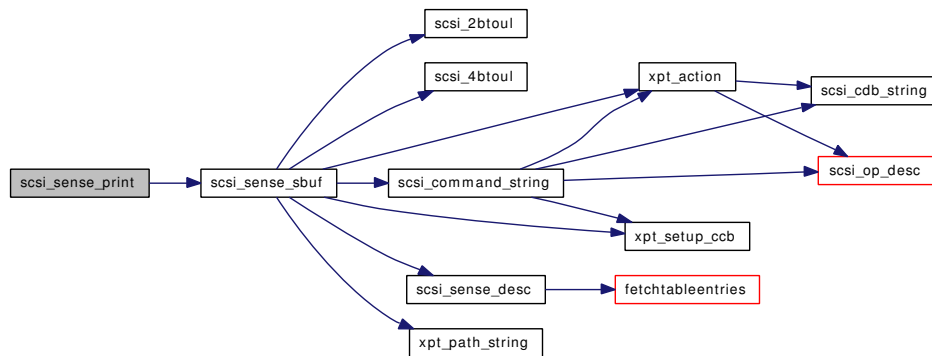
7.16.2.27 void `scsi_sense_print` (struct `ccb_scsiio` * *csio*)

Definition at line 2146 of file `scsi_all.c`.

References `scsi_sense_sbuf()`, and `SSS_FLAG_PRINT_COMMAND`.

Referenced by `camperiphdone()`, `camperiphscsisenseerror()`, `cddone()`, `chdone()`, `chgetparams()`, `daclose()`, `dadone()`, `dadump()`, `dashutdown()`, and `sagetparams()`.

Here is the call graph for this function:



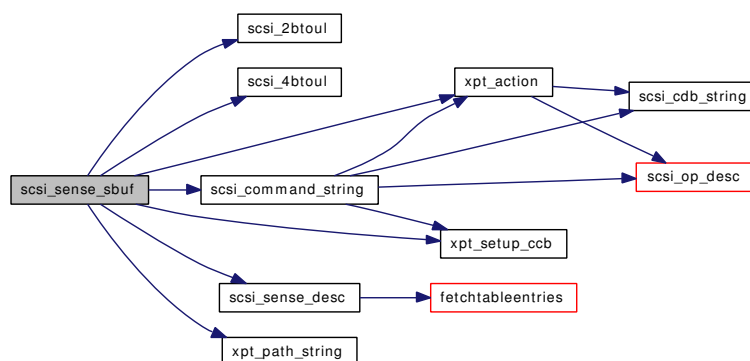
7.16.2.28 int scsi_sense_sbuf (struct [ccb_scsiio](#) * *csio*, struct sbuf * *sb*, [scsi_sense_string_flags](#) *flags*)

Definition at line 1885 of file `scsi_all.c`.

References `scsi_sense_data::add_sense_code`, `scsi_sense_data::add_sense_code_qual`, `CAM_CDB_PHYS`, `CAM_DEV_NOT_THERE`, `CAM_SENSE_PHYS`, `CAM_SENSE_PTR`, `ccb_scsiio::ccb_h`, `ccb_getdev::ccb_h`, `ccb::ccb_h`, `scsi_sense_data::cmd_spec_info`, `scsi_sense_data::error_code`, `scsi_sense_data::extra_len`, `ccb_hdr::flags`, `scsi_sense_data::flags`, `scsi_sense_data::fru`, `ccb_hdr::func_code`, `scsi_sense_data::info`, `ccb_getdev::inq_data`, `ccb_hdr::path`, `scsi_2btoul()`, `scsi_4btoul()`, `scsi_command_string()`, `scsi_sense_desc()`, `ccb_scsiio::sense_data`, `scsi_sense_data::sense_key_spec`, `SSD_CURRENT_ERROR`, `SSD_DEFERRED_ERROR`, `SSD_ERRCODE`, `SSD_ERRCODE_VALID`, `SSD_ILI`, `SSD_KEY`, `SSD_KEY_BLANK_CHECK`, `SSD_KEY_DATA_PROTECT`, `SSD_KEY_HARDWARE_ERROR`, `SSD_KEY_ILLEGAL_REQUEST`, `SSD_KEY_MEDIUM_ERROR`, `SSD_KEY_NOT_READY`, `SSD_KEY_RECOVERED_ERROR`, `SSD_KEY_UNIT_ATTENTION`, `SSD_SCS_VALID`, `SSS_FLAG_PRINT_COMMAND`, `ccb_hdr::status`, `T_DIRECT`, `xpt_action()`, `XPT_GDEV_TYPE`, `xpt_path_string()`, and `xpt_setup_ccb()`.

Referenced by `cam_error_string()`, `scsi_sense_print()`, and `scsi_sense_string()`.

Here is the call graph for this function:

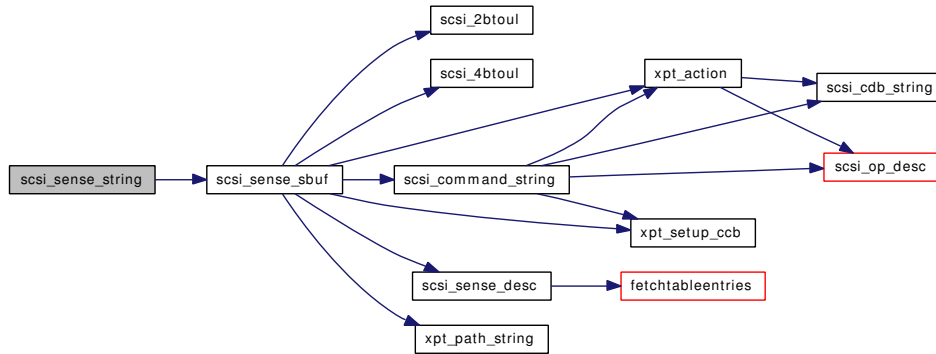


7.16.2.29 char* scsi_sense_string (struct [ccb_scsiio](#) * *csio*, char * *str*, int *str_len*)

Definition at line 2122 of file `scsi_all.c`.

References `scsi_sense_sbuf()`, and `SSS_FLAG_PRINT_COMMAND`.

Here is the call graph for this function:



7.16.2.30 `void scsi_start_stop (struct ccb_scsiio * csio, u_int32_t retries, void(*) (struct cam_periph *, union ccb *) cbfcn, u_int8_t tag_action, int start, int load_eject, int immediate, u_int8_t sense_len, u_int32_t timeout)`

Definition at line 2917 of file `scsi_all.c`.

References `CAM_DIR_NONE`, `cam_fill_csio()`, `CAM_HIGH_POWER`, `ccb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `SSS_IMMED`, `SSS_LOEJ`, `SSS_START`, and `START_STOP_UNIT`.

Referenced by `camperiphscsisenseerror()`, `cdstartunit()`, and `cdstopunit()`.

Here is the call graph for this function:



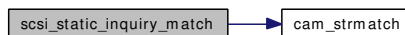
7.16.2.31 `int scsi_static_inquiry_match (caddr_t inqbuffer, caddr_t table_entry)`

Definition at line 2984 of file `scsi_all.c`.

References `cam_strmatch()`, `scsi_static_inquiry_pattern::media_type`, `scsi_static_inquiry_pattern::product`, `scsi_static_inquiry_pattern::revision`, `SID_IS_REMOVABLE`, `SID_TYPE`, `SIP_MEDIA_FIXED`, `SIP_MEDIA_REMOVABLE`, `T_ANY`, `scsi_static_inquiry_pattern::type`, and `scsi_static_inquiry_pattern::vendor`.

Referenced by `xptdevicematch()`.

Here is the call graph for this function:



7.16.2.32 `const char* scsi_status_string (struct ccb_scsiio * csio)`

Definition at line 1789 of file `scsi_all.c`.

References `ccb_scsiio::scsi_status`, `SCSI_STATUS_ACA_ACTIVE`, `SCSI_STATUS_BUSY`, `SCSI_STATUS_CHECK_COND`, `SCSI_STATUS_CMD_TERMINATED`, `SCSI_STATUS_INTERMED`, `SCSI_STATUS_INTERMED_COND_MET`, `SCSI_STATUS_OK`, `SCSI_STATUS_QUEUE_FULL`, `SCSI_STATUS_RESERV_CONFLICT`, and `SCSI_STATUS_TASK_ABORTED`.

Referenced by `cam_error_string()`, and `cddone()`.

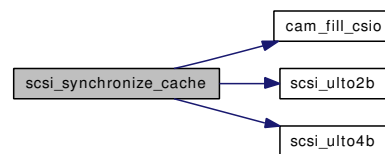
7.16.2.33 `void scsi_synchronize_cache (struct ccb_scsiio * csio, u_int32_t retries, void(*)(struct cam_periph *, union ccb *) cbfcnp, u_int8_t tag_action, u_int32_t begin_lba, u_int16_t lb_count, u_int8_t sense_len, u_int32_t timeout)`

Definition at line 2781 of file `scsi_all.c`.

References `CAM_DIR_NONE`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `scsi_ulto2b()`, `scsi_ulto4b()`, and `SYNCHRONIZE_CACHE`.

Referenced by `daclose()`, `dadump()`, `dashutdown()`, and `dastart()`.

Here is the call graph for this function:



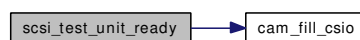
7.16.2.34 `void scsi_test_unit_ready (struct ccb_scsiio * csio, u_int32_t retries, void(*)(struct cam_periph *, union ccb *) cbfcnp, u_int8_t tag_action, u_int8_t sense_len, u_int32_t timeout)`

Definition at line 2374 of file `scsi_all.c`.

References `CAM_DIR_NONE`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, and `TEST_UNIT_READY`.

Referenced by `camperiphscsisenseerror()`, `probstart()`, and `samount()`.

Here is the call graph for this function:



7.16.2.35 `static int senseentrycomp (const void * key, const void * member)` `[static]`

Definition at line 1520 of file `scsi_all.c`.

References `sense_key_table_entry::sense_key`.

Referenced by `fetchtableentries()`.

7.16.2.36 `static int set_scsi_delay (int delay)` `[static]`

Definition at line 3038 of file `scsi_all.c`.

References `scsi_delay`, and `SCSI_MIN_DELAY`.

Referenced by `init_scsi_delay()`, and `sysctl_scsi_delay()`.

7.16.2.37 `SYSCTL_PROC` (`_kern_cam`, `OID_AUTO`, `scsi_delay`, `CTLTYPE_INT|CTLFLAG_RW`, `0`, `0`, `sysctl_scsi_delay`, "I", "Delay to allow devices to settle after a SCSI bus reset (ms)")

7.16.2.38 `static int sysctl_scsi_delay` (`SYSCTL_HANDLER_ARGS`) [`static`]

Definition at line 3023 of file `scsi_all.c`.

References `scsi_delay`, and `set_scsi_delay()`.

Here is the call graph for this function:



7.16.2.39 `SYSINIT` (`scsi_delay`, `SI_SUB_TUNABLES`, `SI_ORDER_ANY`, `init_scsi_delay`, `NULL`)

7.16.3 Variable Documentation

7.16.3.1 `struct asc_table_entry asc_table` [`static`]

Definition at line 808 of file `scsi_all.c`.

7.16.3.2 `const int asc_table_size = sizeof(asc_table)/sizeof(asc_table[0])`

Definition at line 1482 of file `scsi_all.c`.

7.16.3.3 `u_int period`

Definition at line 2296 of file `scsi_all.c`.

Referenced by `scsi_calc_syncsrates()`, `scsi_low_calcf_target()`, and `scsi_low_synch()`.

7.16.3.4 `u_int period_factor`

Definition at line 2295 of file `scsi_all.c`.

Referenced by `scsi_calc_syncparam()`.

7.16.3.5 `struct op_table_entry plextor_cd_ops` [`static`]

Initial value:

```

{
    {0xD8, R, "CD-DA READ"}
}
  
```

Definition at line 121 of file `scsi_all.c`.

7.16.3.6 struct asc_table_entry quantum_fireball_entries[] [static]**Initial value:**

```
{
    {SST(0x04, 0x0b, SS_START|SSQ_DECREMENT_COUNT|ENXIO,
      "Logical unit not ready, initializing cmd. required")}
}
```

Definition at line 768 of file scsi_all.c.

7.16.3.7 int scsi_delay

Definition at line 90 of file scsi_all.c.

Referenced by cam_periph_async(), proberegister(), set_scsi_delay(), sysctl_scsi_delay(), and xpt_config().

7.16.3.8 struct op_table_entry scsi_op_codes[] [static]

Definition at line 142 of file scsi_all.c.

Referenced by scsi_op_desc().

7.16.3.9 struct scsi_op_quirk_entry scsi_op_quirk_table[] [static]**Initial value:**

```
{
    {
        {T_CDROM, SIP_MEDIA_REMOVABLE, "PLEXTOR", "CD-ROM PX*", "*"},
        sizeof(plextor_cd_ops)/sizeof(struct op_table_entry),
        plextor_cd_ops
    }
}
```

Definition at line 125 of file scsi_all.c.

Referenced by scsi_op_desc().

7.16.3.10 struct { ... } scsi_syncrates[] [static]

Referenced by scsi_calc_syncparam(), and scsi_calc_syncsrate().

7.16.3.11 struct sense_key_table_entry sense_key_table[]**Initial value:**

```
{
    { SSD_KEY_NO_SENSE, SS_NOP, "NO SENSE" },
    { SSD_KEY_RECOVERED_ERROR, SS_NOP|SSQ_PRINT_SENSE, "RECOVERED ERROR" },
    {
        SSD_KEY_NOT_READY, SS_TUR|SSQ_MANY|SSQ_DECREMENT_COUNT|EBUSY,
```

```

    "NOT READY"
},
{ SSD_KEY_MEDIUM_ERROR, SS_RDEF, "MEDIUM ERROR" },
{ SSD_KEY_HARDWARE_ERROR, SS_RDEF, "HARDWARE FAILURE" },
{ SSD_KEY_ILLEGAL_REQUEST, SS_FATAL|EINVAL, "ILLEGAL REQUEST" },
{ SSD_KEY_UNIT_ATTENTION, SS_FATAL|ENXIO, "UNIT ATTENTION" },
{ SSD_KEY_DATA_PROTECT, SS_FATAL|EACCES, "DATA PROTECT" },
{ SSD_KEY_BLANK_CHECK, SS_FATAL|ENOSPC, "BLANK CHECK" },
{ SSD_KEY_Vendor_Specific, SS_FATAL|EIO, "Vendor Specific" },
{ SSD_KEY_COPY_ABORTED, SS_FATAL|EIO, "COPY ABORTED" },
{ SSD_KEY_ABORTED_COMMAND, SS_RDEF, "ABORTED COMMAND" },
{ SSD_KEY_EQUAL, SS_NOP, "EQUAL" },
{ SSD_KEY_VOLUME_OVERFLOW, SS_FATAL|EIO, "VOLUME OVERFLOW" },
{ SSD_KEY_MISCOMPARE, SS_NOP, "MISCOMPARE" },
{ SSD_KEY_RESERVED, SS_FATAL|EIO, "RESERVED" }
}

```

Definition at line 742 of file `scsi_all.c`.

7.16.3.12 `const int sense_key_table_size`

Initial value:

```
sizeof(sense_key_table)/sizeof(sense_key_table[0])
```

Definition at line 765 of file `scsi_all.c`.

7.16.3.13 `struct scsi_sense_quirk_entry sense_quirk_table[]` [static]

Initial value:

```

{
    {
        {T_DIRECT, SIP_MEDIA_FIXED, "QUANTUM", "FIREBALL S*", "*"},
        0,
        sizeof(quantum_fireball_entries)/sizeof(struct asc_table_entry),
        NULL,
        quantum_fireball_entries
    },
    {
        {T_DIRECT, SIP_MEDIA_REMOVABLE, "SONY", "SMO-*", "*"},
        0,
        sizeof(sony_mo_entries)/sizeof(struct asc_table_entry),
        NULL,
        sony_mo_entries
    }
}

```

Definition at line 778 of file `scsi_all.c`.

7.16.3.14 `const int sense_quirk_table_size`

Initial value:

```
sizeof(sense_quirk_table)/sizeof(sense_quirk_table[0])
```

Definition at line 805 of file `scsi_all.c`.

7.16.3.15 struct `asc_table_entry` `sony_mo_entries[]` [static]**Initial value:**

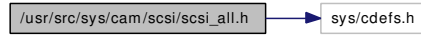
```
{
    {SST(0x04, 0x00, SS_START|SSQ_DECREMENT_COUNT|ENXIO,
      "Logical unit not ready, cause not reportable")}
```

Definition at line 773 of file `scsi_all.c`.

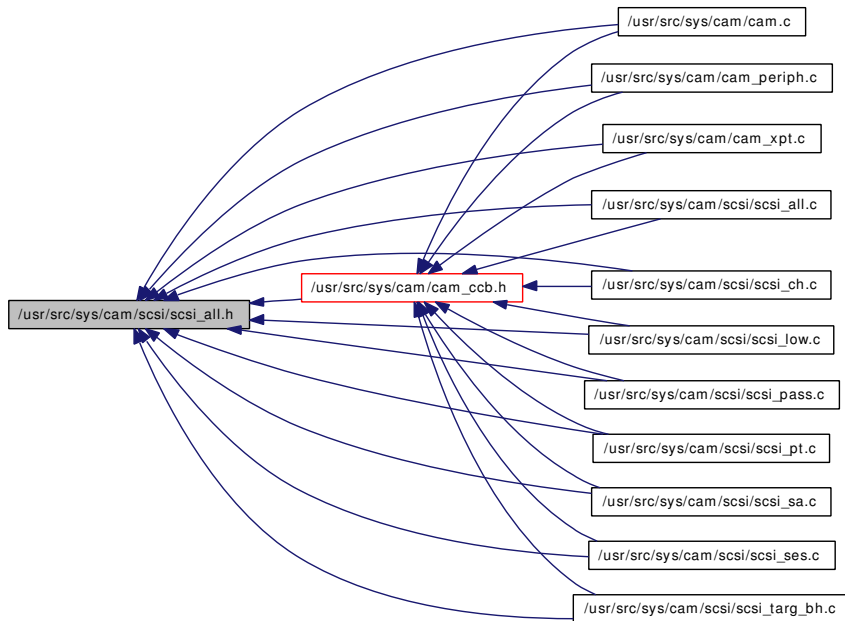
7.17 /usr/src/sys/cam/scsi/scsi_all.h File Reference

```
#include <sys/cdefs.h>
```

Include dependency graph for scsi_all.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- struct [scsi_generic](#)
- struct [scsi_request_sense](#)
- struct [scsi_test_unit_ready](#)
- struct [scsi_send_diag](#)
- struct [scsi_sense](#)
- struct [scsi_inquiry](#)
- struct [scsi_mode_sense_6](#)
- struct [scsi_mode_sense_10](#)
- struct [scsi_mode_select_6](#)
- struct [scsi_mode_select_10](#)
- struct [scsi_mode_hdr_6](#)
- struct [scsi_mode_hdr_10](#)
- struct [scsi_mode_block_descr](#)
- struct [scsi_log_sense](#)
- struct [scsi_log_select](#)
- struct [scsi_log_header](#)

- struct [scsi_log_param_header](#)
- struct [scsi_control_page](#)
- struct [scsi_reserve](#)
- struct [scsi_release](#)
- struct [scsi_prevent](#)
- struct [scsi_sync_cache](#)
- struct [scsi_changedef](#)
- struct [scsi_read_buffer](#)
- struct [scsi_write_buffer](#)
- struct [scsi_rw_6](#)
- struct [scsi_rw_10](#)
- struct [scsi_rw_12](#)
- struct [scsi_rw_16](#)
- struct [scsi_start_stop_unit](#)
- struct [scsi_inquiry_data](#)
- struct [scsi_vpd_unit_serial_number](#)
- struct [scsi_read_capacity](#)
- struct [scsi_read_capacity_16](#)
- struct [scsi_read_capacity_data](#)
- struct [scsi_read_capacity_data_long](#)
- struct [scsi_report_luns](#)
- struct [scsi_report_luns_data](#)
- struct [scsi_sense_data](#)
- struct [scsi_mode_header_6](#)
- struct [scsi_mode_header_10](#)
- struct [scsi_mode_page_header](#)
- struct [scsi_mode_blk_desc](#)
- struct [scsi_inquiry_pattern](#)
- struct [scsi_static_inquiry_pattern](#)
- struct [scsi_sense_quirk_entry](#)
- struct [sense_key_table_entry](#)
- struct [asc_table_entry](#)
- struct [op_table_entry](#)
- struct [scsi_op_quirk_entry](#)

Defines

- #define [_SCSI_SCSI_ALL_H](#) 1
- #define [SCSI_CTL_LINK](#) 0x01
- #define [SCSI_CTL_FLAG](#) 0x02
- #define [SCSI_CTL_VENDOR](#) 0xC0
- #define [SCSI_CMD_LUN](#) 0xA0
- #define [SCSI_CMD_LUN_SHIFT](#) 5
- #define [SCSI_MAX_CDBLEN](#) 16
- #define [SCSI_CDB6_LEN](#)(len) ((len) == 0 ? 256 : len)
- #define [SS_ERRMASK](#) 0xff
- #define [SS_RDEF](#) SS_RETRY|SSQ_DECREMENT_COUNT|SSQ_PRINT_SENSE|EIO
- #define [SS_RET](#) SS_RETRY|SSQ_DECREMENT_COUNT|SSQ_PRINT_SENSE
- #define [SS_FATAL](#) SS_FAIL|SSQ_PRINT_SENSE
- #define [SSD_UOL](#) 0x01

- #define `SSD_DOL` 0x02
- #define `SSD_SELFTEST` 0x04
- #define `SSD_PF` 0x10
- #define `SI_EVPD` 0x01
- #define `SMS_DBD` 0x08
- #define `SMS_PAGE_CODE` 0x3F
- #define `SMS_VENDOR_SPECIFIC_PAGE` 0x00
- #define `SMS_DISCONNECT_RECONNECT_PAGE` 0x02
- #define `SMS_PERIPHERAL_DEVICE_PAGE` 0x09
- #define `SMS_CONTROL_MODE_PAGE` 0x0A
- #define `SMS_ALL_PAGES_PAGE` 0x3F
- #define `SMS_PAGE_CTRL_MASK` 0xC0
- #define `SMS_PAGE_CTRL_CURRENT` 0x00
- #define `SMS_PAGE_CTRL_CHANGEABLE` 0x40
- #define `SMS_PAGE_CTRL_DEFAULT` 0x80
- #define `SMS_PAGE_CTRL_SAVED` 0xC0
- #define `SMS_SP` 0x01
- #define `SMS_PF` 0x10
- #define `SLS_SP` 0x01
- #define `SLS_PPC` 0x02
- #define `SLS_PAGE_CODE` 0x3F
- #define `SLS_ALL_PAGES_PAGE` 0x00
- #define `SLS_OVERRUN_PAGE` 0x01
- #define `SLS_ERROR_WRITE_PAGE` 0x02
- #define `SLS_ERROR_READ_PAGE` 0x03
- #define `SLS_ERROR_READREVERSE_PAGE` 0x04
- #define `SLS_ERROR_VERIFY_PAGE` 0x05
- #define `SLS_ERROR_NONMEDIUM_PAGE` 0x06
- #define `SLS_ERROR_LASTN_PAGE` 0x07
- #define `SLS_PAGE_CTRL_MASK` 0xC0
- #define `SLS_PAGE_CTRL_THRESHOLD` 0x00
- #define `SLS_PAGE_CTRL_CUMULATIVE` 0x40
- #define `SLS_PAGE_CTRL_THRESH_DEFAULT` 0x80
- #define `SLS_PAGE_CTRL_CUMUL_DEFAULT` 0xC0
- #define `SLS_PCR` 0x02
- #define `SLP_LP` 0x01
- #define `SLP_LBIN` 0x02
- #define `SLP_TMC_MASK` 0x0C
- #define `SLP_TMC_ALWAYS` 0x00
- #define `SLP_TMC_EQUAL` 0x04
- #define `SLP_TMC_NOTEQUAL` 0x08
- #define `SLP_TMC_GREATER` 0x0C
- #define `SLP_ETC` 0x10
- #define `SLP_TSD` 0x20
- #define `SLP_DS` 0x40
- #define `SLP_DU` 0x80
- #define `SCB_RLEC` 0x01
- #define `SCP_QUEUE_ALG_MASK` 0xF0
- #define `SCP_QUEUE_ALG_RESTRICTED` 0x00
- #define `SCP_QUEUE_ALG_UNRESTRICTED` 0x10

- #define SCP_QUEUE_ERR 0x02
- #define SCP_QUEUE_DQUE 0x01
- #define SCP_EECA 0x80
- #define SCP_RAENP 0x04
- #define SCP_UAAENP 0x02
- #define SCP_EAENP 0x01
- #define PR_PREVENT 0x01
- #define PR_ALLOW 0x00
- #define RWB_MODE 0x07
- #define RWB_MODE_HDR_DATA 0x00
- #define RWB_MODE_DATA 0x02
- #define RWB_MODE_DOWNLOAD 0x04
- #define RWB_MODE_DOWNLOAD_SAVE 0x05
- #define SRW_TOPADDR 0x1F
- #define SRW10_RELADDR 0x01
- #define SRW10_EBP 0x04
- #define SRW10_FUA 0x08
- #define SRW10_DPO 0x10
- #define SRW12_RELADDR 0x01
- #define SRW12_FUA 0x08
- #define SRW12_DPO 0x10
- #define SRW16_RELADDR 0x01
- #define SRW16_FUA 0x08
- #define SRW16_DPO 0x10
- #define SSS_IMMED 0x01
- #define SSS_START 0x01
- #define SSS_LOEJ 0x02
- #define SC_SCSI_1 0x01
- #define SC_SCSI_2 0x03
- #define TEST_UNIT_READY 0x00
- #define REQUEST_SENSE 0x03
- #define READ_6 0x08
- #define WRITE_6 0x0a
- #define INQUIRY 0x12
- #define MODE_SELECT_6 0x15
- #define MODE_SENSE_6 0x1a
- #define START_STOP_UNIT 0x1b
- #define START_STOP 0x1b
- #define RESERVE 0x16
- #define RELEASE 0x17
- #define RECEIVE_DIAGNOSTIC 0x1c
- #define SEND_DIAGNOSTIC 0x1d
- #define PREVENT_ALLOW 0x1e
- #define READ_CAPACITY 0x25
- #define READ_10 0x28
- #define WRITE_10 0x2a
- #define POSITION_TO_ELEMENT 0x2b
- #define SYNCHRONIZE_CACHE 0x35
- #define WRITE_BUFFER 0x3b
- #define READ_BUFFER 0x3c

- #define CHANGE_DEFINITION 0x40
- #define LOG_SELECT 0x4c
- #define LOG_SENSE 0x4d
- #define MODE_SELECT_10 0x55
- #define MODE_SENSE_10 0x5A
- #define READ_16 0x88
- #define WRITE_16 0x8a
- #define SERVICE_ACTION_IN 0x9e
- #define REPORT_LUNS 0xA0
- #define MOVE_MEDIUM 0xa5
- #define READ_12 0xa8
- #define WRITE_12 0xaa
- #define READ_ELEMENT_STATUS 0xb8
- #define T_DIRECT 0x00
- #define T_SEQUENTIAL 0x01
- #define T_PRINTER 0x02
- #define T_PROCESSOR 0x03
- #define T_WORM 0x04
- #define T_CDROM 0x05
- #define T_SCANNER 0x06
- #define T_OPTICAL 0x07
- #define T_CHANGER 0x08
- #define T_COMM 0x09
- #define T_ASC0 0x0a
- #define T_ASC1 0x0b
- #define T_STORARRAY 0x0c
- #define T_ENCLOSURE 0x0d
- #define T_RBC 0x0e
- #define T_OCRW 0x0f
- #define T_NODEVICE 0x1F
- #define T_ANY 0xFF
- #define T_REMOV 1
- #define T_FIXED 0
- #define SHORT_INQUIRY_LENGTH 36
- #define SID_TYPE(inq_data) ((inq_data) → device & 0x1f)
- #define SID_QUAL(inq_data) (((inq_data) → device & 0xE0) >> 5)
- #define SID_QUAL_LU_CONNECTED 0x00
- #define SID_QUAL_LU_OFFLINE 0x01
- #define SID_QUAL_RSVD 0x02
- #define SID_QUAL_BAD_LU 0x03
- #define SID_QUAL_IS_VENDOR_UNIQUE(inq_data) ((SID_QUAL(inq_data) & 0x08) != 0)
- #define SID_QUAL2 0x7F
- #define SID_IS_REMOVABLE(inq_data) (((inq_data) → dev_qual2 & 0x80) != 0)
- #define SID_ANSI_REV(inq_data) ((inq_data) → version & 0x07)
- #define SCSI_REV_0 0
- #define SCSI_REV_CCS 1
- #define SCSI_REV_2 2
- #define SCSI_REV_SPC 3
- #define SCSI_REV_SPC2 4
- #define SID_ECMA 0x38

- #define [SID_ISO](#) 0xC0
- #define [SID_AENC](#) 0x80
- #define [SID_TrmIOP](#) 0x40
- #define [SID_ADDITIONAL_LENGTH](#)(iqd)
- #define [SPC2_SID_MChngr](#) 0x08
- #define [SPC2_SID_MultiP](#) 0x10
- #define [SPC2_SID_EncServ](#) 0x40
- #define [SPC2_SID_BQueue](#) 0x80
- #define [INQ_DATA_TQ_ENABLED](#)(iqd)
- #define [SID_SftRe](#) 0x01
- #define [SID_CmdQue](#) 0x02
- #define [SID_Linked](#) 0x08
- #define [SID_Sync](#) 0x10
- #define [SID_WBus16](#) 0x20
- #define [SID_WBus32](#) 0x40
- #define [SID_RelAdr](#) 0x80
- #define [SID_VENDOR_SIZE](#) 8
- #define [SID_PRODUCT_SIZE](#) 16
- #define [SID_REVISION_SIZE](#) 4
- #define [SID_VENDOR_SPECIFIC_0_SIZE](#) 20
- #define [SID_SPI_IUS](#) 0x01
- #define [SID_SPI_QAS](#) 0x02
- #define [SID_SPI_CLOCK_ST](#) 0x00
- #define [SID_SPI_CLOCK_DT](#) 0x04
- #define [SID_SPI_CLOCK_DT_ST](#) 0x0C
- #define [SID_SPI_MASK](#) 0x0F
- #define [SID_VENDOR_SPECIFIC_1_SIZE](#) 160
- #define [SVPD_UNIT_SERIAL_NUMBER](#) 0x80
- #define [SVPD_SERIAL_NUM_SIZE](#) 251
- #define [SRC16_SERVICE_ACTION](#) 0x10
- #define [SRC16_PMI](#) 0x01
- #define [SRC16_RELADR](#) 0x02
- #define [RPL_REPORT_DEFAULT](#) 0x00
- #define [RPL_REPORT_WELLKNOWN](#) 0x01
- #define [RPL_REPORT_ALL](#) 0x02
- #define [RPL_LUNDATA_PERIPH_BUS_MASK](#) 0x3f
- #define [RPL_LUNDATA_FLAT_LUN_MASK](#) 0x3f
- #define [RPL_LUNDATA_LUN_TARG_MASK](#) 0x3f
- #define [RPL_LUNDATA_LUN_BUS_MASK](#) 0xe0
- #define [RPL_LUNDATA_LUN_LUN_MASK](#) 0x1f
- #define [RPL_LUNDATA_EXT_LEN_MASK](#) 0x30
- #define [RPL_LUNDATA_EXT_EAM_MASK](#) 0x0f
- #define [RPL_LUNDATA_EXT_EAM_WK](#) 0x01
- #define [RPL_LUNDATA_EXT_EAM_NOT_SPEC](#) 0x0f
- #define [RPL_LUNDATA_ATYP_MASK](#) 0xc0
- #define [RPL_LUNDATA_ATYP_PERIPH](#) 0x00
- #define [RPL_LUNDATA_ATYP_FLAT](#) 0x40
- #define [RPL_LUNDATA_ATYP_LUN](#) 0x80
- #define [RPL_LUNDATA_ATYP_EXTLUN](#) 0xc0
- #define [SSD_ERRCODE](#) 0x7F

- #define `SSD_CURRENT_ERROR` 0x70
- #define `SSD_DEFERRED_ERROR` 0x71
- #define `SSD_ERRCODE_VALID` 0x80
- #define `SSD_KEY` 0x0F
- #define `SSD_KEY_NO_SENSE` 0x00
- #define `SSD_KEY_RECOVERED_ERROR` 0x01
- #define `SSD_KEY_NOT_READY` 0x02
- #define `SSD_KEY_MEDIUM_ERROR` 0x03
- #define `SSD_KEY_HARDWARE_ERROR` 0x04
- #define `SSD_KEY_ILLEGAL_REQUEST` 0x05
- #define `SSD_KEY_UNIT_ATTENTION` 0x06
- #define `SSD_KEY_DATA_PROTECT` 0x07
- #define `SSD_KEY_BLANK_CHECK` 0x08
- #define `SSD_KEY_Vendor_Specific` 0x09
- #define `SSD_KEY_COPY_ABORTED` 0x0a
- #define `SSD_KEY_ABORTED_COMMAND` 0x0b
- #define `SSD_KEY_EQUAL` 0x0c
- #define `SSD_KEY_VOLUME_OVERFLOW` 0x0d
- #define `SSD_KEY_MISCOMPARE` 0x0e
- #define `SSD_KEY_RESERVED` 0x0f
- #define `SSD_ILI` 0x20
- #define `SSD_EOM` 0x40
- #define `SSD_FILEMARK` 0x80
- #define `SSD_SCS_VALID` 0x80
- #define `SSD_FIELDPTR_CMD` 0x40
- #define `SSD_BITPTR_VALID` 0x08
- #define `SSD_BITPTR_VALUE` 0x07
- #define `SSD_MIN_SIZE` 18
- #define `SSD_FULL_SIZE` sizeof(struct `scsi_sense_data`)
- #define `SCSI_DEFAULT_DENSITY` 0x00
- #define `SCSI_SAME_DENSITY` 0x7f
- #define `SCSI_STATUS_OK` 0x00
- #define `SCSI_STATUS_CHECK_COND` 0x02
- #define `SCSI_STATUS_COND_MET` 0x04
- #define `SCSI_STATUS_BUSY` 0x08
- #define `SCSI_STATUS_INTERMED` 0x10
- #define `SCSI_STATUS_INTERMED_COND_MET` 0x14
- #define `SCSI_STATUS_RESERV_CONFLICT` 0x18
- #define `SCSI_STATUS_CMD_TERMINATED` 0x22
- #define `SCSI_STATUS_QUEUE_FULL` 0x28
- #define `SCSI_STATUS_ACA_ACTIVE` 0x30
- #define `SCSI_STATUS_TASK_ABORTED` 0x40
- #define `SIP_MEDIA_REMOVABLE` 0x01
- #define `SIP_MEDIA_FIXED` 0x02
- #define `SF_RETRY_UA` 0x01
- #define `SF_NO_PRINT` 0x02
- #define `SF_QUIET_IR` 0x04
- #define `SF_PRINT_ALWAYS` 0x08

Enumerations

- enum `scsi_sense_action` {
`SS_NOP` = 0x000000, `SS_RETRY` = 0x010000, `SS_FAIL` = 0x020000, `SS_START` = 0x030000,
`SS_TUR` = 0x040000, `SS_REQSENSE` = 0x050000, `SS_MASK` = 0xff0000 }
- enum `scsi_sense_action_qualifier` {
`SSQ_NONE` = 0x0000, `SSQ_DECREMENT_COUNT` = 0x0100, `SSQ_MANY` = 0x0200, `SSQ_RANGE` = 0x0400,
`SSQ_PRINT_SENSE` = 0x0800, `SSQ_MASK` = 0xff00 }
- enum `scsi_sense_string_flags` { `SSS_FLAG_NONE` = 0x00, `SSS_FLAG_PRINT_COMMAND` = 0x01 }

Functions

- `__BEGIN_DECLS` void `scsi_sense_desc` (int sense_key, int asc, int ascq, struct `scsi_inquiry_data` *inq_data, const char **sense_key_desc, const char **asc_desc)
- `scsi_sense_action` `scsi_error_action` (struct `ccb_scsiio` *csio, struct `scsi_inquiry_data` *inq_data, u_int32_t sense_flags)
- const char * `scsi_status_string` (struct `ccb_scsiio` *csio)
- int `scsi_command_string` (struct `ccb_scsiio` *csio, struct sbuf *sb)
- int `scsi_sense_sbuf` (struct `ccb_scsiio` *csio, struct sbuf *sb, `scsi_sense_string_flags` flags)
- char * `scsi_sense_string` (struct `ccb_scsiio` *csio, char *str, int str_len)
- void `scsi_sense_print` (struct `ccb_scsiio` *csio)
- int `scsi_interpret_sense` (union `ccb` *ccb, u_int32_t sense_flags, u_int32_t *relsim_flags, u_int32_t *reduction, u_int32_t *timeout, `scsi_sense_action` error_action)
- const char * `scsi_op_desc` (u_int16_t opcode, struct `scsi_inquiry_data` *inq_data)
- char * `scsi_cdb_string` (u_int8_t *cdb_ptr, char *cdb_string, size_t len)
- void `scsi_print_inquiry` (struct `scsi_inquiry_data` *inq_data)
- u_int `scsi_calc_syncrate` (u_int period_factor)
- u_int `scsi_calc_syncparam` (u_int period)
- void `scsi_test_unit_ready` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcn)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_request_sense` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcn)(struct `cam_periph` *, union `ccb` *), void *data_ptr, u_int8_t dxfer_len, u_int8_t tag_action, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_inquiry` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcn)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, u_int8_t *inq_buf, u_int32_t inq_len, int evpd, u_int8_t page_code, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_mode_sense` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcn)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, int dbd, u_int8_t page_code, u_int8_t page, u_int8_t *param_buf, u_int32_t param_len, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_mode_sense_len` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcn)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, int dbd, u_int8_t page_code, u_int8_t page, u_int8_t *param_buf, u_int32_t param_len, int minimum_cmd_size, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_mode_select` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcn)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, int scsi_page_fmt, int save_pages, u_int8_t *param_buf, u_int32_t param_len, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_mode_select_len` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcn)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, int scsi_page_fmt, int save_pages, u_int8_t *param_buf, u_int32_t param_len, int minimum_cmd_size, u_int8_t sense_len, u_int32_t timeout)

- void `scsi_log_sense` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, u_int8_t page_code, u_int8_t page, int save_pages, int ppc, u_int32_t paramptr, u_int8_t *param_buf, u_int32_t param_len, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_log_select` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, u_int8_t page_code, int save_pages, int pc_reset, u_int8_t *param_buf, u_int32_t param_len, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_prevent` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, u_int8_t action, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_read_capacity` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, struct `scsi_read_capacity_data` *, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_read_capacity_16` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, u_int64_t lba, int reladr, int pmi, struct `scsi_read_capacity_data_long` *rcap_buf, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_report_luns` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, u_int8_t select_report, struct `scsi_report_luns_data` *rpl_buf, u_int32_t alloc_len, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_synchronize_cache` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, u_int32_t begin_lba, u_int16_t lb_count, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_read_write` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, int readop, u_int8_t byte2, int minimum_cmd_size, u_int64_t lba, u_int32_t block_count, u_int8_t *data_ptr, u_int32_t dxfer_len, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_start_stop` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, int start, int load_eject, int immediate, u_int8_t sense_len, u_int32_t timeout)
- int `scsi_inquiry_match` (caddr_t inqbuffer, caddr_t table_entry)
- int `scsi_static_inquiry_match` (caddr_t inqbuffer, caddr_t table_entry)
- static __inline void `scsi_extract_sense` (struct `scsi_sense_data` *sense, int *error_code, int *sense_key, int *asc, int *ascq)
- static __inline void `scsi_ulto2b` (u_int32_t val, u_int8_t *bytes)
- static __inline void `scsi_ulto3b` (u_int32_t val, u_int8_t *bytes)
- static __inline void `scsi_ulto4b` (u_int32_t val, u_int8_t *bytes)
- static __inline void `scsi_u64to8b` (u_int64_t val, u_int8_t *bytes)
- static __inline u_int32_t `scsi_2btoul` (u_int8_t *bytes)
- static __inline u_int32_t `scsi_3btoul` (u_int8_t *bytes)
- static __inline int32_t `scsi_3btol` (u_int8_t *bytes)
- static __inline u_int32_t `scsi_4btoul` (u_int8_t *bytes)
- static __inline u_int64_t `scsi_8btou64` (u_int8_t *bytes)
- static __inline void * `find_mode_page_6` (struct `scsi_mode_header_6` *mode_header)
- static __inline void * `find_mode_page_10` (struct `scsi_mode_header_10` *mode_header)
- static __inline u_int64_t `scsi_8btou64` (u_int8_t *bytes)

Variables

- int `scsi_delay`
- const char * `scsi_sense_key_text` []

7.17.1 Define Documentation

7.17.1.1 #define _SCSI_SCSI_ALL_H 1

Definition at line 25 of file scsi_all.h.

7.17.1.2 #define CHANGE_DEFINITION 0x40

Definition at line 499 of file scsi_all.h.

7.17.1.3 #define INQ_DATA_TQ_ENABLED(iqd)

Value:

```
((SID_ANSI_REV(iqd) < SCSI_REV_SPC2)? ((iqd)->flags & SID_CmdQue) : \
  (((iqd)->flags & SID_CmdQue) && !((iqd)->spc2_flags & SPC2_SID_BQueue)) || \
  (!((iqd)->flags & SID_CmdQue) && ((iqd)->spc2_flags & SPC2_SID_BQueue)))
```

Definition at line 612 of file scsi_all.h.

Referenced by probedone(), xpt_action(), and xpt_set_transfer_settings().

7.17.1.4 #define INQUIRY 0x12

Definition at line 482 of file scsi_all.h.

Referenced by scsi_inquiry(), and targbhdone().

7.17.1.5 #define LOG_SELECT 0x4c

Definition at line 500 of file scsi_all.h.

Referenced by scsi_log_select().

7.17.1.6 #define LOG_SENSE 0x4d

Definition at line 501 of file scsi_all.h.

Referenced by scsi_log_sense().

7.17.1.7 #define MODE_SELECT_10 0x55

Definition at line 502 of file scsi_all.h.

Referenced by cd6byteworkaround(), and scsi_mode_select_len().

7.17.1.8 #define MODE_SELECT_6 0x15

Definition at line 483 of file scsi_all.h.

Referenced by cd6byteworkaround(), and scsi_mode_select_len().

7.17.1.9 #define MODE_SENSE_10 0x5A

Definition at line 503 of file scsi_all.h.

Referenced by cd6byteworkaround(), and scsi_mode_sense_len().

7.17.1.10 #define MODE_SENSE_6 0x1a

Definition at line 484 of file scsi_all.h.

Referenced by cd6byteworkaround(), and scsi_mode_sense_len().

7.17.1.11 #define MOVE_MEDIUM 0xa5

Definition at line 508 of file scsi_all.h.

Referenced by scsi_move_medium().

7.17.1.12 #define POSITION_TO_ELEMENT 0x2b

Definition at line 495 of file scsi_all.h.

Referenced by scsi_position_to_element().

7.17.1.13 #define PR_ALLOW 0x00

Definition at line 359 of file scsi_all.h.

Referenced by cdcheckmedia(), cdclose(), cdioctl(), cdprevent(), daclose(), daprevent(), saclose(), saioctl(), and saprevent().

7.17.1.14 #define PR_PREVENT 0x01

Definition at line 358 of file scsi_all.h.

Referenced by cdcheckmedia(), cdioctl(), cdprevent(), daprevent(), saopen(), saprevent(), sastrategy(), and SLIST_HEAD().

7.17.1.15 #define PREVENT_ALLOW 0x1e

Definition at line 491 of file scsi_all.h.

Referenced by scsi_prevent().

7.17.1.16 #define READ_10 0x28

Definition at line 493 of file scsi_all.h.

Referenced by cmd6workaround(), and scsi_read_write().

7.17.1.17 #define READ_12 0xa8

Definition at line 509 of file scsi_all.h.

Referenced by scsi_read_write().

7.17.1.18 #define READ_16 0x88

Definition at line 504 of file scsi_all.h.

Referenced by scsi_read_write().

7.17.1.19 #define READ_6 0x08

Definition at line 480 of file scsi_all.h.

Referenced by cmd6workaround(), and scsi_read_write().

7.17.1.20 #define READ_BUFFER 0x3c

Definition at line 498 of file scsi_all.h.

Referenced by safte_getconfig(), and safte_rdstat().

7.17.1.21 #define READ_CAPACITY 0x25

Definition at line 492 of file scsi_all.h.

Referenced by scsi_read_capacity().

7.17.1.22 #define READ_ELEMENT_STATUS 0xb8

Definition at line 511 of file scsi_all.h.

Referenced by scsi_read_element_status().

7.17.1.23 #define RECEIVE_DIAGNOSTIC 0x1c

Definition at line 489 of file scsi_all.h.

Referenced by ses_getconfig(), and ses_getputstat().

7.17.1.24 #define RELEASE 0x17

Definition at line 488 of file scsi_all.h.

7.17.1.25 #define REPORT_LUNS 0xA0

Definition at line 507 of file scsi_all.h.

Referenced by scsi_report_luns().

7.17.1.26 #define REQUEST_SENSE 0x03

Definition at line 479 of file scsi_all.h.

Referenced by camperiphdone(), scsi_low_sense_abort_start(), scsi_request_sense(), and targbhdone().

7.17.1.27 #define RESERVE 0x16

Definition at line 487 of file scsi_all.h.

7.17.1.28 #define RPL_LUNDATA_ATYP_EXTLUN 0xc0

Definition at line 748 of file scsi_all.h.

7.17.1.29 #define RPL_LUNDATA_ATYP_FLAT 0x40

Definition at line 746 of file scsi_all.h.

7.17.1.30 #define RPL_LUNDATA_ATYP_LUN 0x80

Definition at line 747 of file scsi_all.h.

7.17.1.31 #define RPL_LUNDATA_ATYP_MASK 0xc0

Definition at line 744 of file scsi_all.h.

7.17.1.32 #define RPL_LUNDATA_ATYP_PERIPH 0x00

Definition at line 745 of file scsi_all.h.

7.17.1.33 #define RPL_LUNDATA_EXT_EAM_MASK 0x0f

Definition at line 741 of file scsi_all.h.

7.17.1.34 #define RPL_LUNDATA_EXT_EAM_NOT_SPEC 0x0f

Definition at line 743 of file scsi_all.h.

7.17.1.35 #define RPL_LUNDATA_EXT_EAM_WK 0x01

Definition at line 742 of file scsi_all.h.

7.17.1.36 #define RPL_LUNDATA_EXT_LEN_MASK 0x30

Definition at line 740 of file scsi_all.h.

7.17.1.37 #define RPL_LUNDATA_FLAT_LUN_MASK 0x3f

Definition at line 736 of file scsi_all.h.

7.17.1.38 #define RPL_LUNDATA_LUN_BUS_MASK 0xe0

Definition at line 738 of file scsi_all.h.

7.17.1.39 #define RPL_LUNDATA_LUN_LUN_MASK 0x1f

Definition at line 739 of file scsi_all.h.

7.17.1.40 #define RPL_LUNDATA_LUN_TARG_MASK 0x3f

Definition at line 737 of file scsi_all.h.

7.17.1.41 #define RPL_LUNDATA_PERIPH_BUS_MASK 0x3f

Definition at line 735 of file scsi_all.h.

7.17.1.42 #define RPL_REPORT_ALL 0x02

Definition at line 717 of file scsi_all.h.

7.17.1.43 #define RPL_REPORT_DEFAULT 0x00

Definition at line 715 of file scsi_all.h.

7.17.1.44 #define RPL_REPORT_WELLKNOWN 0x01

Definition at line 716 of file scsi_all.h.

7.17.1.45 #define RWB_MODE 0x07

Definition at line 387 of file scsi_all.h.

7.17.1.46 #define RWB_MODE_DATA 0x02

Definition at line 389 of file scsi_all.h.

7.17.1.47 #define RWB_MODE_DOWNLOAD 0x04

Definition at line 390 of file scsi_all.h.

7.17.1.48 #define RWB_MODE_DOWNLOAD_SAVE 0x05

Definition at line 391 of file scsi_all.h.

7.17.1.49 #define RWB_MODE_HDR_DATA 0x00

Definition at line 388 of file scsi_all.h.

7.17.1.50 #define SC_SCSI_1 0x01

Definition at line 471 of file scsi_all.h.

7.17.1.51 #define SC_SCSI_2 0x03

Definition at line 472 of file scsi_all.h.

7.17.1.52 #define SCB_RLEC 0x01

Definition at line 316 of file scsi_all.h.

7.17.1.53 #define SCP_EAENP 0x01

Definition at line 327 of file scsi_all.h.

7.17.1.54 #define SCP_EECA 0x80

Definition at line 324 of file scsi_all.h.

7.17.1.55 #define SCP_QUEUE_ALG_MASK 0xF0

Definition at line 318 of file scsi_all.h.

7.17.1.56 #define SCP_QUEUE_ALG_RESTRICTED 0x00

Definition at line 319 of file scsi_all.h.

7.17.1.57 #define SCP_QUEUE_ALG_UNRESTRICTED 0x10

Definition at line 320 of file scsi_all.h.

7.17.1.58 #define SCP_QUEUE_DQUE 0x01

Definition at line 322 of file scsi_all.h.

Referenced by xpt_set_transfer_settings().

7.17.1.59 #define SCP_QUEUE_ERR 0x02

Definition at line 321 of file scsi_all.h.

7.17.1.60 #define SCP_RAENP 0x04

Definition at line 325 of file scsi_all.h.

7.17.1.61 #define SCP_UAAENP 0x02

Definition at line 326 of file scsi_all.h.

7.17.1.62 #define SCSI_CDB6_LEN(len) ((len) == 0 ? 256 : len)

Definition at line 59 of file scsi_all.h.

Referenced by targbdone().

7.17.1.63 #define SCSI_CMD_LUN 0xA0

Definition at line 47 of file scsi_all.h.

7.17.1.64 #define SCSI_CMD_LUN_SHIFT 5

Definition at line 48 of file scsi_all.h.

7.17.1.65 #define SCSI_CTL_FLAG 0x02

Definition at line 45 of file scsi_all.h.

7.17.1.66 #define SCSI_CTL_LINK 0x01

Definition at line 44 of file scsi_all.h.

7.17.1.67 #define SCSI_CTL_VENDOR 0xC0

Definition at line 46 of file scsi_all.h.

7.17.1.68 #define SCSI_DEFAULT_DENSITY 0x00

Definition at line 827 of file scsi_all.h.

Referenced by samount().

7.17.1.69 #define SCSI_MAX_CDBLEN 16

Definition at line 50 of file scsi_all.h.

Referenced by scsi_command_string(), and xpt_action().

7.17.1.70 #define SCSI_REV_0 0

Definition at line 590 of file scsi_all.h.

7.17.1.71 #define SCSI_REV_2 2

Definition at line 592 of file scsi_all.h.

Referenced by xpt_action().

7.17.1.72 #define SCSI_REV_CCS 1

Definition at line 591 of file scsi_all.h.

Referenced by sasetparams(), and scsi_print_inquiry().

7.17.1.73 #define SCSI_REV_SPC 3

Definition at line 593 of file scsi_all.h.

Referenced by samount().

7.17.1.74 #define SCSI_REV_SPC2 4

Definition at line 594 of file scsi_all.h.

7.17.1.75 #define SCSI_SAME_DENSITY 0x7f

Definition at line 828 of file scsi_all.h.

Referenced by sasetparams().

7.17.1.76 #define SCSI_STATUS_ACA_ACTIVE 0x30

Definition at line 843 of file scsi_all.h.

Referenced by scsi_status_string().

7.17.1.77 #define SCSI_STATUS_BUSY 0x08

Definition at line 837 of file scsi_all.h.

Referenced by camperiphscsistatuserror(), cddone(), and scsi_status_string().

7.17.1.78 #define SCSI_STATUS_CHECK_COND 0x02

Definition at line 835 of file scsi_all.h.

Referenced by cam_error_string(), camperiphscsistatuserror(), cderror(), daerror(), dashutdown(), scsi_status_string(), and targbhdone().

7.17.1.79 #define SCSI_STATUS_CMD_TERMINATED 0x22

Definition at line 841 of file scsi_all.h.

Referenced by camperiphscsistatuserror(), and scsi_status_string().

7.17.1.80 #define SCSI_STATUS_COND_MET 0x04

Definition at line 836 of file scsi_all.h.

Referenced by camperiphscsistatuserror().

7.17.1.81 #define SCSI_STATUS_INTERMED 0x10

Definition at line 838 of file scsi_all.h.

Referenced by camperiphscsistatuserror(), and scsi_status_string().

7.17.1.82 #define SCSI_STATUS_INTERMED_COND_MET 0x14

Definition at line 839 of file scsi_all.h.

Referenced by camperiphscsistatuserror(), and scsi_status_string().

7.17.1.83 #define SCSI_STATUS_OK 0x00

Definition at line 834 of file scsi_all.h.

Referenced by camperiphscsistatuserror(), scsi_status_string(), targbhdone(), and xpt_action().

7.17.1.84 #define SCSI_STATUS_QUEUE_FULL 0x28

Definition at line 842 of file scsi_all.h.

Referenced by camperiphscsistatuserror(), and scsi_status_string().

7.17.1.85 #define SCSI_STATUS_RESERV_CONFLICT 0x18

Definition at line 840 of file scsi_all.h.

Referenced by camperiphscsistatuserror(), and scsi_status_string().

7.17.1.86 #define SCSI_STATUS_TASK_ABORTED 0x40

Definition at line 844 of file scsi_all.h.

Referenced by `scsi_status_string()`.

7.17.1.87 #define SEND_DIAGNOSTIC 0x1d

Definition at line 490 of file `scsi_all.h`.

Referenced by `saft_e_init_enc()`, and `ses_getputstat()`.

7.17.1.88 #define SERVICE_ACTION_IN 0x9e

Definition at line 506 of file `scsi_all.h`.

Referenced by `scsi_read_capacity_16()`.

7.17.1.89 #define SF_NO_PRINT 0x02

Definition at line 955 of file `scsi_all.h`.

Referenced by `cdcheckmedia()`, `cddone()`, `cdprevent()`, `cddsize()`, `chdone()`, `chgetparams()`, `dadone()`, `dagetcapacity()`, `probedone()`, `sagetparams()`, `saiocctl()`, `samount()`, `sareservereleaseunit()`, and `scsi_error_action()`.

7.17.1.90 #define SF_PRINT_ALWAYS 0x08

Definition at line 957 of file `scsi_all.h`.

Referenced by `scsi_error_action()`.

7.17.1.91 #define SF_QUIET_IR 0x04

Definition at line 956 of file `scsi_all.h`.

Referenced by `probedone()`, `saprevent()`, and `scsi_error_action()`.

7.17.1.92 #define SF_RETRY_UA 0x01

Definition at line 954 of file `scsi_all.h`.

Referenced by `cddone()`, `cderror()`, `cdgetmode()`, `cdpause()`, `cdplay()`, `cdplaymsf()`, `cdplaytracks()`, `cdprevent()`, `cdreadvdstructure()`, `cdreadsubchannel()`, `cdreadtoc()`, `cdreportkey()`, `cdsendkey()`, `cdsetmode()`, `cdsetspeed()`, `cddsize()`, `cdstartunit()`, `cdstopunit()`, `chdone()`, `chexchange()`, `chgetelemstatus()`, `chgetparams()`, `chielem()`, `chmove()`, `chposition()`, `daclose()`, `dadone()`, `daerror()`, `dagetcapacity()`, `daprevent()`, `passendccb()`, `probedone()`, `ptdone()`, `samount()`, `sareservereleaseunit()`, and `scsi_error_action()`.

7.17.1.93 #define SHORT_INQUIRY_LENGTH 36

Definition at line 545 of file `scsi_all.h`.

Referenced by `probedone()`, and `probestart()`.

7.17.1.94 #define SI_EVPD 0x01

Definition at line 157 of file scsi_all.h.

Referenced by scsi_inquiry(), and targbhdone().

7.17.1.95 #define SID_ADDITIONAL_LENGTH(iqd)**Value:**

```
((iqd)->additional_length +  
         offsetof(struct scsi_inquiry_data, additional_length) + 1)
```

Definition at line 602 of file scsi_all.h.

Referenced by probestart().

7.17.1.96 #define SID_AENC 0x80

Definition at line 599 of file scsi_all.h.

7.17.1.97 #define SID_ANSI_REV(inq_data) ((inq_data) → version & 0x07)

Definition at line 589 of file scsi_all.h.

Referenced by saregister(), scsi_print_inquiry(), and xpt_devise_transport().

7.17.1.98 #define SID_CmdQue 0x02

Definition at line 619 of file scsi_all.h.

Referenced by cdregister(), daregister(), passregister(), xpt_announce_periph(), xpt_dev_ccbq_resize(), xpt_run_dev_sendq(), xpt_set_transfer_settings(), xpt_start_tags(), and xpt_toggle_tags().

7.17.1.99 #define SID_ECMA 0x38

Definition at line 596 of file scsi_all.h.

7.17.1.100 #define SID_IS_REMOVABLE(inq_data) (((inq_data) → dev_qual2 & 0x80) != 0)

Definition at line 587 of file scsi_all.h.

Referenced by camperiphscsisenseerror(), cdregister(), daregister(), scsi_inquiry_match(), scsi_print_inquiry(), and scsi_static_inquiry_match().

7.17.1.101 #define SID_ISO 0xC0

Definition at line 597 of file scsi_all.h.

7.17.1.102 #define SID_Linked 0x08

Definition at line 620 of file scsi_all.h.

7.17.1.103 #define SID_PRODUCT_SIZE 16

Definition at line 627 of file scsi_all.h.

7.17.1.104 #define SID_QUAL(inq_data) (((inq_data) → device & 0xE0) >> 5)

Definition at line 551 of file scsi_all.h.

Referenced by probedone(), and scsi_print_inquiry().

7.17.1.105 #define SID_QUAL2 0x7F

Definition at line 586 of file scsi_all.h.

7.17.1.106 #define SID_QUAL_BAD_LU 0x03

Definition at line 572 of file scsi_all.h.

Referenced by scsi_print_inquiry().

7.17.1.107 #define SID_QUAL_IS_VENDOR_UNIQUE(inq_data) ((SID_QUAL(inq_data) & 0x08) != 0)

Definition at line 584 of file scsi_all.h.

Referenced by scsi_print_inquiry().

7.17.1.108 #define SID_QUAL_LU_CONNECTED 0x00

Definition at line 552 of file scsi_all.h.

Referenced by probedone(), and scsi_print_inquiry().

7.17.1.109 #define SID_QUAL_LU_OFFLINE 0x01

Definition at line 564 of file scsi_all.h.

Referenced by scsi_print_inquiry().

7.17.1.110 #define SID_QUAL_RSVD 0x02

Definition at line 571 of file scsi_all.h.

Referenced by scsi_print_inquiry().

7.17.1.111 #define SID_RelAdr 0x80

Definition at line 624 of file scsi_all.h.

7.17.1.112 #define SID_REVISION_SIZE 4

Definition at line 629 of file scsi_all.h.

7.17.1.113 #define SID_SftRe 0x01

Definition at line 618 of file scsi_all.h.

7.17.1.114 #define SID_SPI_CLOCK_DT 0x04

Definition at line 643 of file scsi_all.h.

Referenced by xpt_set_transfer_settings().

7.17.1.115 #define SID_SPI_CLOCK_DT_ST 0x0C

Definition at line 644 of file scsi_all.h.

7.17.1.116 #define SID_SPI_CLOCK_ST 0x00

Definition at line 642 of file scsi_all.h.

7.17.1.117 #define SID_SPI_IUS 0x01

Definition at line 640 of file scsi_all.h.

Referenced by xpt_set_transfer_settings().

7.17.1.118 #define SID_SPI_MASK 0x0F

Definition at line 645 of file scsi_all.h.

Referenced by xpt_devise_transport().

7.17.1.119 #define SID_SPI_QAS 0x02

Definition at line 641 of file scsi_all.h.

Referenced by xpt_set_transfer_settings().

7.17.1.120 #define SID_Sync 0x10

Definition at line 621 of file scsi_all.h.

Referenced by probedone(), xpt_set_transfer_settings(), and xpt_toggle_tags().

7.17.1.121 #define SID_TrmIOP 0x40

Definition at line 600 of file scsi_all.h.

7.17.1.122 #define SID_TYPE(inq_data) ((inq_data) → device & 0x1f)

Definition at line 550 of file scsi_all.h.

Referenced by cdasync(), cddone(), chasync(), chregister(), daasync(), daregister(), passregister(), ptasync(), ptctor(), saasync(), saregister(), scsi_inquiry_match(), scsi_op_desc(), scsi_print_inquiry(), and scsi_static_inquiry_match().

7.17.1.123 #define SID_VENDOR_SIZE 8

Definition at line 625 of file scsi_all.h.

7.17.1.124 #define SID_VENDOR_SPECIFIC_0_SIZE 20

Definition at line 635 of file scsi_all.h.

7.17.1.125 #define SID_VENDOR_SPECIFIC_1_SIZE 160

Definition at line 662 of file scsi_all.h.

7.17.1.126 #define SID_WBus16 0x20

Definition at line 622 of file scsi_all.h.

Referenced by xpt_set_transfer_settings(), and xpt_toggle_tags().

7.17.1.127 #define SID_WBus32 0x40

Definition at line 623 of file scsi_all.h.

Referenced by xpt_set_transfer_settings(), and xpt_toggle_tags().

7.17.1.128 #define SIP_MEDIA_FIXED 0x02

Definition at line 850 of file scsi_all.h.

Referenced by scsi_inquiry_match(), and scsi_static_inquiry_match().

7.17.1.129 #define SIP_MEDIA_REMOVABLE 0x01

Definition at line 849 of file scsi_all.h.

Referenced by scsi_inquiry_match(), and scsi_static_inquiry_match().

7.17.1.130 #define SLP_DS 0x40

Definition at line 307 of file scsi_all.h.

7.17.1.131 #define SLP_DU 0x80

Definition at line 308 of file scsi_all.h.

7.17.1.132 #define SLP_ETC 0x10

Definition at line 305 of file scsi_all.h.

7.17.1.133 #define SLP_LBIN 0x02

Definition at line 299 of file scsi_all.h.

7.17.1.134 #define SLP_LP 0x01

Definition at line 298 of file scsi_all.h.

7.17.1.135 #define SLP_TMC_ALWAYS 0x00

Definition at line 301 of file scsi_all.h.

7.17.1.136 #define SLP_TMC_EQUAL 0x04

Definition at line 302 of file scsi_all.h.

7.17.1.137 #define SLP_TMC_GREATER 0x0C

Definition at line 304 of file scsi_all.h.

7.17.1.138 #define SLP_TMC_MASK 0x0C

Definition at line 300 of file scsi_all.h.

7.17.1.139 #define SLP_TMC_NOTEQUAL 0x08

Definition at line 303 of file scsi_all.h.

7.17.1.140 #define SLP_TSD 0x20

Definition at line 306 of file scsi_all.h.

7.17.1.141 #define SLS_ALL_PAGES_PAGE 0x00

Definition at line 252 of file scsi_all.h.

7.17.1.142 #define SLS_ERROR_LASTN_PAGE 0x07

Definition at line 259 of file scsi_all.h.

7.17.1.143 #define SLS_ERROR_NONMEDIUM_PAGE 0x06

Definition at line 258 of file scsi_all.h.

7.17.1.144 #define SLS_ERROR_READ_PAGE 0x03

Definition at line 255 of file scsi_all.h.

7.17.1.145 #define SLS_ERROR_READREVERSE_PAGE 0x04

Definition at line 256 of file scsi_all.h.

7.17.1.146 #define SLS_ERROR_VERIFY_PAGE 0x05

Definition at line 257 of file scsi_all.h.

7.17.1.147 #define SLS_ERROR_WRITE_PAGE 0x02

Definition at line 254 of file scsi_all.h.

7.17.1.148 #define SLS_OVERRUN_PAGE 0x01

Definition at line 253 of file scsi_all.h.

7.17.1.149 #define SLS_PAGE_CODE 0x3F

Definition at line 251 of file scsi_all.h.

Referenced by scsi_log_select().

7.17.1.150 #define SLS_PAGE_CTRL_CUMUL_DEFAULT 0xC0

Definition at line 264 of file scsi_all.h.

7.17.1.151 #define SLS_PAGE_CTRL_CUMULATIVE 0x40

Definition at line 262 of file scsi_all.h.

7.17.1.152 #define SLS_PAGE_CTRL_MASK 0xC0

Definition at line 260 of file scsi_all.h.

7.17.1.153 #define SLS_PAGE_CTRL_THRESH_DEFAULT 0x80

Definition at line 263 of file scsi_all.h.

7.17.1.154 #define SLS_PAGE_CTRL_THRESHOLD 0x00

Definition at line 261 of file scsi_all.h.

7.17.1.155 #define SLS_PCR 0x02

Definition at line 276 of file scsi_all.h.

Referenced by scsi_log_select().

7.17.1.156 #define SLS_PPC 0x02

Definition at line 249 of file scsi_all.h.

Referenced by scsi_log_sense().

7.17.1.157 #define SLS_SP 0x01

Definition at line 248 of file scsi_all.h.

Referenced by scsi_log_select(), and scsi_log_sense().

7.17.1.158 #define SMS_ALL_PAGES_PAGE 0x3F

Definition at line 175 of file scsi_all.h.

7.17.1.159 #define SMS_CONTROL_MODE_PAGE 0x0A

Definition at line 174 of file scsi_all.h.

Referenced by probestart().

7.17.1.160 #define SMS_DBD 0x08

Definition at line 168 of file scsi_all.h.

Referenced by chdone(), chgetparams(), and scsi_mode_sense_len().

7.17.1.161 #define SMS_DISCONNECT_RECONNECT_PAGE 0x02

Definition at line 172 of file scsi_all.h.

7.17.1.162 #define SMS_PAGE_CODE 0x3F

Definition at line 170 of file scsi_all.h.

7.17.1.163 #define SMS_PAGE_CTRL_CHANGEABLE 0x40

Definition at line 178 of file scsi_all.h.

7.17.1.164 #define SMS_PAGE_CTRL_CURRENT 0x00

Definition at line 177 of file scsi_all.h.

Referenced by cdgetmode(), chgetparams(), chstart(), probestart(), and sagetparams().

7.17.1.165 #define SMS_PAGE_CTRL_DEFAULT 0x80

Definition at line 179 of file scsi_all.h.

7.17.1.166 #define SMS_PAGE_CTRL_MASK 0xC0

Definition at line 176 of file scsi_all.h.

7.17.1.167 #define SMS_PAGE_CTRL_SAVED 0xC0

Definition at line 180 of file scsi_all.h.

7.17.1.168 #define SMS_PERIPHERAL_DEVICE_PAGE 0x09

Definition at line 173 of file scsi_all.h.

7.17.1.169 #define SMS_PF 0x10

Definition at line 201 of file scsi_all.h.

Referenced by scsi_mode_select_len().

7.17.1.170 #define SMS_SP 0x01

Definition at line 200 of file scsi_all.h.

Referenced by scsi_mode_select_len().

7.17.1.171 #define SMS_VENDOR_SPECIFIC_PAGE 0x00

Definition at line 171 of file scsi_all.h.

Referenced by sagetparams().

7.17.1.172 #define SPC2_SID_BQueue 0x80

Definition at line 610 of file scsi_all.h.

7.17.1.173 #define SPC2_SID_EncServ 0x40

Definition at line 609 of file scsi_all.h.

7.17.1.174 #define SPC2_SID_MChngr 0x08

Definition at line 607 of file scsi_all.h.

7.17.1.175 #define SPC2_SID_MultiP 0x10

Definition at line 608 of file scsi_all.h.

7.17.1.176 #define SRC16_PMI 0x01

Definition at line 693 of file scsi_all.h.

Referenced by scsi_read_capacity_16().

7.17.1.177 #define SRC16_RELADR 0x02

Definition at line 694 of file scsi_all.h.

Referenced by scsi_read_capacity_16().

7.17.1.178 #define SRC16_SERVICE_ACTION 0x10

Definition at line 689 of file scsi_all.h.

Referenced by scsi_read_capacity_16().

7.17.1.179 #define SRW10_DPO 0x10

Definition at line 425 of file scsi_all.h.

7.17.1.180 #define SRW10_EBP 0x04

Definition at line 423 of file scsi_all.h.

7.17.1.181 #define SRW10_FUA 0x08

Definition at line 424 of file scsi_all.h.

7.17.1.182 #define SRW10_RELADDR 0x01

Definition at line 421 of file scsi_all.h.

7.17.1.183 #define SRW12_DPO 0x10

Definition at line 438 of file scsi_all.h.

7.17.1.184 #define SRW12_FUA 0x08

Definition at line 437 of file scsi_all.h.

7.17.1.185 #define SRW12_RELADDR 0x01

Definition at line 436 of file scsi_all.h.

7.17.1.186 #define SRW16_DPO 0x10

Definition at line 451 of file scsi_all.h.

7.17.1.187 #define SRW16_FUA 0x08

Definition at line 450 of file scsi_all.h.

7.17.1.188 #define SRW16_RELADDR 0x01

Definition at line 449 of file scsi_all.h.

7.17.1.189 #define SRW_TOPADDR 0x1F

Definition at line 413 of file scsi_all.h.

7.17.1.190 #define SS_ERRMASK 0xff

Definition at line 97 of file scsi_all.h.

Referenced by camperiphscsisenseerror(), and scsi_error_action().

7.17.1.191 #define SS_FATAL SS_FAIL|SSQ_PRINT_SENSE

Definition at line 106 of file scsi_all.h.

7.17.1.192 #define SS_RDEF SS_RETRY|SSQ_DECREMENT_COUNT|SSQ_PRINT_SENSE|EIO

Definition at line 100 of file scsi_all.h.

7.17.1.193 #define SS_RET SS_RETRY|SSQ_DECREMENT_COUNT|SSQ_PRINT_SENSE

Definition at line 103 of file scsi_all.h.

7.17.1.194 #define SSD_BITPTR_VALID 0x08

Definition at line 789 of file scsi_all.h.

7.17.1.195 #define SSD_BITPTR_VALUE 0x07

Definition at line 790 of file scsi_all.h.

7.17.1.196 #define SSD_CURRENT_ERROR 0x70

Definition at line 755 of file scsi_all.h.

Referenced by cddone(), dadone(), and scsi_sense_sbuf().

7.17.1.197 #define SSD_DEFERRED_ERROR 0x71

Definition at line 756 of file scsi_all.h.

Referenced by scsi_error_action(), and scsi_sense_sbuf().

7.17.1.198 #define SSD_DOL 0x02

Definition at line 136 of file scsi_all.h.

7.17.1.199 #define SSD_EOM 0x40

Definition at line 778 of file scsi_all.h.

Referenced by saerror().

7.17.1.200 #define SSD_ERRCODE 0x7F

Definition at line 754 of file scsi_all.h.

Referenced by scsi_extract_sense(), and scsi_sense_sbuf().

7.17.1.201 #define SSD_ERRCODE_VALID 0x80

Definition at line 757 of file scsi_all.h.

Referenced by saerror(), and scsi_sense_sbuf().

7.17.1.202 #define SSD_FIELDPTR_CMD 0x40

Definition at line 788 of file scsi_all.h.

7.17.1.203 #define SSD_FILEMARK 0x80

Definition at line 779 of file `scsi_all.h`.

Referenced by `saerror()`.

7.17.1.204 #define SSD_FULL_SIZE sizeof(struct scsi_sense_data)

Definition at line 793 of file `scsi_all.h`.

Referenced by `camperiphscsisenseerror()`, `cdgetmode()`, `cdpause()`, `cdplay()`, `cdplaymsf()`, `cdplaytracks()`, `cdprevent()`, `cdreaddvdstructure()`, `cdreadsubchannel()`, `cdreadtoc()`, `cdreportkey()`, `cdsendkey()`, `cdsetmode()`, `cdsetspeed()`, `cdsize()`, `cdstart()`, `cdstartunit()`, `cdstopunit()`, `chexchange()`, `chgetelemstatus()`, `chgetparams()`, `chielem()`, `chmove()`, `chposition()`, `chstart()`, `daclose()`, `dadump()`, `dagetcapacity()`, `daprevent()`, `dashutdown()`, `dastart()`, `probstart()`, `ptstart()`, `saerase()`, `sagetparams()`, `saloadunload()`, `samount()`, `saprevent()`, `sardpos()`, `sareservereleaseunit()`, `saretension()`, `sarewind()`, `sasetparams()`, `sasetpos()`, `saspace()`, `sastart()`, and `sawritefilemarks()`.

7.17.1.205 #define SSD_ILI 0x20

Definition at line 777 of file `scsi_all.h`.

Referenced by `saerror()`, and `scsi_sense_sbuf()`.

7.17.1.206 #define SSD_KEY 0x0F

Definition at line 760 of file `scsi_all.h`.

Referenced by `camperiphdone()`, `scsi_extract_sense()`, and `scsi_sense_sbuf()`.

7.17.1.207 #define SSD_KEY_ABORTED_COMMAND 0x0b

Definition at line 772 of file `scsi_all.h`.

7.17.1.208 #define SSD_KEY_BLANK_CHECK 0x08

Definition at line 769 of file `scsi_all.h`.

Referenced by `saerror()`, and `scsi_sense_sbuf()`.

7.17.1.209 #define SSD_KEY_COPY_ABORTED 0x0a

Definition at line 771 of file `scsi_all.h`.

7.17.1.210 #define SSD_KEY_DATA_PROTECT 0x07

Definition at line 768 of file `scsi_all.h`.

Referenced by `scsi_sense_sbuf()`.

7.17.1.211 #define SSD_KEY_EQUAL 0x0c

Definition at line 773 of file scsi_all.h.

7.17.1.212 #define SSD_KEY_HARDWARE_ERROR 0x04

Definition at line 765 of file scsi_all.h.

Referenced by scsi_sense_sbuf().

7.17.1.213 #define SSD_KEY_ILLEGAL_REQUEST 0x05

Definition at line 766 of file scsi_all.h.

Referenced by cderror(), daclose(), dadump(), daerror(), dashutdown(), scsi_error_action(), and scsi_sense_sbuf().

7.17.1.214 #define SSD_KEY_MEDIUM_ERROR 0x03

Definition at line 764 of file scsi_all.h.

Referenced by scsi_sense_sbuf().

7.17.1.215 #define SSD_KEY_MISCOMPARE 0x0e

Definition at line 775 of file scsi_all.h.

7.17.1.216 #define SSD_KEY_NO_SENSE 0x00

Definition at line 761 of file scsi_all.h.

Referenced by camperiphdone(), and saerror().

7.17.1.217 #define SSD_KEY_NOT_READY 0x02

Definition at line 763 of file scsi_all.h.

Referenced by scsi_sense_sbuf().

7.17.1.218 #define SSD_KEY_RECOVERED_ERROR 0x01

Definition at line 762 of file scsi_all.h.

Referenced by scsi_error_action(), and scsi_sense_sbuf().

7.17.1.219 #define SSD_KEY_RESERVED 0x0f

Definition at line 776 of file scsi_all.h.

Referenced by saerror().

7.17.1.220 #define SSD_KEY_UNIT_ATTENTION 0x06

Definition at line 767 of file scsi_all.h.

Referenced by scsi_error_action(), and scsi_sense_sbuf().

7.17.1.221 #define SSD_KEY_Vendor_Specific 0x09

Definition at line 770 of file scsi_all.h.

7.17.1.222 #define SSD_KEY_VOLUME_OVERFLOW 0x0d

Definition at line 774 of file scsi_all.h.

Referenced by saerror().

7.17.1.223 #define SSD_MIN_SIZE 18

Definition at line 791 of file scsi_all.h.

Referenced by probestart().

7.17.1.224 #define SSD_PF 0x10

Definition at line 138 of file scsi_all.h.

7.17.1.225 #define SSD_SCS_VALID 0x80

Definition at line 787 of file scsi_all.h.

Referenced by scsi_sense_sbuf().

7.17.1.226 #define SSD_SELFTEST 0x04

Definition at line 137 of file scsi_all.h.

7.17.1.227 #define SSD_UOL 0x01

Definition at line 135 of file scsi_all.h.

7.17.1.228 #define SSS_IMMED 0x01

Definition at line 463 of file scsi_all.h.

Referenced by scsi_start_stop().

7.17.1.229 #define SSS_LOEJ 0x02

Definition at line 467 of file scsi_all.h.

Referenced by camperiphdone(), and scsi_start_stop().

7.17.1.230 #define SSS_START 0x01

Definition at line 466 of file scsi_all.h.

Referenced by scsi_start_stop().

7.17.1.231 #define START_STOP 0x1b

Definition at line 486 of file scsi_all.h.

7.17.1.232 #define START_STOP_UNIT 0x1b

Definition at line 485 of file scsi_all.h.

Referenced by camperiphdone(), and scsi_start_stop().

7.17.1.233 #define SVPD_SERIAL_NUM_SIZE 251

Definition at line 673 of file scsi_all.h.

7.17.1.234 #define SVPD_UNIT_SERIAL_NUMBER 0x80

Definition at line 670 of file scsi_all.h.

Referenced by probestart().

7.17.1.235 #define SYNCHRONIZE_CACHE 0x35

Definition at line 496 of file scsi_all.h.

Referenced by scsi_synchronize_cache().

7.17.1.236 #define T_ANY 0xFF

Definition at line 534 of file scsi_all.h.

Referenced by scsi_inquiry_match(), and scsi_static_inquiry_match().

7.17.1.237 #define T_ASC0 0x0a

Definition at line 527 of file scsi_all.h.

7.17.1.238 #define T_ASC1 0x0b

Definition at line 528 of file scsi_all.h.

7.17.1.239 #define T_CDROM 0x05

Definition at line 522 of file scsi_all.h.

Referenced by cdasync(), cddone(), and scsi_print_inquiry().

7.17.1.240 #define T_CHANGER 0x08

Definition at line 525 of file scsi_all.h.

Referenced by chasync(), and scsi_print_inquiry().

7.17.1.241 #define T_COMM 0x09

Definition at line 526 of file scsi_all.h.

Referenced by scsi_print_inquiry().

7.17.1.242 #define T_DIRECT 0x00

Definition at line 517 of file scsi_all.h.

Referenced by daasync(), scsi_command_string(), scsi_op_desc(), scsi_print_inquiry(), and scsi_sense_sbuf().

7.17.1.243 #define T_ENCLOSURE 0x0d

Definition at line 530 of file scsi_all.h.

Referenced by scsi_print_inquiry(), and ses_type().

7.17.1.244 #define T_FIXED 0

Definition at line 537 of file scsi_all.h.

7.17.1.245 #define T_NODEVICE 0x1f

Definition at line 533 of file scsi_all.h.

Referenced by scsi_print_inquiry().

7.17.1.246 #define T_OCRW 0x0f

Definition at line 532 of file scsi_all.h.

Referenced by scsi_print_inquiry().

7.17.1.247 #define T_OPTICAL 0x07

Definition at line 524 of file scsi_all.h.

Referenced by daasync(), and scsi_print_inquiry().

7.17.1.248 #define T_PRINTER 0x02

Definition at line 519 of file scsi_all.h.

Referenced by scsi_print_inquiry().

7.17.1.249 #define T_PROCESSOR 0x03

Definition at line 520 of file scsi_all.h.

Referenced by ptasync(), and scsi_print_inquiry().

7.17.1.250 #define T_RBC 0x0e

Definition at line 531 of file scsi_all.h.

Referenced by daasync(), daregister(), scsi_op_desc(), and scsi_print_inquiry().

7.17.1.251 #define T_REMOV 1

Definition at line 536 of file scsi_all.h.

7.17.1.252 #define T_SCANNER 0x06

Definition at line 523 of file scsi_all.h.

Referenced by scsi_print_inquiry().

7.17.1.253 #define T_SEQUENTIAL 0x01

Definition at line 518 of file scsi_all.h.

Referenced by saasync(), and scsi_print_inquiry().

7.17.1.254 #define T_STORARRAY 0x0c

Definition at line 529 of file scsi_all.h.

Referenced by scsi_print_inquiry().

7.17.1.255 #define T_WORM 0x04

Definition at line 521 of file scsi_all.h.

Referenced by cdasync(), and scsi_print_inquiry().

7.17.1.256 #define TEST_UNIT_READY 0x00

Definition at line 478 of file scsi_all.h.

Referenced by scsi_test_unit_ready().

7.17.1.257 #define WRITE_10 0x2a

Definition at line 494 of file scsi_all.h.

Referenced by cmd6workaround(), and scsi_read_write().

7.17.1.258 #define WRITE_12 0xaa

Definition at line 510 of file scsi_all.h.

Referenced by scsi_read_write().

7.17.1.259 #define WRITE_16 0x8a

Definition at line 505 of file scsi_all.h.

Referenced by scsi_read_write().

7.17.1.260 #define WRITE_6 0x0a

Definition at line 481 of file scsi_all.h.

Referenced by cmd6workaround(), and scsi_read_write().

7.17.1.261 #define WRITE_BUFFER 0x3b

Definition at line 497 of file scsi_all.h.

Referenced by perf_slotop(), wrbuf16(), and wrslot_stat().

7.17.2 Enumeration Type Documentation**7.17.2.1 enum [scsi_sense_action](#)**

Enumerator:

SS_NOP
SS_RETRY
SS_FAIL
SS_START
SS_TUR
SS_REQSENSE
SS_MASK

Definition at line 66 of file scsi_all.h.

7.17.2.2 enum [scsi_sense_action_qualifier](#)

Enumerator:

SSQ_NONE
SSQ_DECREMENT_COUNT
SSQ_MANY
SSQ_RANGE
SSQ_PRINT_SENSE
SSQ_MASK

Definition at line 82 of file scsi_all.h.

7.17.2.3 enum [scsi_sense_string_flags](#)

Enumerator:

SSS_FLAG_NONE

SSS_FLAG_PRINT_COMMAND

Definition at line 897 of file `scsi_all.h`.

7.17.3 Function Documentation

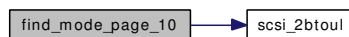
7.17.3.1 `static __inline void * find_mode_page_10 (struct scsi_mode_header_10 * mode_header)`
`[static]`

Definition at line 1230 of file `scsi_all.h`.

References `scsi_mode_header_10::blk_desc_len`, and `scsi_2btoul()`.

Referenced by `cdgetpage()`.

Here is the call graph for this function:



7.17.3.2 `static __inline void * find_mode_page_6 (struct scsi_mode_header_6 * mode_header)`
`[static]`

Definition at line 1219 of file `scsi_all.h`.

References `scsi_mode_header_6::blk_desc_len`.

Referenced by `cd6byteworkaround()`, `cdgetpage()`, `chdone()`, and `chgetparams()`.

7.17.3.3 `static __inline u_int32_t scsi_2btoul (u_int8_t * bytes)` `[static]`

Definition at line 1155 of file `scsi_all.h`.

Referenced by `cdgetmode()`, `cdsetmode()`, `chdone()`, `chgetelemstatus()`, `chgetparams()`, `copy_element_status()`, `copy_voltag()`, `find_mode_page_10()`, `samount()`, and `scsi_sense_sbuf()`.

7.17.3.4 `static __inline int32_t scsi_3btol (u_int8_t * bytes)` `[static]`

Definition at line 1176 of file `scsi_all.h`.

References `scsi_3btoul()`.

Here is the call graph for this function:



7.17.3.5 static __inline u_int32_t scsi_3btoul (u_int8_t * bytes) [static]

Definition at line 1165 of file scsi_all.h.

Referenced by cdreaddvdstructure(), cmd6workaround(), sagetparams(), samount(), and scsi_3btol().

7.17.3.6 static __inline u_int32_t scsi_4btoul (u_int8_t * bytes) [static]

Definition at line 1187 of file scsi_all.h.

Referenced by cddone(), cdsizel(), dadone(), dagetcapacity(), saerror(), sagetparams(), sardpos(), sasetparams(), and scsi_sense_sbuf().

7.17.3.7 static __inline uint64_t scsi_8btou64 (uint8_t * bytes) [static]

Definition at line 1199 of file scsi_all.h.

7.17.3.8 static __inline u_int64_t scsi_8btou64 (u_int8_t * bytes) [static]

Referenced by dadone(), and dagetcapacity().

7.17.3.9 u_int scsi_calc_syncparam (u_int period)

Definition at line 2346 of file scsi_all.c.

References period_factor, and scsi_syncrates.

7.17.3.10 u_int scsi_calc_syncrate (u_int period_factor)

Definition at line 2310 of file scsi_all.c.

References period, and scsi_syncrates.

Referenced by xpt_announce_periph().

7.17.3.11 char* scsi_cdb_string (u_int8_t * cdb_ptr, char * cdb_string, size_t len)

Definition at line 1734 of file scsi_all.c.

Referenced by scsi_command_string(), and xpt_action().

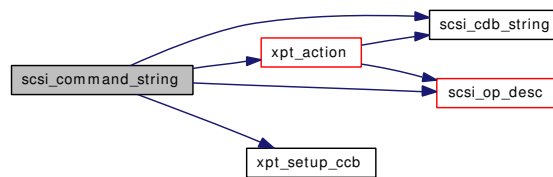
7.17.3.12 int scsi_command_string (struct ccb_scsiio * csio, struct sbuf * sb)

Definition at line 1826 of file scsi_all.c.

References CAM_CDB_POINTER, CAM_DEV_NOT_THERE, ccb::ccb_h, ccb_scsiio::ccb_h, ccb_getdev::ccb_h, cdb_t::cdb_bytes, ccb_scsiio::cdb_io, cdb_t::cdb_ptr, ccb_hdr::flags, ccb_hdr::func_code, ccb_getdev::inq_data, ccb_hdr::path, scsi_cdb_string(), SCSI_MAX_CDBLEN, scsi_op_desc(), ccb_hdr::status, T_DIRECT, xpt_action(), XPT_GDEV_TYPE, and xpt_setup_ccb().

Referenced by cam_error_string(), and scsi_sense_sbuf().

Here is the call graph for this function:



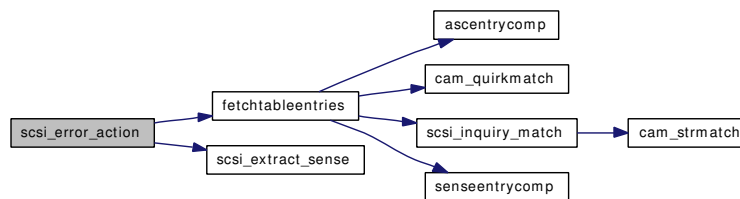
7.17.3.13 `scsi_sense_action` `scsi_error_action` (struct `ccb_scsiio` * `csio`, struct `scsi_inquiry_data` * `inq_data`, `u_int32_t` `sense_flags`)

Definition at line 1649 of file `scsi_all.c`.

References `asc_table_entry::action`, `sense_key_table_entry::action`, `fetchtableentries()`, `scsi_extract_sense()`, `ccb_scsiio::sense_data`, `sense_key_table_entry::sense_key`, `SF_NO_PRINT`, `SF_PRINT_ALWAYS`, `SF_QUIET_IR`, `SF_RETRY_UA`, `SS_ERRMASK`, `SS_FAIL`, `SS_MASK`, `SS_NOP`, `SS_RETRY`, `SSD_DEFERRED_ERROR`, `SSD_KEY_ILLEGAL_REQUEST`, `SSD_KEY_RECOVERED_ERROR`, `SSD_KEY_UNIT_ATTENTION`, `SSQ_DECREMENT_COUNT`, `SSQ_MASK`, and `SSQ_PRINT_SENSE`.

Referenced by `camperiphdone()`, and `camperiphscsisenseerror()`.

Here is the call graph for this function:



7.17.3.14 `static __inline void scsi_extract_sense` (struct `scsi_sense_data` * `sense`, `int` * `error_code`, `int` * `sense_key`, `int` * `asc`, `int` * `ascq`) [static]

Definition at line 1103 of file `scsi_all.h`.

References `scsi_sense_data::add_sense_code`, `scsi_sense_data::add_sense_code_qual`, `scsi_sense_data::error_code`, `scsi_sense_data::extra_len`, `scsi_sense_data::flags`, `SSD_ERRCODE`, and `SSD_KEY`.

Referenced by `camperiphdone()`, `cddone()`, `cderror()`, `daclose()`, `dadone()`, `dadump()`, `daerror()`, `dashutdown()`, `saerror()`, and `scsi_error_action()`.

7.17.3.15 `void scsi_inquiry` (struct `ccb_scsiio` * `csio`, `u_int32_t` `retries`, `void(*)`(struct `cam_periph` *, `union` `ccb` *) `cbfcn`, `u_int8_t` `tag_action`, `u_int8_t` * `inq_buf`, `u_int32_t` `inq_len`, `int` `evpd`, `u_int8_t` `page_code`, `u_int8_t` `sense_len`, `u_int32_t` `timeout`)

Definition at line 2422 of file `scsi_all.c`.

References `CAM_DIR_IN`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `INQUIRY`, and `SI_EVPD`.

Referenced by `probestart()`.

Here is the call graph for this function:



7.17.3.16 int scsi_inquiry_match (caddr_t inqbuffer, caddr_t table_entry)

Definition at line 2957 of file scsi_all.c.

References cam_strmatch(), scsi_inquiry_pattern::media_type, scsi_inquiry_pattern::product, scsi_inquiry_pattern::revision, SID_IS_REMOVABLE, SID_TYPE, SIP_MEDIA_FIXED, SIP_MEDIA_REMOVABLE, T_ANY, scsi_inquiry_pattern::type, and scsi_inquiry_pattern::vendor.

Referenced by cdregister(), daregister(), fetchtableentries(), saregister(), scsi_op_desc(), and xpt_find_quirk().

Here is the call graph for this function:



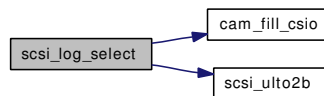
7.17.3.17 int scsi_interpret_sense (union ccb * ccb, u_int32_t sense_flags, u_int32_t * relsim_flags, u_int32_t * reduction, u_int32_t * timeout, scsi_sense_action error_action)

7.17.3.18 void scsi_log_select (struct ccb_scsiio * csio, u_int32_t retries, void(*) (struct cam_periph *, union ccb *) cbfcnp, u_int8_t tag_action, u_int8_t page_code, int save_pages, int pc_reset, u_int8_t * param_buf, u_int32_t param_len, u_int8_t sense_len, u_int32_t timeout)

Definition at line 2631 of file scsi_all.c.

References CAM_DIR_OUT, cam_fill_csio(), cdb_t::cdb_bytes, ccb_scsiio::cdb_io, LOG_SELECT, scsi_ulto2b(), SLS_PAGE_CODE, SLS_PCR, and SLS_SP.

Here is the call graph for this function:

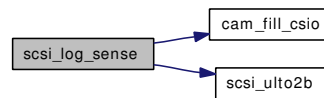


7.17.3.19 void scsi_log_sense (struct ccb_scsiio * csio, u_int32_t retries, void(*) (struct cam_periph *, union ccb *) cbfcnp, u_int8_t tag_action, u_int8_t page_code, u_int8_t page, int save_pages, int ppc, u_int32_t paramptr, u_int8_t * param_buf, u_int32_t param_len, u_int8_t sense_len, u_int32_t timeout)

Definition at line 2596 of file scsi_all.c.

References CAM_DIR_IN, cam_fill_csio(), cdb_t::cdb_bytes, ccb_scsiio::cdb_io, LOG_SENSE, scsi_ulto2b(), SLS_PPC, and SLS_SP.

Here is the call graph for this function:



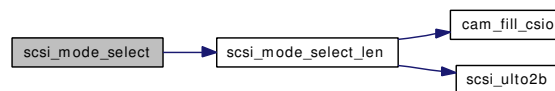
7.17.3.20 void `scsi_mode_select` (struct `ccb_scsiio` * *csio*, u_int32_t *retries*, void(*) (struct `cam_periph` *, union `ccb` *) *cbfcnp*, u_int8_t *tag_action*, int *scsi_page_fmt*, int *save_pages*, u_int8_t * *param_buf*, u_int32_t *param_len*, u_int8_t *sense_len*, u_int32_t *timeout*)

Definition at line 2526 of file `scsi_all.c`.

References `scsi_mode_select_len()`.

Referenced by `sasetparams()`.

Here is the call graph for this function:



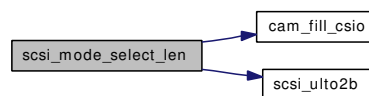
7.17.3.21 void `scsi_mode_select_len` (struct `ccb_scsiio` * *csio*, u_int32_t *retries*, void(*) (struct `cam_periph` *, union `ccb` *) *cbfcnp*, u_int8_t *tag_action*, int *scsi_page_fmt*, int *save_pages*, u_int8_t * *param_buf*, u_int32_t *param_len*, int *minimum_cmd_size*, u_int8_t *sense_len*, u_int32_t *timeout*)

Definition at line 2538 of file `scsi_all.c`.

References `CAM_DIR_OUT`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `MODE_SELECT_10`, `MODE_SELECT_6`, `scsi_ulito2b()`, `SMS_PF`, and `SMS_SP`.

Referenced by `cdsetmode()`, and `scsi_mode_select()`.

Here is the call graph for this function:



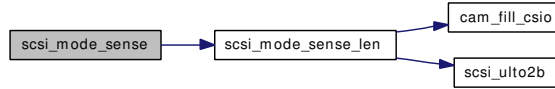
7.17.3.22 void `scsi_mode_sense` (struct `ccb_scsiio` * *csio*, u_int32_t *retries*, void(*) (struct `cam_periph` *, union `ccb` *) *cbfcnp*, u_int8_t *tag_action*, int *dbd*, u_int8_t *page_code*, u_int8_t *page*, u_int8_t * *param_buf*, u_int32_t *param_len*, u_int8_t *sense_len*, u_int32_t *timeout*)

Definition at line 2459 of file `scsi_all.c`.

References `scsi_mode_sense_len()`.

Referenced by `chgetparams()`, `chstart()`, `probstart()`, and `sagetparams()`.

Here is the call graph for this function:



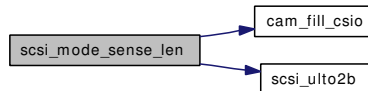
7.17.3.23 `void scsi_mode_sense_len (struct ccb_scsiio * csio, u_int32_t retries, void(*)(struct cam_periph *, union ccb *) cbfcnp, u_int8_t tag_action, int dbd, u_int8_t page_code, u_int8_t page, u_int8_t * param_buf, u_int32_t param_len, int minimum_cmd_size, u_int8_t sense_len, u_int32_t timeout)`

Definition at line 2472 of file `scsi_all.c`.

References `CAM_DIR_IN`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `MODE_SENSE_10`, `MODE_SENSE_6`, `scsi_ulito2b()`, and `SMS_DBD`.

Referenced by `cdgetmode()`, and `scsi_mode_sense()`.

Here is the call graph for this function:



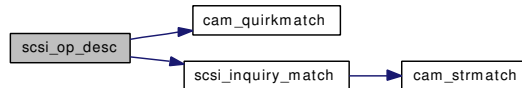
7.17.3.24 `const char* scsi_op_desc (u_int16_t opcode, struct scsi_inquiry_data * inq_data)`

Definition at line 659 of file `scsi_all.c`.

References `cam_quirkmatch()`, `op_table_entry::opcode`, `op_table_entry::opmask`, `scsi_inquiry_match()`, `scsi_op_codes`, `scsi_op_quirk_table`, `SID_TYPE`, `T_DIRECT`, and `T_RBC`.

Referenced by `scsi_command_string()`, and `xpt_action()`.

Here is the call graph for this function:



7.17.3.25 `void scsi_prevent (struct ccb_scsiio * csio, u_int32_t retries, void(*)(struct cam_periph *, union ccb *) cbfcnp, u_int8_t tag_action, u_int8_t action, u_int8_t sense_len, u_int32_t timeout)`

Definition at line 2667 of file `scsi_all.c`.

References `CAM_DIR_NONE`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, and `PREVENT_ALLOW`.

Referenced by `cdprevent()`, `daprevent()`, and `saprevent()`.

Here is the call graph for this function:



7.17.3.26 void `scsi_print_inquiry` (struct `scsi_inquiry_data` * `inq_data`)

Definition at line 2190 of file `scsi_all.c`.

References `cam_strvis()`, `scsi_inquiry_data::product`, `scsi_inquiry_data::revision`, `SCSI_REV_CCS`, `SID_ANSI_REV`, `SID_IS_REMOVABLE`, `SID_QUAL`, `SID_QUAL_BAD_LU`, `SID_QUAL_IS_VENDOR_UNIQUE`, `SID_QUAL_LU_CONNECTED`, `SID_QUAL_LU_OFFLINE`, `SID_QUAL_RSVD`, `SID_TYPE`, `T_CDROM`, `T_CHANGER`, `T_COMM`, `T_DIRECT`, `T_ENCLOSURE`, `T_NODEVICE`, `T_OCRW`, `T_OPTICAL`, `T_PRINTER`, `T_PROCESSOR`, `T_RBC`, `T_SCANNER`, `T_SEQUENTIAL`, `T_STORARRAY`, `T_WORM`, and `scsi_inquiry_data::vendor`.

Referenced by `xpt_announce_periph()`.

Here is the call graph for this function:



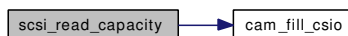
7.17.3.27 void `scsi_read_capacity` (struct `ccb_scsiio` * `csio`, `u_int32_t` `retries`, `void(*)`(struct `cam_periph` *, `union ccb` *) `cbfcnp`, `u_int8_t` `tag_action`, struct `scsi_read_capacity_data` *, `u_int8_t` `sense_len`, `u_int32_t` `timeout`)

Definition at line 2693 of file `scsi_all.c`.

References `CAM_DIR_IN`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, and `READ_CAPACITY`.

Referenced by `cdsize()`, `cdstart()`, `dagetcapacity()`, and `dastart()`.

Here is the call graph for this function:



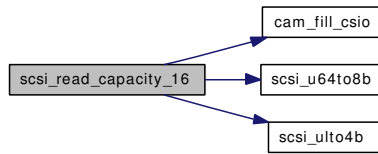
7.17.3.28 void `scsi_read_capacity_16` (struct `ccb_scsiio` * `csio`, `uint32_t` `retries`, `void(*)`(struct `cam_periph` *, `union ccb` *) `cbfcnp`, `uint8_t` `tag_action`, `uint64_t` `lba`, `int` `reladr`, `int` `pmi`, struct `scsi_read_capacity_data_long` * `rcap_buf`, `uint8_t` `sense_len`, `uint32_t` `timeout`)

Definition at line 2718 of file `scsi_all.c`.

References `CAM_DIR_IN`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `scsi_u64to8b()`, `scsi_u16to4b()`, `SERVICE_ACTION_IN`, `SRC16_PMI`, `SRC16_RELADR`, and `SRC16_SERVICE_ACTION`.

Referenced by `dagetcapacity()`, and `dastart()`.

Here is the call graph for this function:



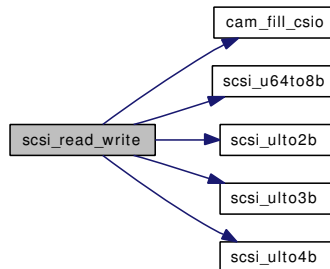
7.17.3.29 void `scsi_read_write` (struct `ccb_scsiio` * *csio*, u_int32_t *retries*, void(*) (struct `cam_periph` *, union `ccb` *) *cbfcnp*, u_int8_t *tag_action*, int *readop*, u_int8_t *byte2*, int *minimum_cmd_size*, u_int64_t *lba*, u_int32_t *block_count*, u_int8_t * *data_ptr*, u_int32_t *dxfer_len*, u_int8_t *sense_len*, u_int32_t *timeout*)

Definition at line 2808 of file `scsi_all.c`.

References `scsi_rw_16::addr`, `scsi_rw_12::addr`, `scsi_rw_10::addr`, `scsi_rw_6::addr`, `scsi_rw_16::byte2`, `scsi_rw_12::byte2`, `scsi_rw_10::byte2`, `CAM_DEBUG`, `CAM_DEBUG_SUBTRACE`, `CAM_DIR_IN`, `CAM_DIR_OUT`, `cam_fill_csio()`, `ccb_scsiio::ccb_h`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `scsi_rw_16::control`, `scsi_rw_12::control`, `scsi_rw_10::control`, `scsi_rw_6::control`, `scsi_rw_16::length`, `scsi_rw_12::length`, `scsi_rw_10::length`, `scsi_rw_6::length`, `scsi_rw_16::opcode`, `scsi_rw_12::opcode`, `scsi_rw_10::opcode`, `scsi_rw_6::opcode`, `ccb_hdr::path`, `READ_10`, `READ_12`, `READ_16`, `READ_6`, `scsi_rw_16::reserved`, `scsi_rw_12::reserved`, `scsi_rw_10::reserved`, `scsi_u64to8b()`, `scsi_ulto2b()`, `scsi_ulto3b()`, `scsi_ulto4b()`, `WRITE_10`, `WRITE_12`, `WRITE_16`, and `WRITE_6`.

Referenced by `cdstart()`, `dadump()`, and `dastart()`.

Here is the call graph for this function:

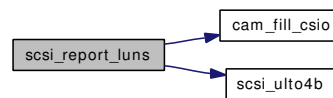


7.17.3.30 void `scsi_report_luns` (struct `ccb_scsiio` * *csio*, u_int32_t *retries*, void(*) (struct `cam_periph` *, union `ccb` *) *cbfcnp*, u_int8_t *tag_action*, u_int8_t *select_report*, struct `scsi_report_luns_data` * *rpl_buf*, u_int32_t *alloc_len*, u_int8_t *sense_len*, u_int32_t *timeout*)

Definition at line 2750 of file `scsi_all.c`.

References `CAM_DIR_IN`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `REPORT_LUNS`, and `scsi_ulto4b()`.

Here is the call graph for this function:



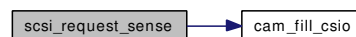
7.17.3.31 void **scsi_request_sense** (struct **ccb_scsiio** * *csio*, u_int32_t *retries*, void(*) (struct **cam_periph** *, union **ccb** *) *cbfcn*, void * *data_ptr*, u_int8_t *dxfer_len*, u_int8_t *tag_action*, u_int8_t *sense_len*, u_int32_t *timeout*)

Definition at line 2397 of file `scsi_all.c`.

References `CAM_DIR_IN`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, and `REQUEST_SENSE`.

Referenced by `camperiphscsisenseerror()`.

Here is the call graph for this function:



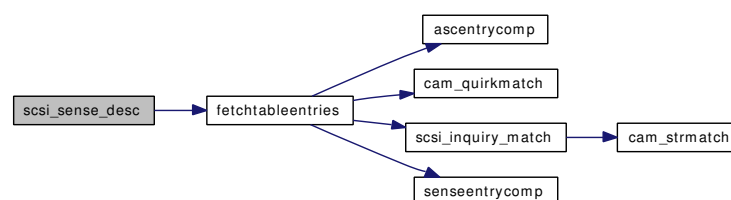
7.17.3.32 `__BEGIN_DECLS` void **scsi_sense_desc** (int *sense_key*, int *asc*, int *ascq*, struct **scsi_inquiry_data** * *inq_data*, const char ** *sense_key_desc*, const char ** *asc_desc*)

Definition at line 1619 of file `scsi_all.c`.

References `asc_table_entry::desc`, `sense_key_table_entry::desc`, and `fetchtableentries()`.

Referenced by `cddone()`, `dadone()`, and `scsi_sense_sbuf()`.

Here is the call graph for this function:



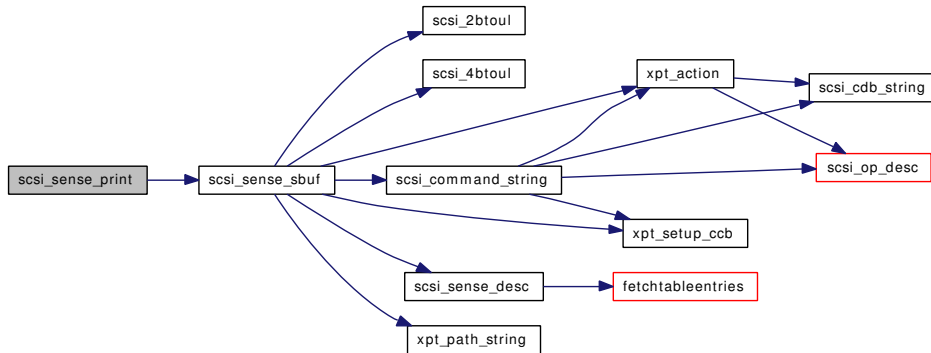
7.17.3.33 void **scsi_sense_print** (struct **ccb_scsiio** * *csio*)

Definition at line 2146 of file `scsi_all.c`.

References `scsi_sense_sbuf()`, and `SSS_FLAG_PRINT_COMMAND`.

Referenced by `camperiphdone()`, `camperiphscsisenseerror()`, `cddone()`, `chdone()`, `chgetparams()`, `daclose()`, `dadone()`, `dadump()`, `dashutdown()`, and `sagetparams()`.

Here is the call graph for this function:



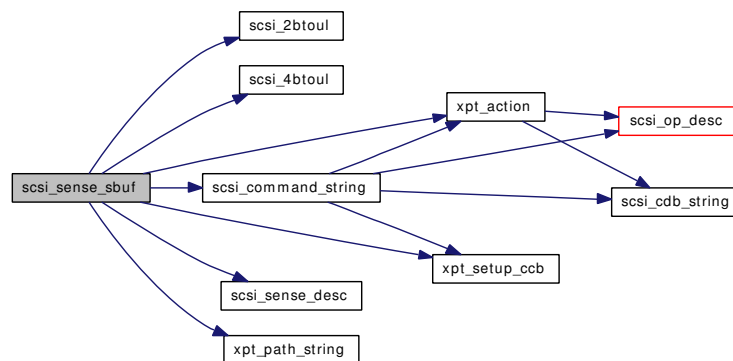
7.17.3.34 int scsi_sense_sbuf (struct ccb_scsiio * csio, struct sbuf * sb, scsi_sense_string_flags flags)

Definition at line 1885 of file scsi_all.c.

References `scsi_sense_data::add_sense_code`, `scsi_sense_data::add_sense_code_qual`, `CAM_CDB_PHYS`, `CAM_DEV_NOT_THERE`, `CAM_SENSE_PHYS`, `CAM_SENSE_PTR`, `ccb::ccb_h`, `ccb_getdev::ccb_h`, `ccb_scsiio::ccb_h`, `scsi_sense_data::cmd_spec_info`, `scsi_sense_data::error_code`, `scsi_sense_data::extra_len`, `scsi_sense_data::flags`, `ccb_hdr::flags`, `scsi_sense_data::fru`, `ccb_hdr::func_code`, `scsi_sense_data::info`, `ccb_getdev::inq_data`, `ccb_hdr::path`, `scsi_2btoul()`, `scsi_4btoul()`, `scsi_command_string()`, `scsi_sense_desc()`, `ccb_scsiio::sense_data`, `scsi_sense_data::sense_key_spec`, `SSD_CURRENT_ERROR`, `SSD_DEFERRED_ERROR`, `SSD_ERRCODE`, `SSD_ERRCODE_VALID`, `SSD_ILI`, `SSD_KEY`, `SSD_KEY_BLANK_CHECK`, `SSD_KEY_DATA_PROTECT`, `SSD_KEY_HARDWARE_ERROR`, `SSD_KEY_ILLEGAL_REQUEST`, `SSD_KEY_MEDIUM_ERROR`, `SSD_KEY_NOT_READY`, `SSD_KEY_RECOVERED_ERROR`, `SSD_KEY_UNIT_ATTENTION`, `SSD_SCS_VALID`, `SSS_FLAG_PRINT_COMMAND`, `ccb_hdr::status`, `T_DIRECT`, `xpt_action()`, `XPT_GDEV_TYPE`, `xpt_path_string()`, and `xpt_setup_ccb()`.

Referenced by `cam_error_string()`, `scsi_sense_print()`, and `scsi_sense_string()`.

Here is the call graph for this function:

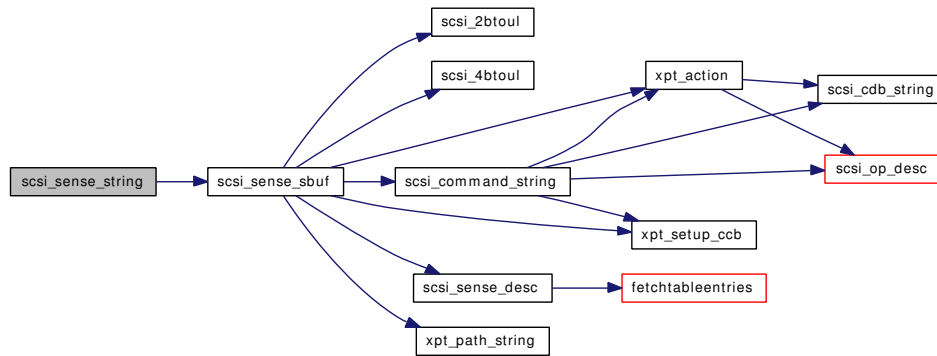


7.17.3.35 char* scsi_sense_string (struct ccb_scsiio * csio, char * str, int str_len)

Definition at line 2122 of file scsi_all.c.

References `scsi_sense_sbuf()`, and `SSS_FLAG_PRINT_COMMAND`.

Here is the call graph for this function:



7.17.3.36 `void scsi_start_stop (struct ccb_scsiio * csio, u_int32_t retries, void(*) (struct cam_periph *, union ccb *) cbfcn, u_int8_t tag_action, int start, int load_eject, int immediate, u_int8_t sense_len, u_int32_t timeout)`

Definition at line 2917 of file `scsi_all.c`.

References `CAM_DIR_NONE`, `cam_fill_csio()`, `CAM_HIGH_POWER`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `SSS_IMMED`, `SSS_LOEJ`, `SSS_START`, and `START_STOP_UNIT`.

Referenced by `camperiphscsisenseerror()`, `cdstartunit()`, and `cdstopunit()`.

Here is the call graph for this function:



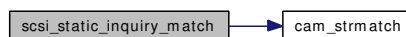
7.17.3.37 `int scsi_static_inquiry_match (caddr_t inqbuffer, caddr_t table_entry)`

Definition at line 2984 of file `scsi_all.c`.

References `cam_strmatch()`, `scsi_static_inquiry_pattern::media_type`, `scsi_static_inquiry_pattern::product`, `scsi_static_inquiry_pattern::revision`, `SID_IS_REMOVABLE`, `SID_TYPE`, `SIP_MEDIA_FIXED`, `SIP_MEDIA_REMOVABLE`, `T_ANY`, `scsi_static_inquiry_pattern::type`, and `scsi_static_inquiry_pattern::vendor`.

Referenced by `xptdevicematch()`.

Here is the call graph for this function:



7.17.3.38 `const char* scsi_status_string (struct ccb_scsiio * csio)`

Definition at line 1789 of file `scsi_all.c`.

References `ccb_scsiio::scsi_status`, `SCSI_STATUS_ACA_ACTIVE`, `SCSI_STATUS_BUSY`, `SCSI_STATUS_CHECK_COND`, `SCSI_STATUS_CMD_TERMINATED`, `SCSI_STATUS_INTERMED`, `SCSI_STATUS_INTERMED_COND_MET`, `SCSI_STATUS_OK`, `SCSI_STATUS_QUEUE_FULL`, `SCSI_STATUS_RESERV_CONFLICT`, and `SCSI_STATUS_TASK_ABORTED`.

Referenced by `cam_error_string()`, and `cddone()`.

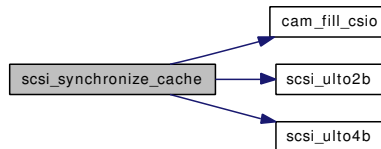
7.17.3.39 `void scsi_synchronize_cache (struct ccb_scsiio * csio, u_int32_t retries, void(*)(struct cam_periph *, union ccb *) cbfcnp, u_int8_t tag_action, u_int32_t begin_lba, u_int16_t lb_count, u_int8_t sense_len, u_int32_t timeout)`

Definition at line 2781 of file `scsi_all.c`.

References `CAM_DIR_NONE`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `scsi_ulto2b()`, `scsi_ulto4b()`, and `SYNCHRONIZE_CACHE`.

Referenced by `daclose()`, `dadump()`, `dashutdown()`, and `dastart()`.

Here is the call graph for this function:



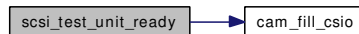
7.17.3.40 `void scsi_test_unit_ready (struct ccb_scsiio * csio, u_int32_t retries, void(*)(struct cam_periph *, union ccb *) cbfcnp, u_int8_t tag_action, u_int8_t sense_len, u_int32_t timeout)`

Definition at line 2374 of file `scsi_all.c`.

References `CAM_DIR_NONE`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, and `TEST_UNIT_READY`.

Referenced by `camperiphscsisenseerror()`, `probstart()`, and `samount()`.

Here is the call graph for this function:



7.17.3.41 `static __inline void scsi_u64to8b (u_int64_t val, u_int8_t * bytes) [static]`

Definition at line 1141 of file `scsi_all.h`.

Referenced by `scsi_read_capacity_16()`, and `scsi_read_write()`.

7.17.3.42 `static __inline void scsi_ulto2b (u_int32_t val, u_int8_t * bytes) [static]`

Definition at line 1114 of file `scsi_all.h`.

Referenced by `cd6byteworkaround()`, `cdplay()`, `cdreadsubchannel()`, `cdsendkey()`, `cdsetmode()`, `cdset-speed()`, `cmd6workaround()`, `scsi_exchange_medium()`, `scsi_log_select()`, `scsi_log_sense()`, `scsi_mode_select_len()`, `scsi_mode_sense_len()`, `scsi_move_medium()`, `scsi_position_to_element()`, `scsi_read_dvd_structure()`, `scsi_read_element_status()`, `scsi_read_write()`, `scsi_report_key()`, `scsi_send_key()`, `scsi_send_volume_tag()`, and `scsi_synchronize_cache()`.

7.17.3.43 `static __inline void scsi_ulto3b (u_int32_t val, u_int8_t * bytes)` [static]

Definition at line 1122 of file `scsi_all.h`.

Referenced by `sasetparams()`, `scsi_read_element_status()`, `scsi_read_write()`, `scsi_sa_read_write()`, `scsi_send_receive()`, `scsi_space()`, and `scsi_write_filemarks()`.

7.17.3.44 `static __inline void scsi_ulto4b (u_int32_t val, u_int8_t * bytes)` [static]

Definition at line 1131 of file `scsi_all.h`.

Referenced by `cdplay()`, `cmd6workaround()`, `sasetparams()`, `scsi_read_capacity_16()`, `scsi_read_dvd_structure()`, `scsi_read_write()`, `scsi_report_key()`, `scsi_report_luns()`, `scsi_set_position()`, and `scsi_synchronize_cache()`.

7.17.4 Variable Documentation

7.17.4.1 `int scsi_delay`

Definition at line 90 of file `scsi_all.c`.

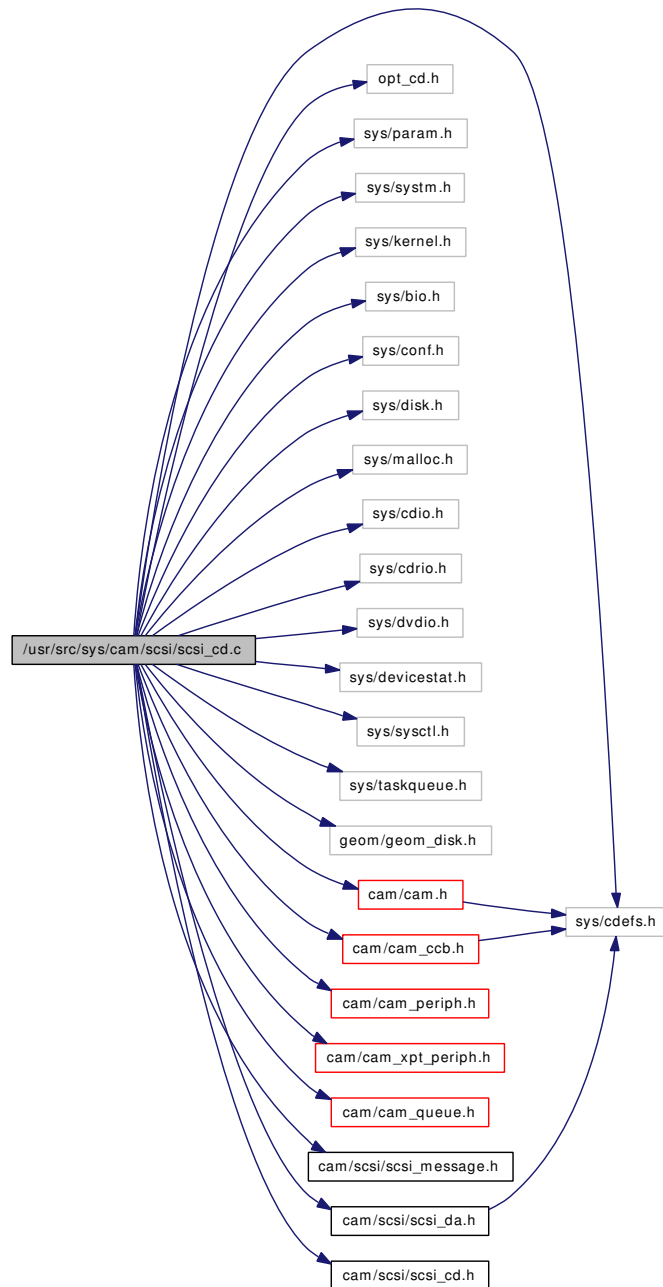
Referenced by `cam_periph_async()`, `proberegister()`, `set_scsi_delay()`, `sysctl_scsi_delay()`, and `xpt_config()`.

7.17.4.2 `const char* scsi_sense_key_text[]`

7.18 /usr/src/sys/cam/scsi/scsi_cd.c File Reference

```
#include <sys/cdefs.h>
#include "opt_cd.h"
#include <sys/param.h>
#include <sys/system.h>
#include <sys/kernel.h>
#include <sys/bio.h>
#include <sys/conf.h>
#include <sys/disk.h>
#include <sys/malloc.h>
#include <sys/cdio.h>
#include <sys/cdrpio.h>
#include <sys/dvdio.h>
#include <sys/devicestat.h>
#include <sys/sysctl.h>
#include <sys/taskqueue.h>
#include <geom/geom_disk.h>
#include <cam/cam.h>
#include <cam/cam_ccb.h>
#include <cam/cam_periph.h>
#include <cam/cam_xpt_periph.h>
#include <cam/cam_queue.h>
#include <cam/scsi/scsi_message.h>
#include <cam/scsi/scsi_da.h>
#include <cam/scsi/scsi_cd.h>
```

Include dependency graph for scsi_cd.c:



Data Structures

- struct [cd_params](#)
- struct [cd_tocdata](#)
- struct [cd_toc_single](#)
- struct [cd_softc](#)
- struct [cd_page_sizes](#)
- struct [cd_quirk_entry](#)
- struct [cdchanger](#)

Defines

- #define LEADOUT 0xaa
- #define ccb_state ppriv_field0
- #define ccb_bp ppriv_ptr1
- #define CHANGER_MIN_BUSY_SECONDS 5
- #define CHANGER_MAX_BUSY_SECONDS 15

Enumerations

- enum cd_quirks {
CD_Q_NONE = 0x00, CD_Q_NO_TOUCH = 0x01, CD_Q_BCD_TRACKS = 0x02, CD_Q_NO_CHANGER = 0x04,
CD_Q_CHANGER = 0x08, CD_Q_10_BYTE_ONLY = 0x10 }
- enum cd_flags {
CD_FLAG_INVALID = 0x0001, CD_FLAG_NEW_DISC = 0x0002, CD_FLAG_DISC_LOCKED = 0x0004, CD_FLAG_DISC_REMOVABLE = 0x0008,
CD_FLAG_TAGGED_QUEUING = 0x0010, CD_FLAG_CHANGER = 0x0040, CD_FLAG_ACTIVE = 0x0080, CD_FLAG_SCHED_ON_COMP = 0x0100,
CD_FLAG_RETRY_UA = 0x0200, CD_FLAG_VALID_MEDIA = 0x0400, CD_FLAG_VALID_TOC = 0x0800, CD_FLAG_SCTX_INIT = 0x1000 }
- enum cd_ccb_state {
CD_CCB_PROBE = 0x01, CD_CCB_BUFFER_IO = 0x02, CD_CCB_WAITING = 0x03, CD_CCB_TYPE_MASK = 0x0F,
CD_CCB_RETRY_UA = 0x10 }
- enum cd_changer_flags { CHANGER_TIMEOUT_SCHED = 0x01, CHANGER_SHORT_TIMEOUT_SCHED = 0x02, CHANGER_MANUAL_CALL = 0x04, CHANGER_NEED_TIMEOUT = 0x08 }
- enum cd_state { CD_STATE_PROBE, CD_STATE_NORMAL }

Functions

- `__FBSDID` ("FreeBSD: src/sys/cam/scsi/scsi_cd.c,v 1.97 2006/12/05 07:45:27 mjakob Exp \$")
- static void `cdasync` (void *callback_arg, u_int32_t code, struct `cam_path` *path, void *arg)
- static int `cdcmdsizesysctl` (SYSCTL_HANDLER_ARGS)
- static void `cdshorttimeout` (void *arg)
- static void `cdschedule` (struct `cam_periph` *periph, int priority)
- static void `cdrunchangerqueue` (void *arg)
- static void `cdchangerschedule` (struct `cd_softc` *softc)
- static int `cdrunccb` (union `ccb` *ccb, int(*error_routine)(union `ccb` *ccb, u_int32_t `cam_flags`, u_int32_t `sense_flags`), u_int32_t `cam_flags`, u_int32_t `sense_flags`)
- static union `ccb` * `cdgetccb` (struct `cam_periph` *periph, u_int32_t priority)
- static void `cddone` (struct `cam_periph` *periph, union `ccb` *start_ccb)
- static union `cd_pages` * `cdgetpage` (struct `cd_mode_params` *mode_params)
- static int `cdgetpagesize` (int page_num)
- static void `cdprevent` (struct `cam_periph` *periph, int action)
- static int `cdcheckmedia` (struct `cam_periph` *periph)
- static int `cdsize` (struct `cam_periph` *periph, u_int32_t *size)

- static int `cd6byteworkaround` (union `ccb` *ccb)
- static int `cderror` (union `ccb` *ccb, u_int32_t `cam_flags`, u_int32_t `sense_flags`)
- static int `cdreadtoc` (struct `cam_periph` *periph, u_int32_t `mode`, u_int32_t `start`, u_int8_t *`data`, u_int32_t `len`, u_int32_t `sense_flags`)
- static int `cdgetmode` (struct `cam_periph` *periph, struct `cd_mode_params` *`data`, u_int32_t `page`)
- static int `cdsetmode` (struct `cam_periph` *periph, struct `cd_mode_params` *`data`)
- static int `cdplay` (struct `cam_periph` *periph, u_int32_t `blk`, u_int32_t `len`)
- static int `cdreadsubchannel` (struct `cam_periph` *periph, u_int32_t `mode`, u_int32_t `format`, int `track`, struct `cd_sub_channel_info` *`data`, u_int32_t `len`)
- static int `cdplaymsf` (struct `cam_periph` *periph, u_int32_t `startm`, u_int32_t `starts`, u_int32_t `startf`, u_int32_t `endm`, u_int32_t `ends`, u_int32_t `endf`)
- static int `cdplaytracks` (struct `cam_periph` *periph, u_int32_t `strack`, u_int32_t `sindex`, u_int32_t `etrack`, u_int32_t `eindex`)
- static int `cdpause` (struct `cam_periph` *periph, u_int32_t `go`)
- static int `cdstopunit` (struct `cam_periph` *periph, u_int32_t `eject`)
- static int `cdstartunit` (struct `cam_periph` *periph, int `load`)
- static int `cdsetspeed` (struct `cam_periph` *periph, u_int32_t `rdspeed`, u_int32_t `wrspeed`)
- static int `cdreportkey` (struct `cam_periph` *periph, struct `dvd_authinfo` *`authinfo`)
- static int `cdsendkey` (struct `cam_periph` *periph, struct `dvd_authinfo` *`authinfo`)
- static int `cdreaddvdstructure` (struct `cam_periph` *periph, struct `dvd_struct` *`dvdstruct`)
- `PERIPHDRIVER_DECLARE` (`cd`, `cddriver`)
- `SYSCTL_NODE` (`_kern_cam`, `OID_AUTO`, `cd`, `CTLFLAG_RD`, 0, "CAM CDROM driver")
- `SYSCTL_NODE` (`_kern_cam_cd`, `OID_AUTO`, `changer`, `CTLFLAG_RD`, 0, "CD Changer")
- `SYSCTL_INT` (`_kern_cam_cd_changer`, `OID_AUTO`, `min_busy_seconds`, `CTLFLAG_RW`, `&changer_min_busy_seconds`, 0, "Minimum changer scheduling quantum")
- `TUNABLE_INT` ("kern.cam.cd.changer.min_busy_seconds", `&changer_min_busy_seconds`)
- `SYSCTL_INT` (`_kern_cam_cd_changer`, `OID_AUTO`, `max_busy_seconds`, `CTLFLAG_RW`, `&changer_max_busy_seconds`, 0, "Maximum changer scheduling quantum")
- `TUNABLE_INT` ("kern.cam.cd.changer.max_busy_seconds", `&changer_max_busy_seconds`)
- static `STAILQ_HEAD` (`changerlist`, `cdchanger`)
- static void `cdoninvalidate` (struct `cam_periph` *periph)
- static void `cdcleanup` (struct `cam_periph` *periph)
- static void `cdsysctlinit` (void *`context`, int `pending`)
- static `cam_status` `cdregister` (struct `cam_periph` *periph, void *`arg`)
- static int `cdopen` (struct `disk` *`dp`)
- static int `cdclose` (struct `disk` *`dp`)
- static void `cdstrategy` (struct `bio` *`bp`)
- static void `cdstart` (struct `cam_periph` *periph, union `ccb` *`start_ccb`)
- static int `cdioctl` (struct `disk` *`dp`, u_long `cmd`, void *`addr`, int `flag`, struct `thread` *`td`)
- void `scsi_report_key` (struct `ccb_scsiio` *`csio`, u_int32_t `retries`, void(*`cbfcn`)(struct `cam_periph` *, union `ccb` *), u_int8_t `tag_action`, u_int32_t `lba`, u_int8_t `agid`, u_int8_t `key_format`, u_int8_t *`data_ptr`, u_int32_t `dxfer_len`, u_int8_t `sense_len`, u_int32_t `timeout`)
- void `scsi_send_key` (struct `ccb_scsiio` *`csio`, u_int32_t `retries`, void(*`cbfcn`)(struct `cam_periph` *, union `ccb` *), u_int8_t `tag_action`, u_int8_t `agid`, u_int8_t `key_format`, u_int8_t *`data_ptr`, u_int32_t `dxfer_len`, u_int8_t `sense_len`, u_int32_t `timeout`)
- void `scsi_read_dvd_structure` (struct `ccb_scsiio` *`csio`, u_int32_t `retries`, void(*`cbfcn`)(struct `cam_periph` *, union `ccb` *), u_int8_t `tag_action`, u_int32_t `address`, u_int8_t `layer_number`, u_int8_t `format`, u_int8_t `agid`, u_int8_t *`data_ptr`, u_int32_t `dxfer_len`, u_int8_t `sense_len`, u_int32_t `timeout`)

Variables

- static struct `cd_page_sizes` `cd_page_size_table` []
- static struct `cd_quirk_entry` `cd_quirk_table` []
- static `disk_open_t` `cdopen`
- static `disk_close_t` `cdcloses`
- static `disk_ioctl_t` `cdioctl`
- static `disk_strategy_t` `cdstrategy`
- static `periph_init_t` `cdinit`
- static `periph_ctor_t` `cdregister`
- static `periph_dtor_t` `cdcleanup`
- static `periph_start_t` `cdstart`
- static `periph_oninv_t` `cdoninvalidate`
- static struct `periph_driver` `cddriver`
- static int `num_changers`
- static int `changer_min_busy_seconds` = `CHANGER_MIN_BUSY_SECONDS`
- static int `changer_max_busy_seconds` = `CHANGER_MAX_BUSY_SECONDS`

7.18.1 Define Documentation

7.18.1.1 #define `ccb_bp` `ppriv_ptr1`

Definition at line 125 of file `scsi_cd.c`.

7.18.1.2 #define `ccb_state` `ppriv_field0`

Definition at line 124 of file `scsi_cd.c`.

7.18.1.3 #define `CHANGER_MAX_BUSY_SECONDS` 15

Definition at line 300 of file `scsi_cd.c`.

7.18.1.4 #define `CHANGER_MIN_BUSY_SECONDS` 5

Definition at line 297 of file `scsi_cd.c`.

7.18.1.5 #define `LEADOUT` `0xaa`

Definition at line 78 of file `scsi_cd.c`.

Referenced by `cdcheckmedia()`, and `cdioctl()`.

7.18.2 Enumeration Type Documentation

7.18.2.1 enum `cd_ccb_state`

Enumerator:

`CD_CCB_PROBE`

CD_CCB_BUFFER_IO
CD_CCB_WAITING
CD_CCB_TYPE_MASK
CD_CCB_RETRY_UA

Definition at line 109 of file scsi_cd.c.

7.18.2.2 enum [cd_changer_flags](#)

Enumerator:

CHANGER_TIMEOUT_SCHED
CHANGER_SHORT_TMOUT_SCHED
CHANGER_MANUAL_CALL
CHANGER_NEED_TIMEOUT

Definition at line 117 of file scsi_cd.c.

7.18.2.3 enum [cd_flags](#)

Enumerator:

CD_FLAG_INVALID
CD_FLAG_NEW_DISC
CD_FLAG_DISC_LOCKED
CD_FLAG_DISC_REMOVABLE
CD_FLAG_TAGGED_QUEUEING
CD_FLAG_CHANGER
CD_FLAG_ACTIVE
CD_FLAG_SCHED_ON_COMP
CD_FLAG_RETRY_UA
CD_FLAG_VALID_MEDIA
CD_FLAG_VALID_TOC
CD_FLAG_SCTX_INIT

Definition at line 94 of file scsi_cd.c.

7.18.2.4 enum [cd_quirks](#)

Enumerator:

CD_Q_NONE
CD_Q_NO_TOUCH
CD_Q_BCD_TRACKS
CD_Q_NO_CHANGER
CD_Q_CHANGER
CD_Q_10_BYTE_ONLY

Definition at line 85 of file scsi_cd.c.

7.18.2.5 enum `cd_state`

Enumerator:

`CD_STATE_PROBE`
`CD_STATE_NORMAL`

Definition at line 137 of file `scsi_cd.c`.

7.18.3 Function Documentation

7.18.3.1 `__FBSDID("$FreeBSD: src/sys/cam/scsi/scsi_cd.c, v 1.97 2006/12/05 07:45:27 mjacob Exp $")`

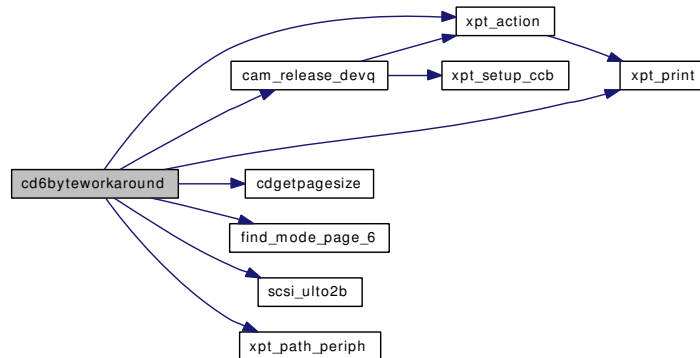
7.18.3.2 `static int cd6byteworkaround (union ccb * ccb) [static]`

Definition at line 2909 of file `scsi_cd.c`.

References `cd_mode_params::alloc_len`, `scsi_mode_header_6::blk_desc_len`, `scsi_mode_select_6::byte2`, `scsi_mode_sense_6::byte2`, `CAM_CDB_POINTER`, `CAM_DEV_QFRZN`, `cam_release_devq()`, `CAM_QUEUE_REQ`, `ccb::ccb_h`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `ccb_scsiio::cdb_len`, `cd_mode_params::cdb_size`, `cdgetpagesize()`, `scsi_mode_select_6::control`, `scsi_mode_sense_6::control`, `ccb::csio`, `ccb_scsiio::data_ptr`, `ccb_scsiio::dxfer_len`, `find_mode_page_6()`, `ccb_hdr::flags`, `cd_mode_params::mode_buf`, `MODE_SELECT_10`, `MODE_SELECT_6`, `MODE_SENSE_10`, `MODE_SENSE_6`, `scsi_mode_sense_6::page`, `scsi_mode_page_header::page_code`, `scsi_mode_page_header::page_length`, `cam_periph::path`, `ccb_hdr::path`, `scsi_ulto2b()`, `cam_periph::softc`, `ccb_hdr::status`, `xpt_action()`, `xpt_path_periph()`, and `xpt_print()`.

Referenced by `cderror()`.

Here is the call graph for this function:



7.18.3.3 `static void cdasync (void * callback_arg, u_int32_t code, struct cam_path * path, void * arg) [static]`

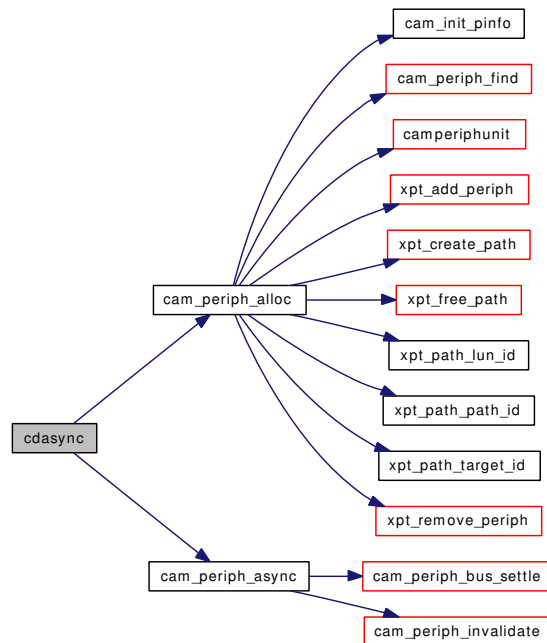
Definition at line 504 of file `scsi_cd.c`.

References `AC_BUS_RESET`, `AC_FOUND_DEVICE`, `AC_SENT_BDR`, `cam_periph_alloc()`, `cam_periph_async()`, `CAM_PERIPH_BIO`, `CAM_REQ_CMP`, `CAM_REQ_INPROG`, `ccb_getdev::ccb_h`,

CD_CCB_RETRY_UA, CD_FLAG_RETRY_UA, cd_softc::flags, ccb_getdev::inq_data, ccb_hdr::path, SID_TYPE, cam_periph::softc, T_CDROM, and T_WORM.

Referenced by cdoninvalidate(), cdregister(), and STAILQ_HEAD().

Here is the call graph for this function:



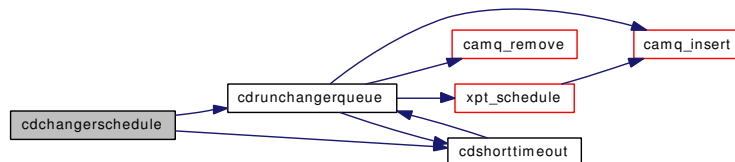
7.18.3.4 static void cdchangerschedule (struct cd_softc * softc) [static]

Definition at line 1265 of file scsi_cd.c.

References cd_softc::bio_queue, CD_FLAG_ACTIVE, CD_FLAG_SCHED_ON_COMP, cdrunchangerqueue(), cdshorttimeout(), CHANGER_MANUAL_CALL, CHANGER_NEED_TIMEOUT, CHANGER_SHORT_TMOUT_SCHED, CHANGER_TIMEOUT_SCHED, cdchanger::flags, cd_softc::flags, cdchanger::long_handle, and cdchanger::short_handle.

Referenced by cddone(), and cdrunccb().

Here is the call graph for this function:



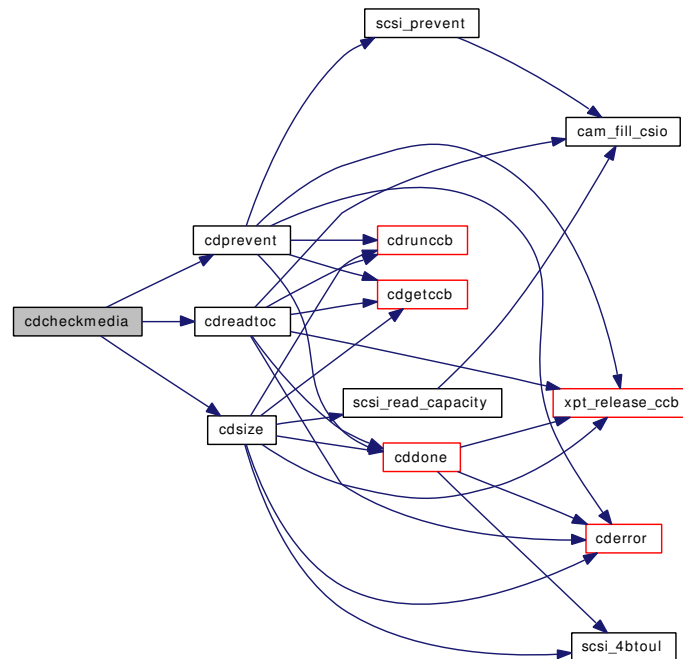
7.18.3.5 static int cdcheckmedia (struct cam_periph * periph) [static]

Definition at line 2724 of file scsi_cd.c.

References CD_FLAG_VALID_MEDIA, CD_FLAG_VALID_TOC, cdprevent(), cdreadtoc(), cdsizes(), cd_softc::flags, LEADOUT, PR_ALLOW, PR_PREVENT, SF_NO_PRINT, and cam_periph::softc.

Referenced by cdioctl(), cdopen(), and cdstrategy().

Here is the call graph for this function:

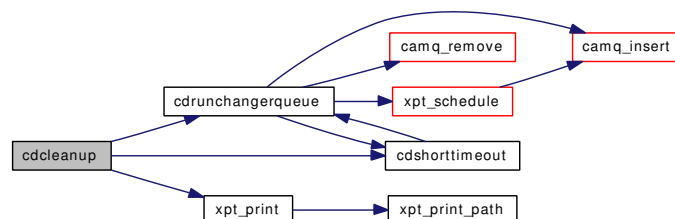


7.18.3.6 static void cdcleanup (struct cam_periph *periph) [static]

Definition at line 415 of file scsi_cd.c.

References CD_FLAG_ACTIVE, CD_FLAG_CHANGER, CD_FLAG_SCTX_INIT, cdchangerqueue(), cdshorttimeout(), CHANGER_MANUAL_CALL, CHANGER_SHORT_TMOUT_SCHED, CHANGER_TIMEOUT_SCHED, cd_softc::flags, cam_periph::path, cam_periph::softc, and xpt_print().

Here is the call graph for this function:

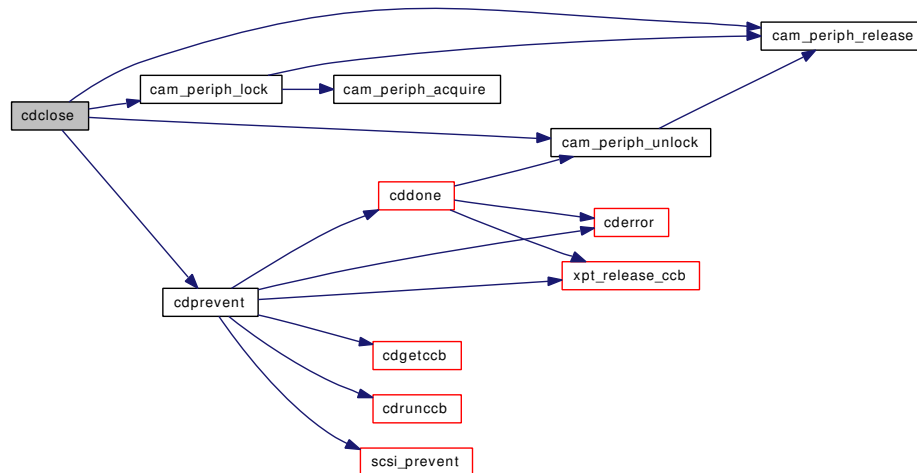


7.18.3.7 static int cdclose (struct disk * dp) [static]

Definition at line 1041 of file scsi_cd.c.

References `cam_periph_lock()`, `cam_periph_release()`, `cam_periph_unlock()`, `CD_FLAG_DISC_REMOVABLE`, `CD_FLAG_VALID_MEDIA`, `CD_FLAG_VALID_TOC`, `cdprevent()`, `cd_softc::flags`, `PR_ALLOW`, and `cam_periph::softc`.

Here is the call graph for this function:

**7.18.3.8 static int cdcmdsizesysctl (SYSCTL_HANDLER_ARGS) [static]**

Definition at line 611 of file scsi_cd.c.

Referenced by `cdsysctlinit()`.

7.18.3.9 static void cddone (struct cam_periph * periph, union ccb * start_ccb) [static]

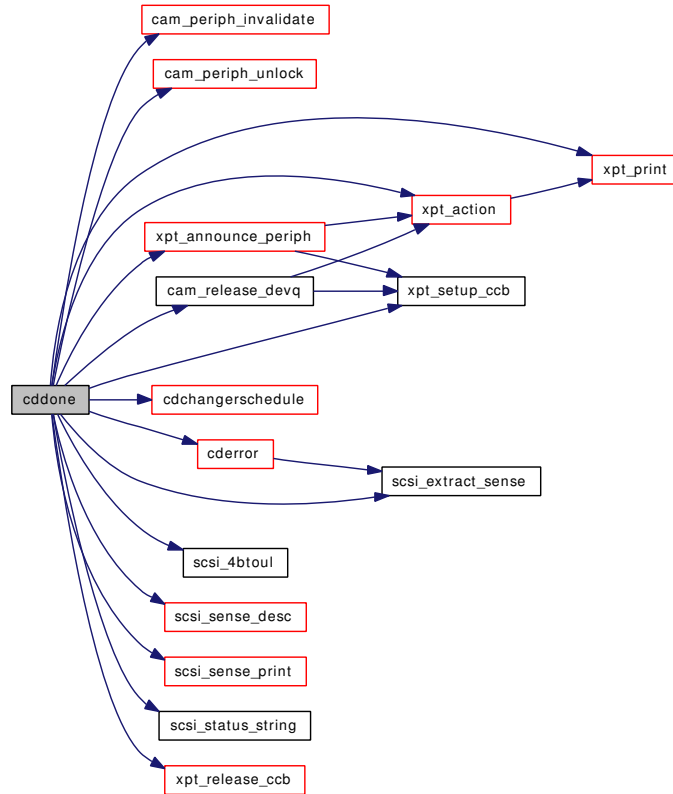
Definition at line 1589 of file scsi_cd.c.

References `scsi_read_capacity_data::addr`, `cd_softc::bio_queue`, `cd_params::blksize`, `CAM_AUTOSNS_VALID`, `CAM_DEBUG`, `CAM_DEBUG_TRACE`, `cam_periph_invalidate()`, `cam_periph_unlock()`, `cam_release_devq()`, `CAM_REQ_CMP`, `CAM_RETRY_SELTO`, `CAM_SCSI_STATUS_ERROR`, `CAM_SENSE_PHYS`, `CAM_SENSE_PTR`, `CAM_STATUS_MASK`, `ccb_hdr::cbfcnp`, `ccb_getdev::ccb_h`, `ccb::ccb_h`, `ccb_scsiio::ccb_h`, `CD_CCB_BUFFER_IO`, `CD_CCB_PROBE`, `CD_CCB_RETRY_UA`, `CD_CCB_TYPE_MASK`, `CD_CCB_WAITING`, `CD_FLAG_CHANGER`, `CD_STATE_NORMAL`, `cd_changerschedule()`, `cderror()`, `ccb::csio`, `ccb_scsiio::data_ptr`, `cd_params::disksize`, `scsi_sense_data::error_code`, `ccb_hdr::flags`, `cd_softc::flags`, `ccb_hdr::func_code`, `scsi_read_capacity_data::length`, `ccb_hdr::path`, `cam_periph::path`, `ccb_scsiio::resid`, `scsi_4btoul()`, `scsi_extract_sense()`, `scsi_sense_desc()`, `scsi_sense_print()`, `ccb_scsiio::scsi_status`, `SCSI_STATUS_BUSY`, `scsi_status_string()`, `ccb_scsiio::sense_data`, `SF_NO_PRINT`, `SF_RETRY_UA`, `SID_TYPE`, `cam_periph::softc`, `SSD_CURRENT_ERROR`, `cd_softc::state`, `ccb_hdr::status`, `T_CDROM`, `xpt_action()`, `xpt_announce_periph()`, `XPT_GDEV_TYPE`, `xpt_print()`, `xpt_release_ccb()`, and `xpt_setup_ccb()`.

Referenced by `cdgetmode()`, `cdpause()`, `cdplay()`, `cdplaymsf()`, `cdplaytracks()`, `cdprevent()`, `cdreadvd-structure()`, `cdreadsubchannel()`, `cdreadtoc()`, `cdreportkey()`, `cdsendkey()`, `cdsetmode()`, `cdsetspeed()`, `cd-`

size(), cdstart(), cdstartunit(), and cdstopunit().

Here is the call graph for this function:



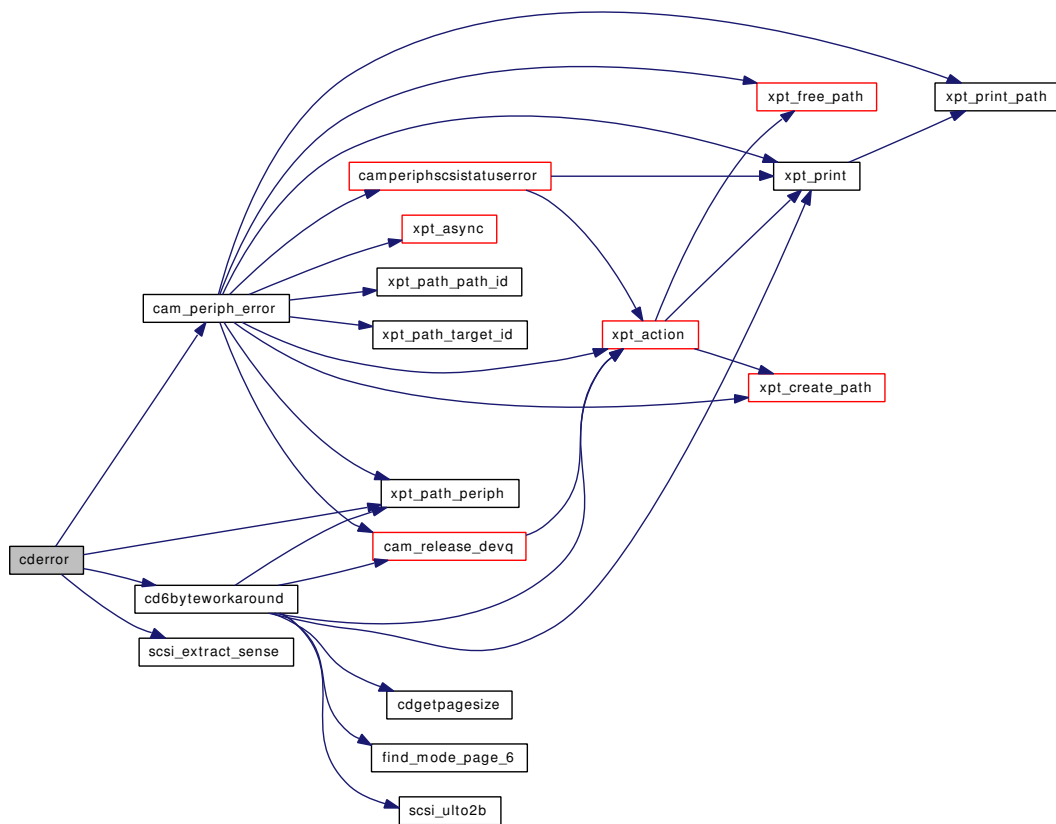
7.18.3.10 static int cderror (union **ccb** * ccb, u_int32_t cam_flags, u_int32_t sense_flags)
[static]

Definition at line 3058 of file scsi_cd.c.

References CAM_AUTOSNS_VALID, cam_periph_error(), CAM_REQ_INVALID, CAM_SCSI_STATUS_ERROR, CAM_SENSE_PHYS, CAM_SENSE_PTR, CAM_STATUS_MASK, ccb::ccb_h, cd6byteworkaround(), ccb::csio, ccb_hdr::flags, ccb_hdr::path, scsi_extract_sense(), ccb_scsiio::scsi_status, SCSI_STATUS_CHECK_COND, ccb_scsiio::sense_data, SF_RETRY_UA, cam_periph::softc, SSD_KEY_ILLEGAL_REQUEST, ccb_hdr::status, and xpt_path_periph().

Referenced by cddone(), cdgetmode(), cdiocctl(), cdpause(), cdplay(), cdplaymsf(), cdplaytracks(), cdprevent(), cdreadvdstructure(), cdreadsubchannel(), cdreadtoc(), cdreportkey(), cdsendkey(), cdsetmode(), cdsetspeed(), cdsizes(), cdstartunit(), and cdstopunit().

Here is the call graph for this function:



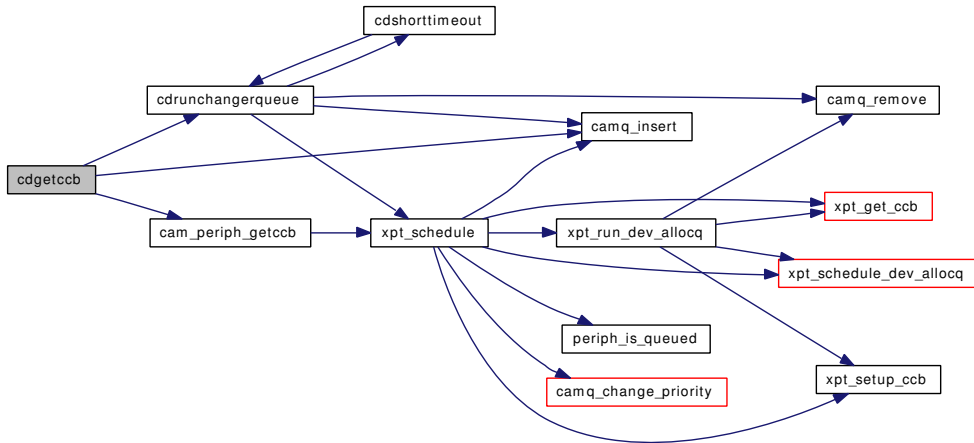
7.18.3.11 static union `ccb` * `cdgetccb` (struct `cam_periph` * `periph`, `u_int32_t` `priority`) [static]

Definition at line 1366 of file `scsi_cd.c`.

References `cam_periph_getccb()`, `CAM_UNQUEUED_INDEX`, `camq_insert()`, `CD_FLAG_ACTIVE`, `CD_FLAG_CHANGER`, `cdrunchangerqueue()`, `CHANGER_MANUAL_CALL`, `CHANGER_NEED_TIMEOUT`, `CHANGER_SHORT_TIMEOUT_SCHED`, `CHANGER_TIMEOUT_SCHED`, `cd_softc::flags`, `cam_pinfo::generation`, `cam_pinfo::index`, `cd_softc::pinfo`, `cam_pinfo::priority`, and `cam_periph::softc`.

Referenced by `cdgetmode()`, `cdpause()`, `cdplay()`, `cdplaymsf()`, `cdplaytracks()`, `cdprevent()`, `cdreadvd-structure()`, `cdreadsubchannel()`, `cdreadtoc()`, `cdreportkey()`, `cdsendkey()`, `cdsetmode()`, `cdsetspeed()`, `cdsize()`, `cdstartunit()`, and `cdstopunit()`.

Here is the call graph for this function:



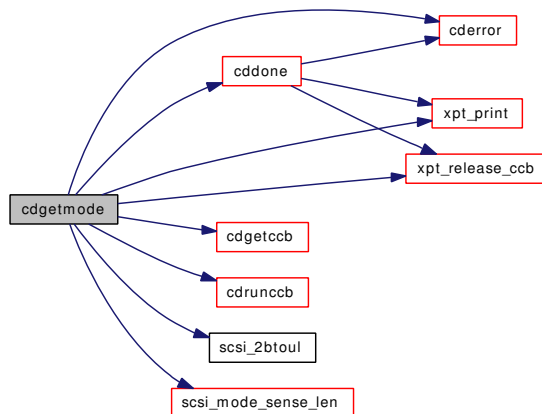
7.18.3.12 static int cdgetmode (struct **cam_periph** * *periph*, struct **cd_mode_params** * *data*, **u_int32_t** *page*) [static]

Definition at line 3207 of file `scsi_cd.c`.

References `cd_mode_params::alloc_len`, `CAM_RETRY_SELTO`, `cd_mode_params::cdb_size`, `cddone()`, `cderror()`, `cdgetccb()`, `cdrunccb()`, `ccb::csio`, `scsi_mode_header_6::data_length`, `scsi_mode_header_10::data_length`, `cd_mode_params::mode_buf`, `MSG_SIMPLE_Q_TAG`, `cam_periph::path`, `scsi_2btoul()`, `scsi_mode_sense_len()`, `SF_RETRY_UA`, `SMS_PAGE_CTRL_CURRENT`, `cam_periph::softc`, `SSD_FULL_SIZE`, `xpt_print()`, and `xpt_release_ccb()`.

Referenced by `cdioctl()`.

Here is the call graph for this function:



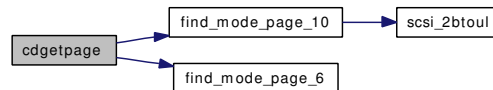
7.18.3.13 static union **cd_pages** * cdgetpage (struct **cd_mode_params** * *mode_params*) [static]

Definition at line 1874 of file `scsi_cd.c`.

References `cd_mode_params::cdb_size`, `find_mode_page_10()`, `find_mode_page_6()`, and `cd_mode_params::mode_buf`.

Referenced by `cdioctl()`.

Here is the call graph for this function:



7.18.3.14 `static int cdgetpagesize(int page_num)` [static]

Definition at line 1889 of file `scsi_cd.c`.

References `cd_page_size_table`.

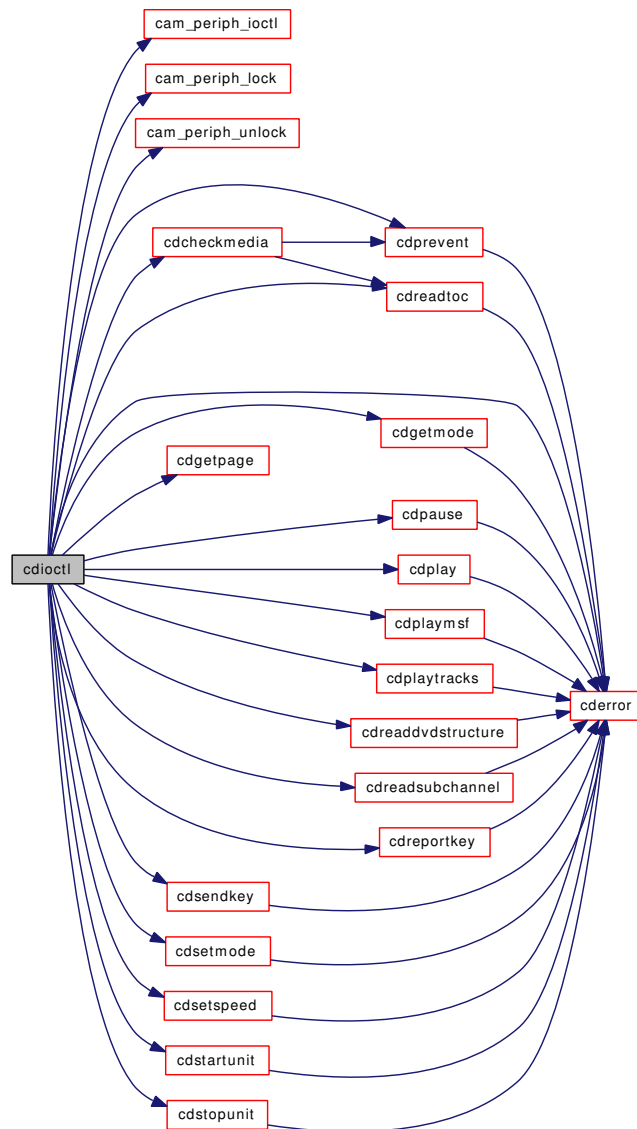
Referenced by `cd6byteworkaround()`.

7.18.3.15 `static int cdioctl(struct disk * dp, u_long cmd, void * addr, int flag, struct thread * td)` [static]

Definition at line 1903 of file `scsi_cd.c`.

References `cd_mode_params::alloc_len`, `cd_pages::audio`, `AUDIO_PAGE`, `CAM_DEBUG`, `CAM_DEBUG_SUBTRACE`, `CAM_DEBUG_TRACE`, `cam_periph_ioctl()`, `cam_periph_lock()`, `cam_periph_unlock()`, `CD_FLAG_VALID_MEDIA`, `CD_FLAG_VALID_TOC`, `CD_PA_IMMED`, `CD_PA_SOTC`, `cd-checkmedia()`, `cderror()`, `cdgetmode()`, `cdgetpage()`, `cdpause()`, `cdplay()`, `cdplaymsf()`, `cdplaytracks()`, `cdprevent()`, `cdreadvdstructure()`, `cdreadsubchannel()`, `cdreadtoc()`, `cdreportkey()`, `cdsendkey()`, `cdsetmode()`, `cdsetspeed()`, `cdstartunit()`, `cdstopunit()`, `CHANNEL_0`, `CHANNEL_1`, `cd_audio_page::port_control::channels`, `cd_audio_page::flags`, `cd_softc::flags`, `LEADOUT`, `LEFT_CHANNEL`, `LEFT_PORT`, `cd_mode_params::mode_buf`, `cam_periph::path`, `cd_audio_page::port`, `PR_ALLOW`, `PR_PREVENT`, `RIGHT_CHANNEL`, `RIGHT_PORT`, `cam_periph::softc`, and `cd_audio_page::port_control::volume`.

Here is the call graph for this function:

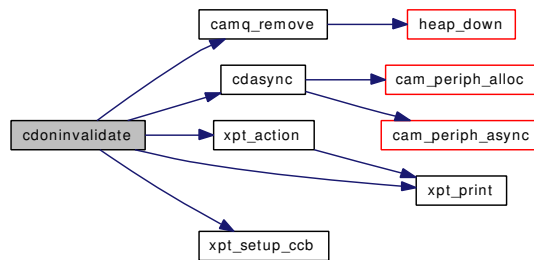


7.18.3.16 static void cdoninvalidate (struct **cam_periph** * *periph*) [static]

Definition at line 365 of file `scsi_cd.c`.

References `ccb_setasync::callback`, `ccb_setasync::callback_arg`, `CAM_UNQUEUED_INDEX`, `camq_remove()`, `ccb_setasync::ccb_h`, `CD_FLAG_CHANGER`, `CD_FLAG_INVALID`, `cdasync()`, `ccb_setasync::event_enable`, `ccb_hdr::func_code`, `cam_periph::path`, `cam_periph::softc`, `xpt_action()`, `xpt_print()`, `XPT_SASYNC_CB`, and `xpt_setup_ccb()`.

Here is the call graph for this function:

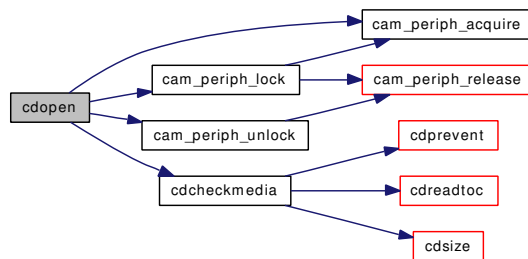


7.18.3.17 static int cdopen (struct disk * dp) [static]

Definition at line 994 of file scsi_cd.c.

References CAM_DEBUG, CAM_DEBUG_TRACE, cam_periph_acquire(), cam_periph_lock(), cam_periph_unlock(), CAM_REQ_CMP, CD_FLAG_INVALID, cdcheckmedia(), cd_softc::flags, cam_periph::path, and cam_periph::softc.

Here is the call graph for this function:



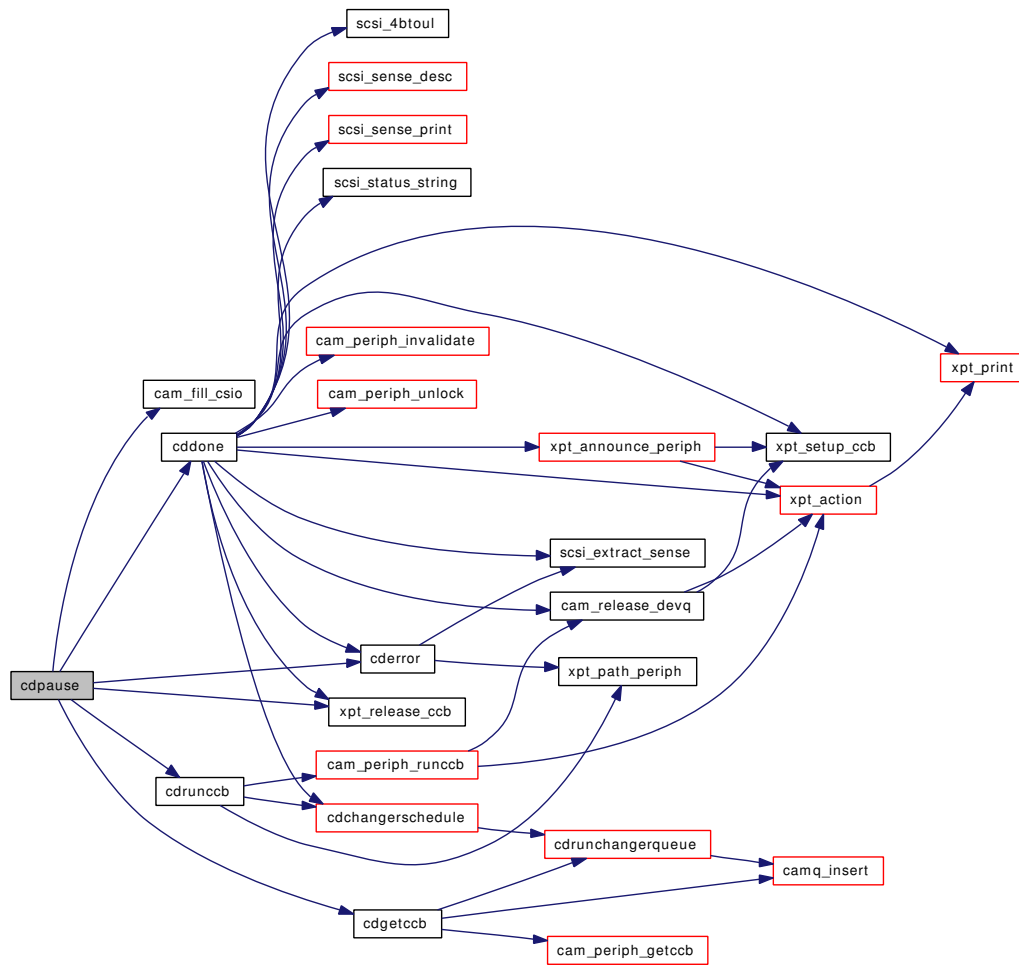
7.18.3.18 static int cdpause (struct cam_periph * periph, u_int32_t go) [static]

Definition at line 3545 of file scsi_cd.c.

References CAM_DIR_NONE, cam_fill_csio(), CAM_RETRY_SELTO, cdb_t::cdb_bytes, ccb_scsiio::cdb_io, cddone(), cderror(), cdgetccb(), cdrunccb(), ccb::csio, MSG_SIMPLE_Q_TAG, PAUSE, SF_RETRY_UA, SSD_FULL_SIZE, and xpt_release_ccb().

Referenced by cdiocctl().

Here is the call graph for this function:



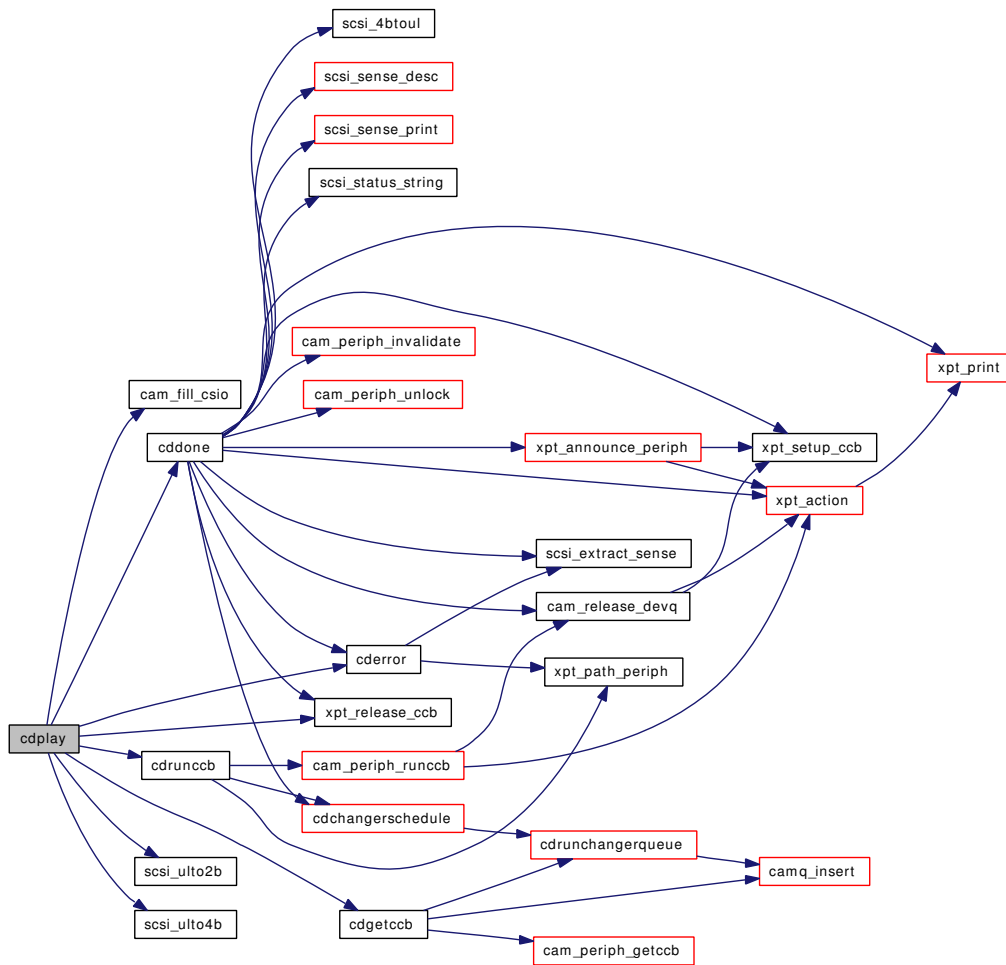
7.18.3.19 static int cdplay (struct cam_periph * periph, u_int32_t blk, u_int32_t len) [static]

Definition at line 3401 of file scsi_cd.c.

References CAM_DIR_NONE, cam_fill_csio(), CAM_RETRY_SELTO, cdb_t::cdb_bytes, ccb_-scsiio::cdb_io, cddone(), cderror(), cdgetccb(), cdrunccb(), ccb::csio, MSG_SIMPLE_Q_TAG, PLAY_10, PLAY_12, scsi_ulpto2b(), scsi_ulpto4b(), SF_RETRY_UA, SSD_FULL_SIZE, and xpt_release_ccb().

Referenced by cdioctl().

Here is the call graph for this function:



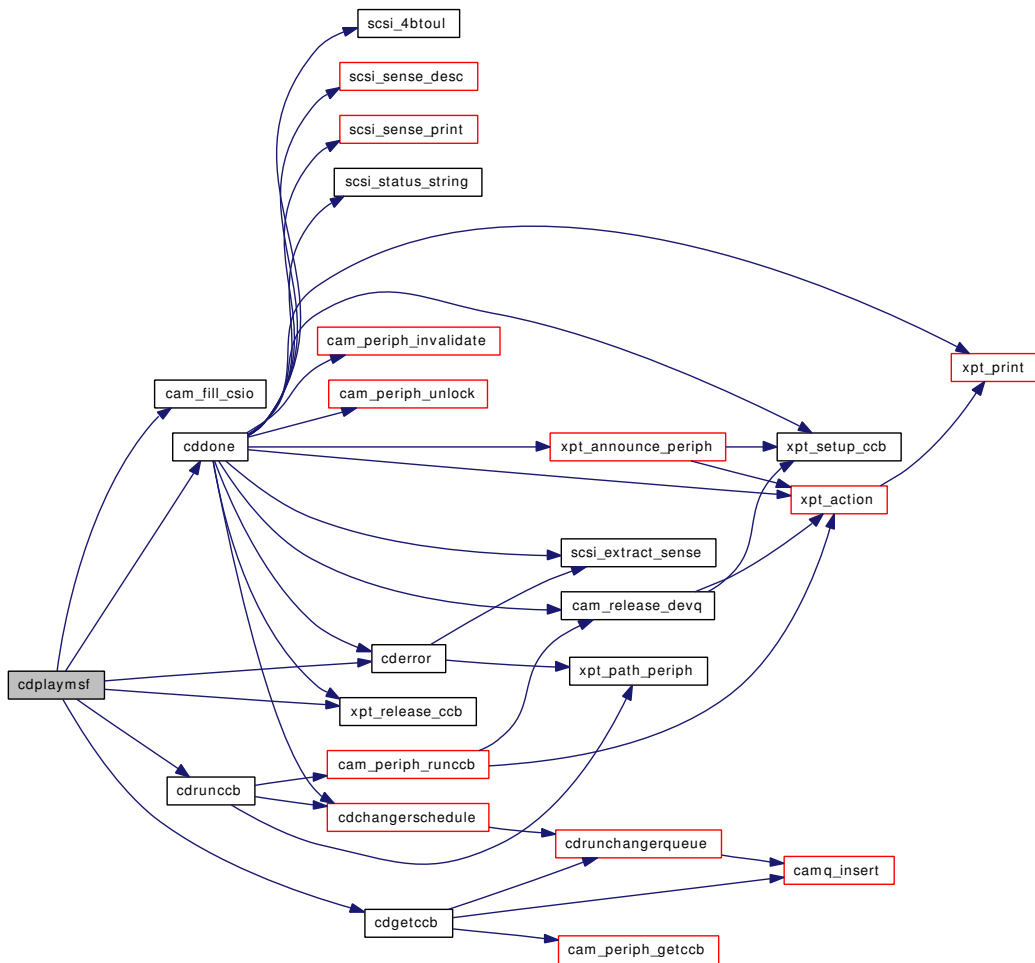
7.18.3.20 static int cdplaymsf (struct [cam_periph](#) * *periph*, u_int32_t *startm*, u_int32_t *starts*, u_int32_t *startf*, u_int32_t *endm*, u_int32_t *ends*, u_int32_t *endf*) [static]

Definition at line 3456 of file `scsi_cd.c`.

References `CAM_DIR_NONE`, `cam_fill_csio()`, `CAM_RETRY_SELTO`, `cdb_t::cdb_bytes`, `ccb_::scsiio::cdb_io`, `cddone()`, `cderror()`, `cdgetccb()`, `cdrunccb()`, `ccb::csio`, `MSG_SIMPLE_Q_TAG`, `PLAY_MSFSF`, `SF_RETRY_UA`, `SSD_FULL_SIZE`, and `xpt_release_ccb()`.

Referenced by `cdioctl()`.

Here is the call graph for this function:



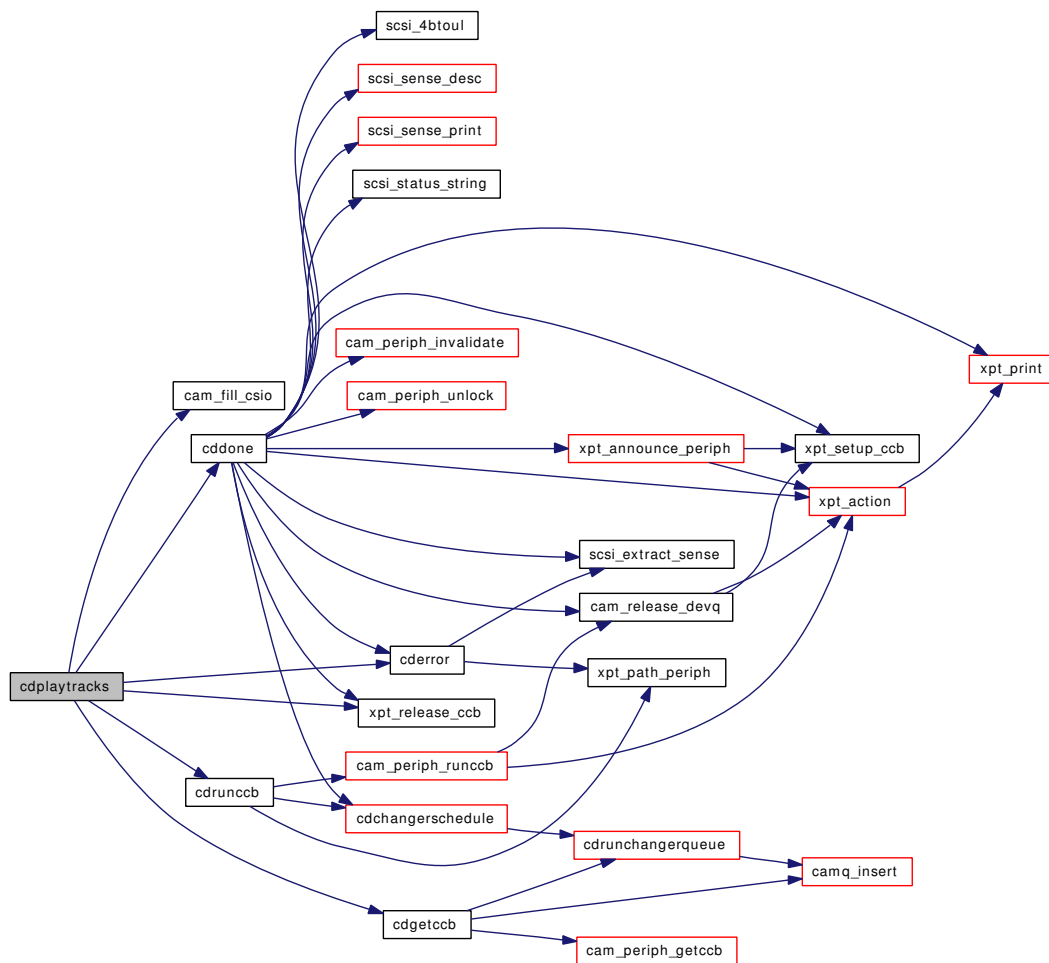
7.18.3.21 `static int cdplaytracks (struct cam_periph * periph, u_int32_t strack, u_int32_t sindex, u_int32_t etrack, u_int32_t eindex)` [static]

Definition at line 3502 of file `scsi_cd.c`.

References `CAM_DIR_NONE`, `cam_fill_csio()`, `CAM_RETRY_SELTO`, `cdb_t::cdb_bytes`, `ccb_::scsiio::cdb_io`, `cddone()`, `cderror()`, `cdgetccb()`, `cdrunccb()`, `ccb::csio`, `MSG_SIMPLE_Q_TAG`, `PLAY_TRACK`, `SF_RETRY_UA`, `SSD_FULL_SIZE`, and `xpt_release_ccb()`.

Referenced by `cdioctl()`.

Here is the call graph for this function:



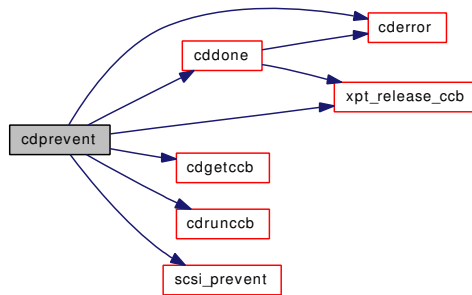
7.18.3.22 static void cdprevent (struct **cam_periph** * *periph*, int *action*) [static]

Definition at line 2679 of file `scsi_cd.c`.

References `CAM_DEBUG`, `CAM_DEBUG_TRACE`, `CAM_RETRY_SELTO`, `CD_FLAG_DISC_LOCKED`, `cddone()`, `cderror()`, `cdgetccb()`, `cdrunccb()`, `ccb::csio`, `cd_softc::flags`, `MSG_SIMPLE_Q_TAG`, `cam_periph::path`, `PR_ALLOW`, `PR_PREVENT`, `scsi_prevent()`, `SF_NO_PRINT`, `SF_RETRY_UA`, `cam_periph::softc`, `SSD_FULL_SIZE`, and `xpt_release_ccb()`.

Referenced by `cdcheckmedia()`, `cdclose()`, and `cdioctl()`.

Here is the call graph for this function:



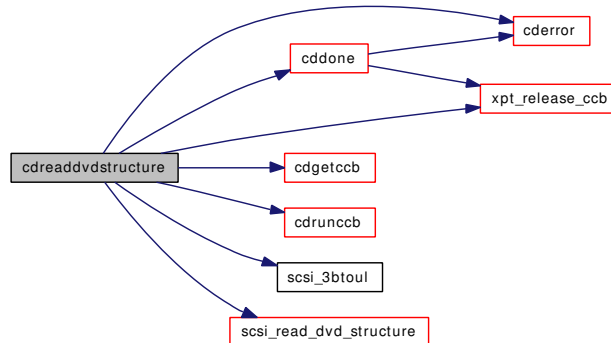
7.18.3.23 static int cdreadvdstructure (struct [cam_periph](#) * *periph*, struct [dvd_struct](#) * *dvdstruct*) [static]

Definition at line 3936 of file [scsi_cd.c](#).

References [CAM_RETRY_SELTO](#), [cddone\(\)](#), [cderror\(\)](#), [cdgetccb\(\)](#), [cdrunccb\(\)](#), [scsi_read_dvd_struct_data_copyright::cps_type](#), [ccb::csio](#), [scsi_read_dvd_struct_data_physical::layer_desc](#), [MSG_SIMPLE_Q_TAG](#), [RSD_BCA_MASK](#), [RSD_BCA_SHIFT](#), [RSD_BOOK_TYPE_MASK](#), [RSD_BOOK_TYPE_SHIFT](#), [RSD_BOOK_VERSION_MASK](#), [RSD_DISC_SIZE_MASK](#), [RSD_DISC_SIZE_SHIFT](#), [RSD_LAYER_TYPE_MASK](#), [RSD_LIN_DENSITY_MASK](#), [RSD_LIN_DENSITY_SHIFT](#), [RSD_MAX_RATE_MASK](#), [RSD_NUM_LAYERS_MASK](#), [RSD_NUM_LAYERS_SHIFT](#), [RSD_TRACK_DENSITY_MASK](#), [RSD_TRACK_PATH_MASK](#), [RSD_TRACK_PATH_SHIFT](#), [scsi_read_dvd_struct_data_copyright::region_info](#), [ccb_scsiio::resid](#), [scsi_3btoul\(\)](#), [scsi_read_dvd_structure\(\)](#), [SF_RETRY_UA](#), [SSD_FULL_SIZE](#), and [xpt_release_ccb\(\)](#).

Referenced by [cdioctl\(\)](#).

Here is the call graph for this function:



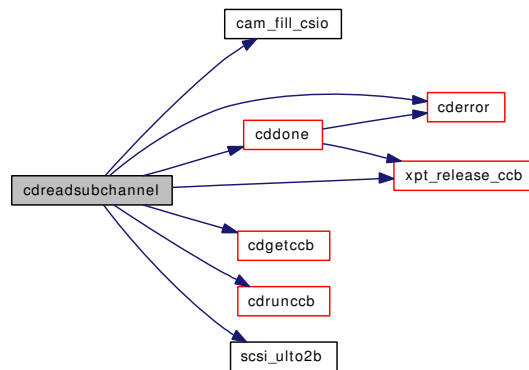
7.18.3.24 static int cdreadsubchannel (struct [cam_periph](#) * *periph*, [u_int32_t](#) *mode*, [u_int32_t](#) *format*, [int](#) *track*, struct [cd_sub_channel_info](#) * *data*, [u_int32_t](#) *len*) [static]

Definition at line 3155 of file [scsi_cd.c](#).

References [CAM_DIR_IN](#), [cam_fill_csio\(\)](#), [CAM_RETRY_SELTO](#), [CD_MSF](#), [cdb_t::cdb_bytes](#), [ccb_scsiio::cdb_io](#), [cddone\(\)](#), [cderror\(\)](#), [cdgetccb\(\)](#), [cdrunccb\(\)](#), [ccb::csio](#), [MSG_SIMPLE_Q_TAG](#), [READ_SUBCHANNEL](#), [scsi_ulto2b\(\)](#), [SF_RETRY_UA](#), [SRS_SUBQ](#), [SSD_FULL_SIZE](#), and [xpt_release_ccb\(\)](#).

Referenced by `cdioctl()`.

Here is the call graph for this function:



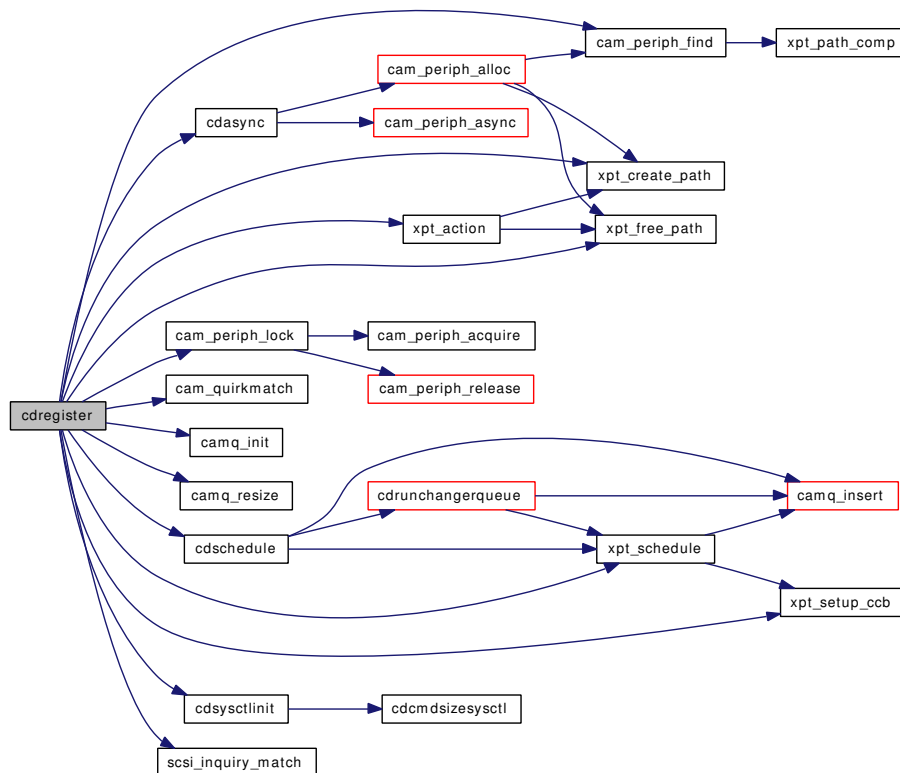
7.18.3.25 `static int cdreadtoc (struct cam_periph * periph, u_int32_t mode, u_int32_t start, u_int8_t * data, u_int32_t len, u_int32_t sense_flags)` `[static]`

Definition at line 3107 of file `scsi_cd.c`.

References `CAM_DIR_IN`, `cam_fill_csio()`, `CAM_RETRY_SELTO`, `CD_MSF`, `cdb_t::cdb_bytes`, `ccb_ - scsiio::cdb_io`, `cddone()`, `cderror()`, `cdgetccb()`, `cdrunccb()`, `ccb::csio`, `MSG_SIMPLE_Q_TAG`, `READ_ - TOC`, `SF_RETRY_UA`, `SSD_FULL_SIZE`, and `xpt_release_ccb()`.

Referenced by `cdcheckmedia()`, and `cdioctl()`.

Here is the call graph for this function:



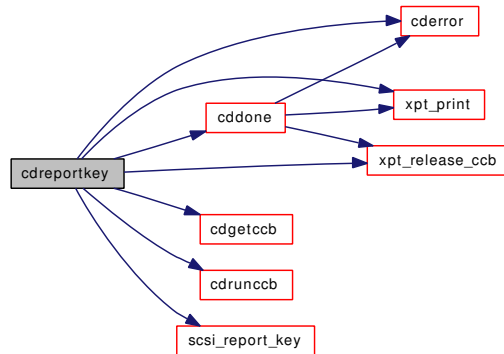
7.18.3.27 static int cdreportkey (struct [cam_periph](#) * *periph*, struct [dvd_authinfo](#) * *authinfo*) [static]

Definition at line 3684 of file `scsi_cd.c`.

References `scsi_report_key_data_agid::agid`, `scsi_report_key_data_title::byte0`, `scsi_report_key_data_rpc::byte4`, `CAM_RETRY_SELTO`, `cddone()`, `cderror()`, `cdgetccb()`, `cdrunccb()`, `scsi_report_key_data_challenge::challenge_key`, `ccb::csio`, `scsi_report_key_data_key1_key2::key1`, `MSG_SIMPLE_Q_TAG`, `cam_periph::path`, `scsi_report_key_data_rpc::region_mask`, `ccb_scsiio::resid`, `RKD_AGID_MASK`, `RKD_AGID_SHIFT`, `RKD_ASF_SUCCESS`, `RKD_RPC_TYPE_MASK`, `RKD_RPC_TYPE_SHIFT`, `RKD_RPC_USER_RESET_MASK`, `RKD_RPC_VENDOR_RESET_MASK`, `RKD_RPC_VENDOR_RESET_SHIFT`, `RKD_TITLE_CMGS_MASK`, `RKD_TITLE_CMGS_SHIFT`, `RKD_TITLE_CP_SEC`, `RKD_TITLE_CP_SEC_SHIFT`, `RKD_TITLE_CPM`, `RKD_TITLE_CPM_SHIFT`, `scsi_report_key_data_rpc::rpc_scheme1`, `scsi_report_key()`, `SF_RETRY_UA`, `SSD_FULL_SIZE`, `scsi_report_key_data_asf::success`, `scsi_report_key_data_title::title_key`, `xpt_print()`, and `xpt_release_ccb()`.

Referenced by `cdioctl()`.

Here is the call graph for this function:



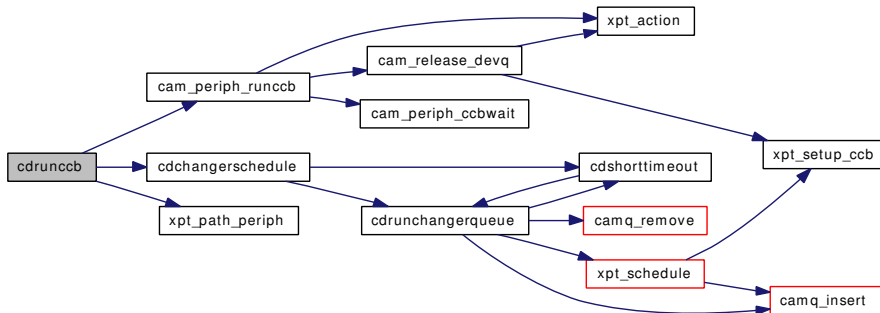
7.18.3.28 `static int cdrunccb (union ccb * ccb, int(*) (union ccb * ccb, u_int32_t cam_flags, u_int32_t sense_flags) error_routine, u_int32_t cam_flags, u_int32_t sense_flags)` [static]

Definition at line 1344 of file `scsi_cd.c`.

References `cam_periph_runccb()`, `CD_FLAG_CHANGER`, `cdchangerschedule()`, `cd_softc::flags`, `cam_periph::softc`, and `xpt_path_periph()`.

Referenced by `cdgetmode()`, `cdpause()`, `cdplay()`, `cdplaymsf()`, `cdplaytracks()`, `cdprevent()`, `cdreadvd-structure()`, `cdreadsubchannel()`, `cdreadtoc()`, `cdreportkey()`, `cdsendkey()`, `cdsetmode()`, `cdsetspeed()`, `cd-size()`, `cdstartunit()`, and `cdstopunit()`.

Here is the call graph for this function:



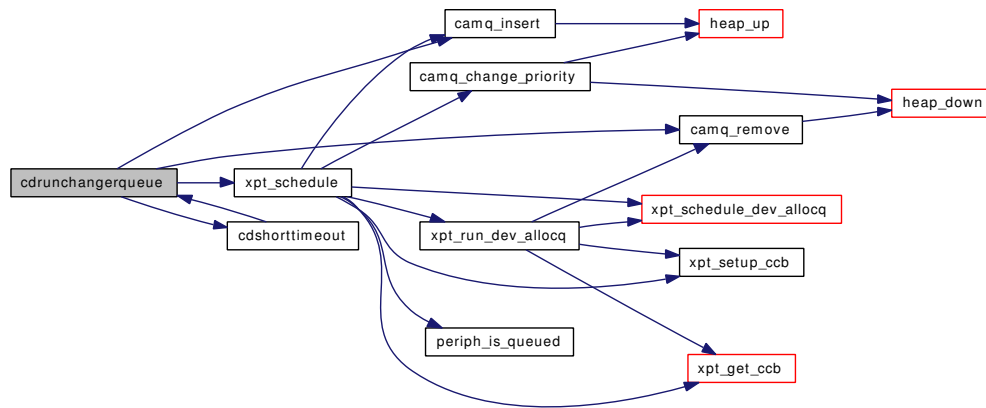
7.18.3.29 `static void cdrunchangerqueue (void * arg)` [static]

Definition at line 1156 of file `scsi_cd.c`.

References `cd_softc::bio_queue`, `CAMQ_HEAD`, `camq_insert()`, `camq_remove()`, `CD_FLAG_ACTIVE`, `CD_FLAG_SCHED_ON_COMP`, `cdshorttimeout()`, `CHANGER_MANUAL_CALL`, `CHANGER_NEED_TIMEOUT`, `CHANGER_SHORT_TMOUT_SCHED`, `CHANGER_TIMEOUT_SCHEDULED`, `cdchanger::cur_device`, `cdchanger::devq`, `camq::entries`, `cd_softc::flags`, `cdchanger::flags`, `camq::generation`, `cam_pinfo::generation`, `cdchanger::long_handle`, `cd_softc::pinfo`, `camq::qfrozen_cnt`, `cdchanger::short_handle`, and `xpt_schedule()`.

Referenced by `cdchangerschedule()`, `cdcleanup()`, `cdgetccb()`, `cdschedule()`, and `cdshorttimeout()`.

Here is the call graph for this function:



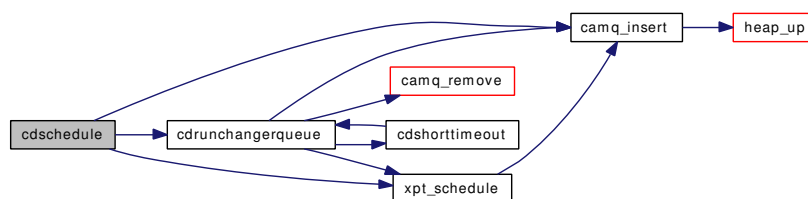
7.18.3.30 static void cdschedule (struct [cam_periph](#) * *periph*, int *priority*) [static]

Definition at line 1106 of file `scsi_cd.c`.

References `CAM_UNQUEUED_INDEX`, `camq_insert()`, `CD_FLAG_ACTIVE`, `CD_FLAG_SCHED_ON_COMP`, `cdchangerschedule()`, `CHANGER_MANUAL_CALL`, `CHANGER_NEED_TIMEOUT`, `CHANGER_SHORT_TMOUT_SCHED`, `CHANGER_TIMEOUT_SCHED`, `cd_softc::flags`, `cam_pinfo::generation`, `cam_pinfo::index`, `cd_softc::pinfo`, `cam_pinfo::priority`, `cam_periph::softc`, and `xpt_schedule()`.

Referenced by `cdregister()`, and `cdstrategy()`.

Here is the call graph for this function:



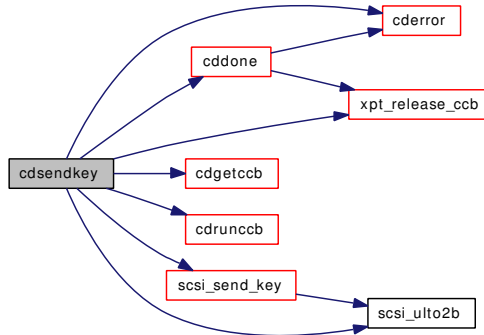
7.18.3.31 static int cdsendkey (struct [cam_periph](#) * *periph*, struct [dvd_authinfo](#) * *authinfo*) [static]

Definition at line 3843 of file `scsi_cd.c`.

References `CAM_RETRY_SELTO`, `cddone()`, `cderror()`, `cdgetccb()`, `cdrunccb()`, `scsi_report_key_data_challenge::challenge_key`, `ccb::csio`, `scsi_send_key_data_rpc::data_len`, `scsi_report_key_data_key1_key2::data_len`, `scsi_report_key_data_challenge::data_len`, `scsi_report_key_data_key1_key2::key1`, `MSG_SIMPLE_Q_TAG`, `scsi_send_key_data_rpc::region_code`, `scsi_send_key()`, `scsi_ulto2b()`, `SF_RETRY_UA`, `SSD_FULL_SIZE`, and `xpt_release_ccb()`.

Referenced by `cdioctl()`.

Here is the call graph for this function:



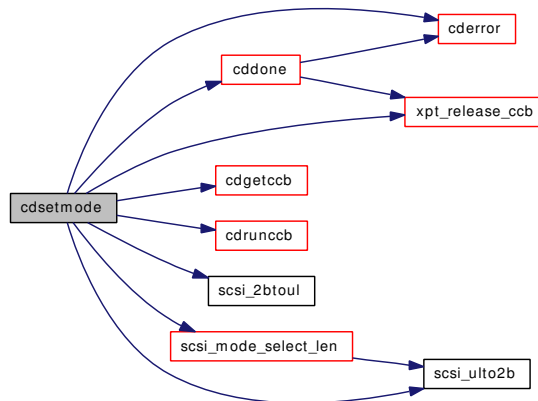
7.18.3.32 `static int cdsetmode (struct cam_periph * periph, struct cd_mode_params * data)` `[static]`

Definition at line 3307 of file `scsi_cd.c`.

References `cd_mode_params::alloc_len`, `CAM_RETRY_SELTO`, `cd_mode_params::cdb_size`, `cddone()`, `cderror()`, `cdgetccb()`, `cdrunccb()`, `ccb::csio`, `scsi_mode_header_6::data_length`, `scsi_mode_header_10::data_length`, `scsi_mode_header_6::medium_type`, `scsi_mode_header_10::medium_type`, `cd_mode_params::mode_buf`, `MSG_SIMPLE_Q_TAG`, `scsi_2btoul()`, `scsi_mode_select_len()`, `scsi_ulito2b()`, `SF_RETRY_UA`, `cam_periph::softc`, `SSD_FULL_SIZE`, and `xpt_release_ccb()`.

Referenced by `cdioctl()`.

Here is the call graph for this function:



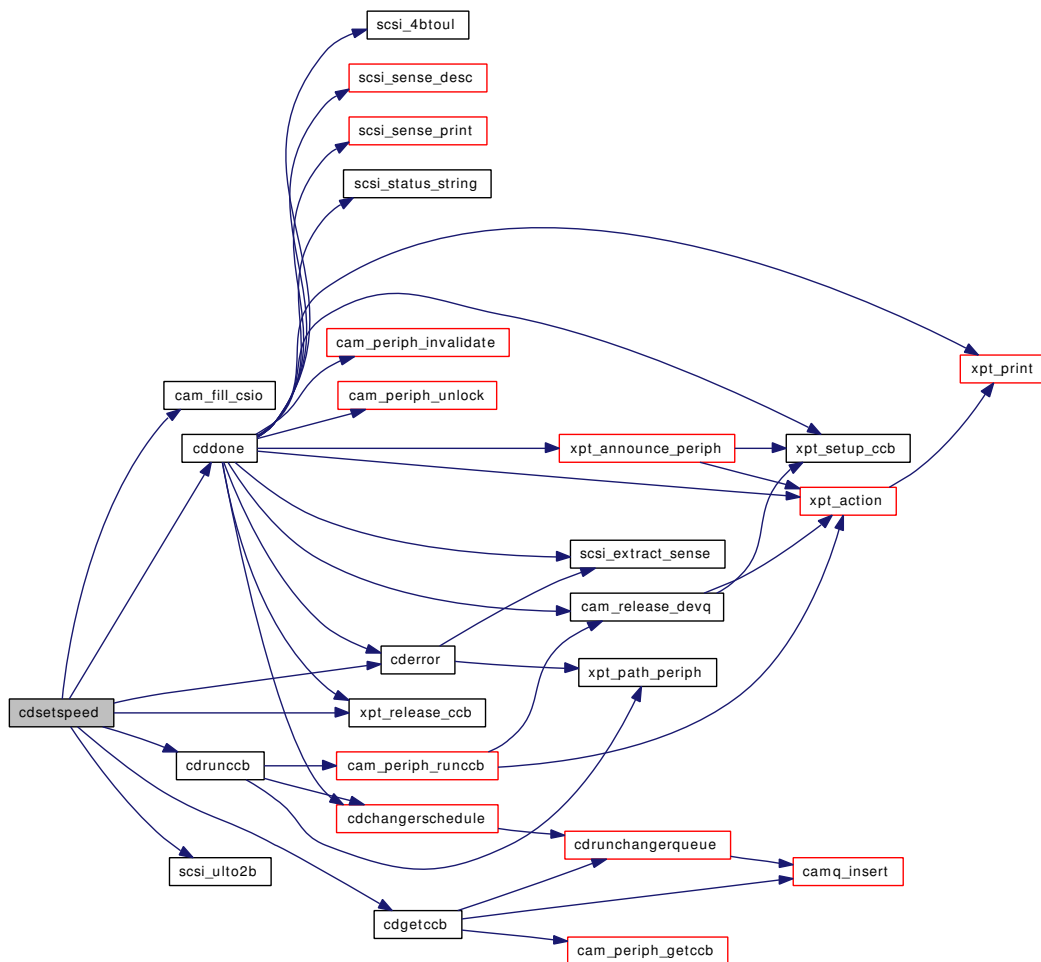
7.18.3.33 `static int cdsetspeed (struct cam_periph * periph, u_int32_t rdspeed, u_int32_t wrspeed)` `[static]`

Definition at line 3640 of file `scsi_cd.c`.

References CAM_DIR_NONE, cam_fill_csio(), CAM_RETRY_SELTO, cdb_t::cdb_bytes, ccb_scsiio::cdb_io, cddone(), cderror(), cdgetccb(), cdrunccb(), ccb::csio, MSG_SIMPLE_Q_TAG, scsi_ulto2b(), SET_CD_SPEED, SF_RETRY_UA, SSD_FULL_SIZE, and xpt_release_ccb().

Referenced by cdiotl().

Here is the call graph for this function:



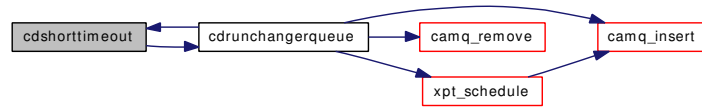
7.18.3.34 static void cdshorttimeout (void *arg) [static]

Definition at line 1077 of file scsi_cd.c.

References cd_softc::bio_queue, cdrunchangerqueue(), CHANGER_MANUAL_CALL, CHANGER_SHORT_TMOUOT_SCHED, cdchanger::cur_device, and cdchanger::flags.

Referenced by cdchangerschedule(), cdcleanup(), and cdrunchangerqueue().

Here is the call graph for this function:



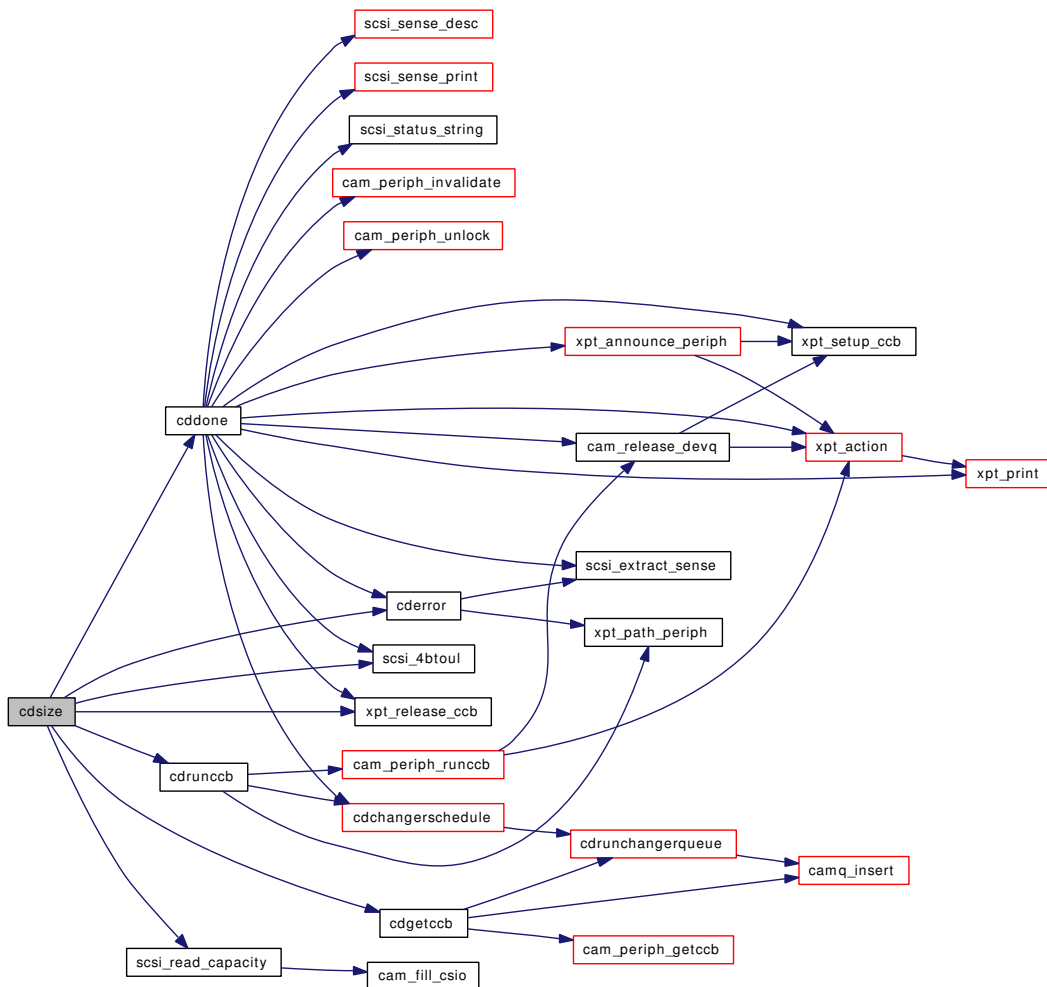
7.18.3.35 static int cdsize (struct cam_periph * periph, u_int32_t * size) [static]

Definition at line 2859 of file scsi_cd.c.

References scsi_read_capacity_data::addr, CAM_DEBUG, CAM_DEBUG_TRACE, CAM_RETRY_SELTO, cddone(), cderror(), cdgetccb(), cdruncb(), ccb::csio, scsi_read_capacity_data::length, MSG_SIMPLE_Q_TAG, cam_periph::path, scsi_4btoul(), scsi_read_capacity(), SF_NO_PRINT, SF_RETRY_UA, cam_periph::softc, SSD_FULL_SIZE, and xpt_release_ccb().

Referenced by cdcheckmedia().

Here is the call graph for this function:

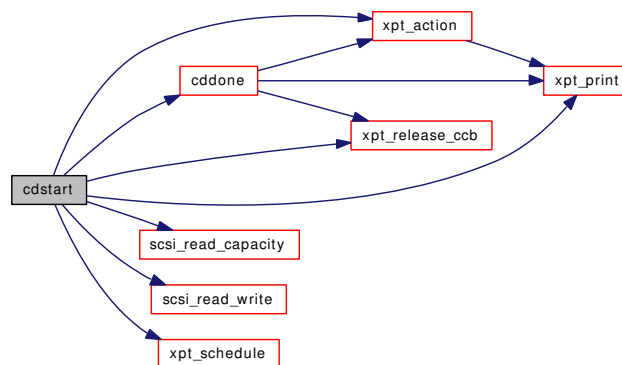


7.18.3.36 `static void cdstart (struct cam_periph * periph, union ccb * start_ccb)` [static]

Definition at line 1480 of file `scsi_cd.c`.

References `cd_softc::bio_queue`, `CAM_DEBUG`, `CAM_DEBUG_TRACE`, `CAM_PRIORITY_NONE`, `ccb::ccb_h`, `CD_CCB_BUFFER_IO`, `CD_CCB_PROBE`, `CD_CCB_RETRY_UA`, `CD_CCB_WAITING`, `CD_FLAG_RETRY_UA`, `CD_STATE_NORMAL`, `CD_STATE_PROBE`, `cddone()`, `ccb::csio`, `cd_softc::flags`, `cam_periph::immediate_priority`, `MSG_SIMPLE_Q_TAG`, `cam_periph::path`, `cam_periph::pinfo`, `cam_pinfo::priority`, `scsi_read_capacity()`, `scsi_read_write()`, `cam_periph::softc`, `SSD_FULL_SIZE`, `cd_softc::state`, `xpt_action()`, `xpt_print()`, `xpt_release_ccb()`, and `xpt_schedule()`.

Here is the call graph for this function:

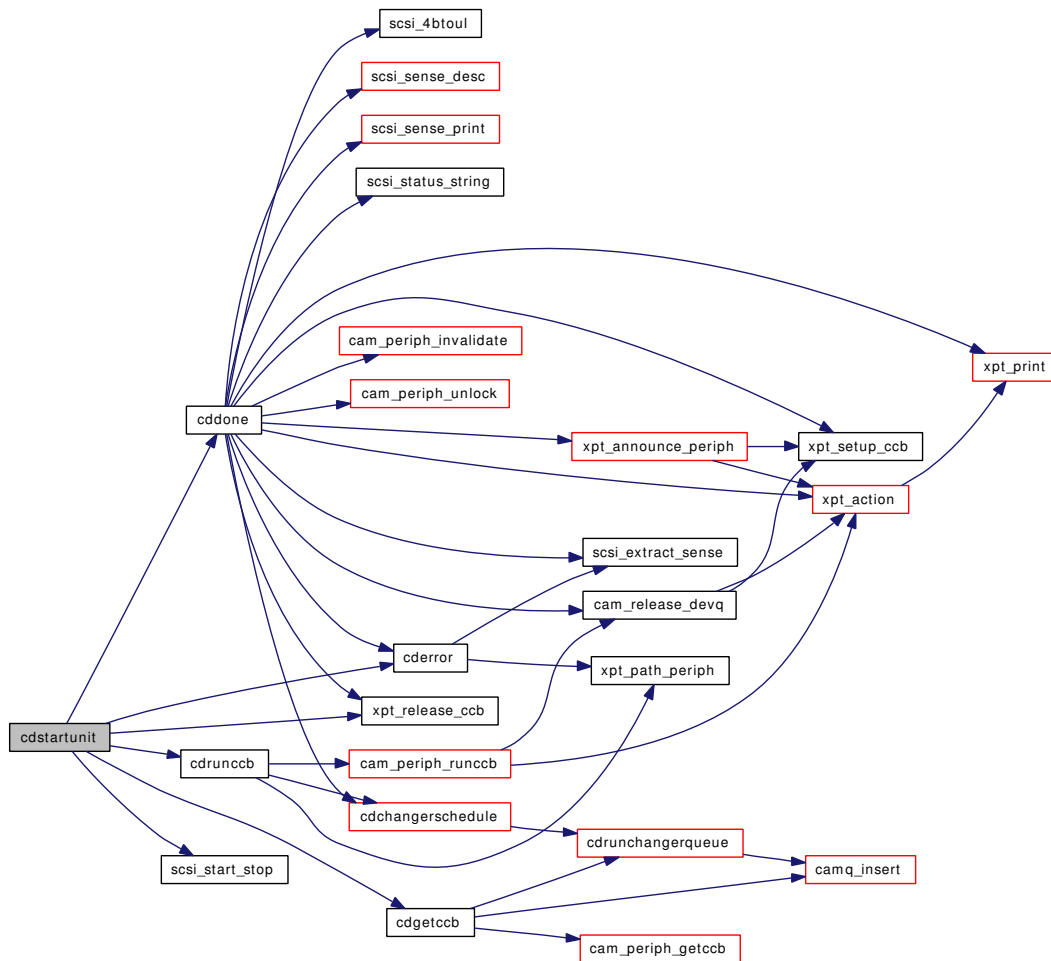
**7.18.3.37** `static int cdstartunit (struct cam_periph * periph, int load)` [static]

Definition at line 3584 of file `scsi_cd.c`.

References `CAM_RETRY_SELTO`, `cddone()`, `cderror()`, `cdgetccb()`, `cdrunccb()`, `ccb::csio`, `MSG_SIMPLE_Q_TAG`, `scsi_start_stop()`, `SF_RETRY_UA`, `SSD_FULL_SIZE`, and `xpt_release_ccb()`.

Referenced by `cdioctl()`.

Here is the call graph for this function:



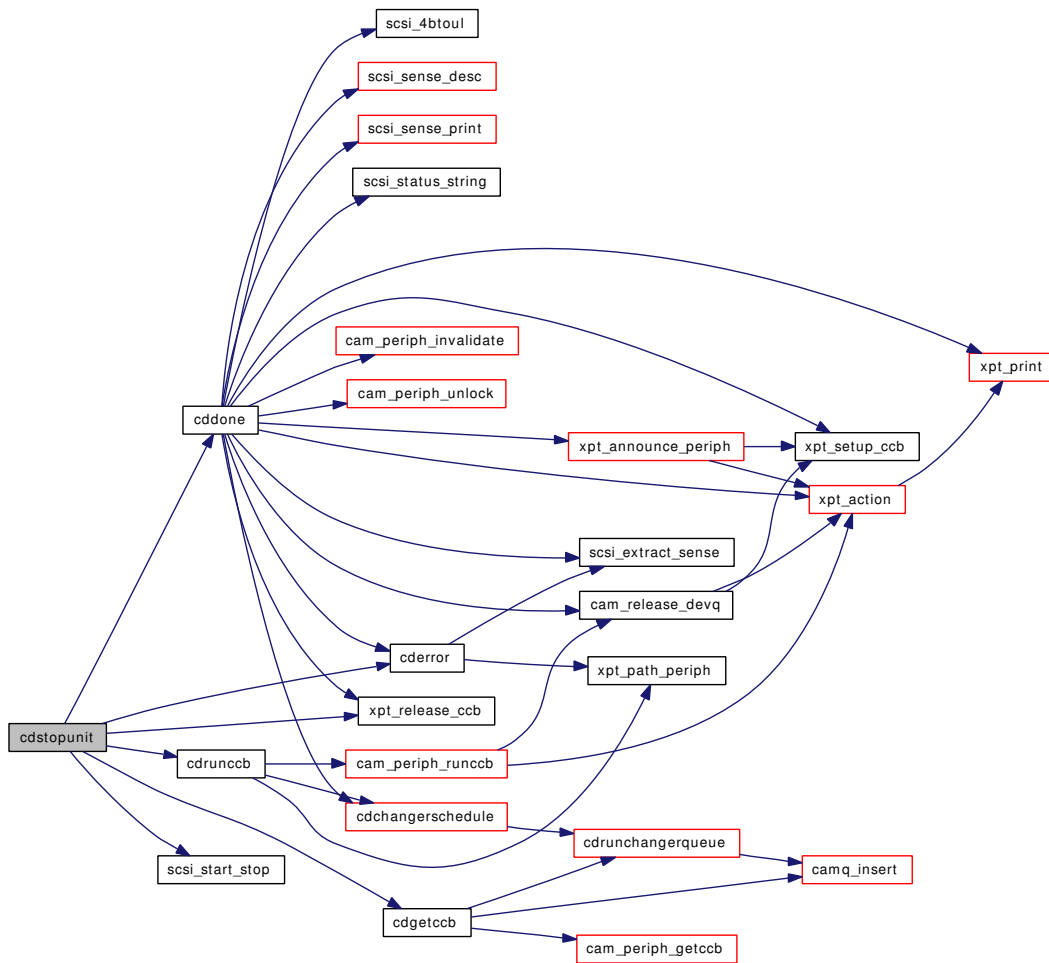
7.18.3.38 static int cdstopunit (struct **cam_periph** * *periph*, u_int32_t *eject*) [static]

Definition at line 3612 of file scsi_cd.c.

References CAM_RETRY_SELTO, cddone(), cderror(), cdgetccb(), cdrunccb(), ccb::csio, MSG_SIMPLE_Q_TAG, scsi_start_stop(), SF_RETRY_UA, SSD_FULL_SIZE, and xpt_release_ccb().

Referenced by cdioctl().

Here is the call graph for this function:

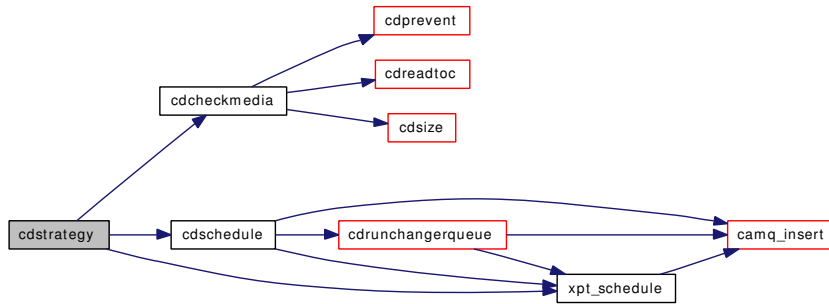


7.18.3.39 static void cdstrategy (struct bio * bp) [static]

Definition at line 1413 of file scsi_cd.c.

References `cd_softc::bio_queue`, `CAM_DEBUG`, `CAM_DEBUG_TRACE`, `CD_FLAG_CHANGER`, `CD_FLAG_INVALID`, `CD_FLAG_VALID_MEDIA`, `cdcheckmedia()`, `cdschedule()`, `cd_softc::flags`, `cam_periph::path`, `cam_periph::softc`, and `xpt_schedule()`.

Here is the call graph for this function:



7.18.3.40 static void cdsysctlinit (void * context, int pending) [static]

Definition at line 568 of file scsi_cd.c.

References CD_FLAG_SCTX_INIT, cdcmdsizesysctl(), cd_softc::flags, cam_periph::softc, and cam_periph::unit_number.

Referenced by cdregister().

Here is the call graph for this function:



7.18.3.41 PERIPHDRIVER_DECLARE (cd, cd driver)

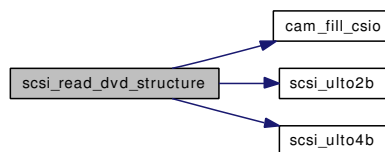
7.18.3.42 void scsi_read_dvd_structure (struct ccb_scsiio * csio, u_int32_t retries, void(*) (struct cam_periph *, union ccb *) cbfcn, u_int8_t tag_action, u_int32_t address, u_int8_t layer_number, u_int8_t format, u_int8_t agid, u_int8_t * data_ptr, u_int32_t dxfer_len, u_int8_t sense_len, u_int32_t timeout)

Definition at line 4198 of file scsi_cd.c.

References CAM_DIR_IN, cam_fill_csio(), cdb_t::cdb_bytes, ccb_scsiio::cdb_io, READ_DVD_STRUCTURE, scsi_ulto2b(), and scsi_ulto4b().

Referenced by cdreadvdstructure().

Here is the call graph for this function:



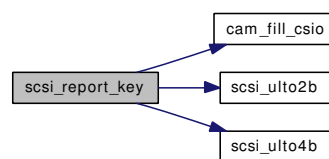
7.18.3.43 void **scsi_report_key** (struct **ccb_scsiio** * *csio*, u_int32_t *retries*, void(*) (struct **cam_periph** *, union **ccb** *) *cbfcn*, u_int8_t *tag_action*, u_int32_t *lba*, u_int8_t *agid*, u_int8_t *key_format*, u_int8_t * *data_ptr*, u_int32_t *dxfer_len*, u_int8_t *sense_len*, u_int32_t *timeout*)

Definition at line 4139 of file `scsi_cd.c`.

References `CAM_DIR_IN`, `CAM_DIR_NONE`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `REPORT_KEY`, `RK_KF_AGID_SHIFT`, `RK_KF_KEYFORMAT_MASK`, `scsi_ulito2b()`, and `scsi_ulito4b()`.

Referenced by `cdreportkey()`.

Here is the call graph for this function:



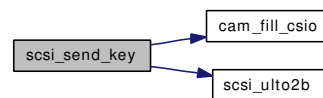
7.18.3.44 void **scsi_send_key** (struct **ccb_scsiio** * *csio*, u_int32_t *retries*, void(*) (struct **cam_periph** *, union **ccb** *) *cbfcn*, u_int8_t *tag_action*, u_int8_t *agid*, u_int8_t *key_format*, u_int8_t * *data_ptr*, u_int32_t *dxfer_len*, u_int8_t *sense_len*, u_int32_t *timeout*)

Definition at line 4168 of file `scsi_cd.c`.

References `CAM_DIR_OUT`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `RK_KF_AGID_SHIFT`, `RK_KF_KEYFORMAT_MASK`, `scsi_ulito2b()`, and `SEND_KEY`.

Referenced by `cdsendkey()`.

Here is the call graph for this function:

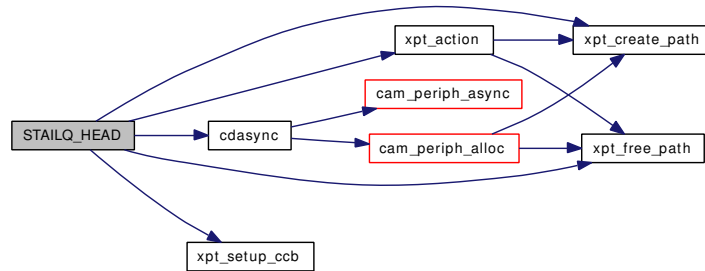


7.18.3.45 static **STAILQ_HEAD** (**changerlist**, **cdchanger**) [static]

Definition at line 329 of file `scsi_cd.c`.

References `AC_FOUND_DEVICE`, `ccb_setasync::callback`, `ccb_setasync::callback_arg`, `CAM_LUN_WILDCARD`, `CAM_REQ_CMP`, `CAM_TARGET_WILDCARD`, `CAM_XPT_PATH_ID`, `ccb::ccb_h`, `ccb_setasync::ccb_h`, `cdasync()`, `ccb_setasync::event_enable`, `ccb_hdr::func_code`, `ccb_hdr::status`, `xpt_action()`, `xpt_create_path()`, `xpt_free_path()`, `XPT_SASYNC_CB`, and `xpt_setup_ccb()`.

Here is the call graph for this function:



7.18.3.46 `SYSCTL_INT` (`_kern_cam_cd_changer`, `OID_AUTO`, `max_busy_seconds`, `CTLFLAG_RW`, & `changer_max_busy_seconds`, 0, "Maximum changer scheduling quantum")

7.18.3.47 `SYSCTL_INT` (`_kern_cam_cd_changer`, `OID_AUTO`, `min_busy_seconds`, `CTLFLAG_RW`, & `changer_min_busy_seconds`, 0, "Minimum changer scheduling quantum")

7.18.3.48 `SYSCTL_NODE` (`_kern_cam_cd`, `OID_AUTO`, `changer`, `CTLFLAG_RD`, 0, "CD Changer")

7.18.3.49 `SYSCTL_NODE` (`_kern_cam`, `OID_AUTO`, `cd`, `CTLFLAG_RD`, 0, "CAM CDROM driver")

7.18.3.50 `TUNABLE_INT` ("kern.cam.cd.changer.max_busy_seconds", & `changer_max_busy_seconds`)

7.18.3.51 `TUNABLE_INT` ("kern.cam.cd.changer.min_busy_seconds", & `changer_min_busy_seconds`)

7.18.4 Variable Documentation

7.18.4.1 `struct cd_page_sizes cd_page_size_table[]` [static]

Initial value:

```
{
    { AUDIO_PAGE, sizeof(struct cd_audio_page) }
}
```

Definition at line 170 of file `scsi_cd.c`.

Referenced by `cdgetpagesize()`.

7.18.4.2 `struct cd_quirk_entry cd_quirk_table[]` [static]

Initial value:

```
{
    {
```

```

        { T_CDROM, SIP_MEDIA_REMOVABLE, "NRC", "MBR-7", "*"},
          CD_Q_CHANGER
    },
    {
        { T_CDROM, SIP_MEDIA_REMOVABLE, "PIONEER", "CD-ROM DRM*",
          "*"}, CD_Q_CHANGER
    },
    {
        { T_CDROM, SIP_MEDIA_REMOVABLE, "NAKAMICH", "MJ-*", "*"},
          CD_Q_CHANGER
    },
    {
        { T_CDROM, SIP_MEDIA_REMOVABLE, "CHINON", "CD-ROM CDS-535", "*"},
          CD_Q_BCD_TRACKS
    }
}

```

Definition at line 198 of file scsi_cd.c.

Referenced by cdregister().

7.18.4.3 `periph_dtor_t cdcleanup` [static]

Definition at line 225 of file scsi_cd.c.

7.18.4.4 `disk_close_t cdclose` [static]

Definition at line 219 of file scsi_cd.c.

7.18.4.5 `struct periph_driver cddriver` [static]

Initial value:

```

{
    cdinit, "cd",
    TAILQ_HEAD_INITIALIZER(cddriver.units), 0
}

```

Definition at line 285 of file scsi_cd.c.

7.18.4.6 `periph_init_t cdinit` [static]

Definition at line 223 of file scsi_cd.c.

7.18.4.7 `disk_ioctl_t cdiioctl` [static]

Definition at line 220 of file scsi_cd.c.

7.18.4.8 `periph_oninv_t cdoninvalidate` [static]

Definition at line 227 of file scsi_cd.c.

7.18.4.9 `disk_open_t cdopen` [static]

Definition at line 218 of file `scsi_cd.c`.

7.18.4.10 `periph_ctor_t cdregister` [static]

Definition at line 224 of file `scsi_cd.c`.

7.18.4.11 `periph_start_t cdstart` [static]

Definition at line 226 of file `scsi_cd.c`.

7.18.4.12 `disk_strategy_t cdstrategy` [static]

Definition at line 221 of file `scsi_cd.c`.

7.18.4.13 `int changer_max_busy_seconds = CHANGER_MAX_BUSY_SECONDS` [static]

Definition at line 304 of file `scsi_cd.c`.

7.18.4.14 `int changer_min_busy_seconds = CHANGER_MIN_BUSY_SECONDS` [static]

Definition at line 303 of file `scsi_cd.c`.

7.18.4.15 `int num_changers` [static]

Definition at line 294 of file `scsi_cd.c`.

7.19 /usr/src/sys/cam/scsi/scsi_cd.h File Reference

This graph shows which files directly or indirectly include this file:



Data Structures

- struct [scsi_pause](#)
- struct [scsi_play_msf](#)
- struct [scsi_play_track](#)
- struct [scsi_play_10](#)
- struct [scsi_play_12](#)
- struct [scsi_play_rel_12](#)
- struct [scsi_read_header](#)
- struct [scsi_read_subchannel](#)
- struct [scsi_read_toc](#)
- struct [scsi_read_cd_capacity](#)
- struct [scsi_set_speed](#)
- struct [scsi_report_key](#)
- struct [scsi_send_key](#)
- struct [scsi_read_dvd_structure](#)
- struct [scsi_report_key_data_header](#)
- struct [scsi_report_key_data_agid](#)
- struct [scsi_report_key_data_challenge](#)
- struct [scsi_report_key_data_key1_key2](#)
- struct [scsi_report_key_data_title](#)
- struct [scsi_report_key_data_asf](#)
- struct [scsi_report_key_data_rpc](#)
- struct [scsi_send_key_data_rpc](#)
- struct [scsi_read_dvd_struct_data_header](#)
- struct [scsi_read_dvd_struct_data_layer_desc](#)
- struct [scsi_read_dvd_struct_data_physical](#)
- struct [scsi_read_dvd_struct_data_copyright](#)
- struct [scsi_read_dvd_struct_data_disc_key](#)
- struct [scsi_read_dvd_struct_data_bca](#)
- struct [scsi_read_dvd_struct_data_manufacturer](#)
- struct [scsi_read_dvd_struct_data_copy_manage](#)
- struct [scsi_read_dvd_struct_data_prot_discid](#)
- struct [scsi_read_dvd_struct_data_disc_key_blk](#)
- struct [scsi_read_dvd_struct_data_dds](#)
- struct [scsi_read_dvd_struct_data_medium_status](#)
- struct [scsi_read_dvd_struct_data_spare_area](#)
- struct [scsi_read_dvd_struct_data_rmd_borderout](#)
- struct [scsi_read_dvd_struct_data_rmd](#)
- struct [scsi_read_dvd_struct_data_lead_in](#)
- struct [scsi_read_dvd_struct_data_disc_id](#)
- struct [scsi_read_dvd_struct_data_generic_dcb](#)

- struct `scsi_read_dvd_struct_data_dcb`
- struct `read_dvd_struct_write_prot`
- struct `read_dvd_struct_list_entry`
- struct `read_dvd_struct_data_list`
- struct `scsi_read_cd_cap_data`
- struct `cd_audio_page`
- struct `cd_audio_page::port_control`
- union `cd_pages`
- struct `cd_mode_data_10`
- struct `cd_mode_data`
- union `cd_mode_data_6_10`
- struct `cd_mode_params`

Defines

- #define `_SCSI_SCSI_CD_H` 1
- #define `CD_RELADDR` 0x01
- #define `CD_MSF` 0x02
- #define `PA_PAUSE` 1
- #define `PA_RESUME` 0
- #define `SRS_SUBQ` 0x40
- #define `RK_KF_AGID_MASK` 0xc0
- #define `RK_KF_AGID_SHIFT` 6
- #define `RK_KF_KEYFORMAT_MASK` 0x3f
- #define `RK_KF_AGID` 0x00
- #define `RK_KF_CHALLENGE` 0x01
- #define `RF_KF_KEY1` 0x02
- #define `RK_KF_KEY2` 0x03
- #define `RF_KF_TITLE` 0x04
- #define `RF_KF_ASF` 0x05
- #define `RK_KF_RPC_SET` 0x06
- #define `RF_KF_RPC_REPORT` 0x08
- #define `RF_KF_INV_AGID` 0x3f
- #define `RDS_FORMAT_PHYSICAL` 0x00
- #define `RDS_FORMAT_COPYRIGHT` 0x01
- #define `RDS_FORMAT_DISC_KEY` 0x02
- #define `RDS_FORMAT_BCA` 0x03
- #define `RDS_FORMAT_MANUFACTURER` 0x04
- #define `RDS_FORMAT_CMGS_CPM` 0x05
- #define `RDS_FORMAT_PROT_DISCID` 0x06
- #define `RDS_FORMAT_DISC_KEY_BLOCK` 0x07
- #define `RDS_FORMAT_DDS` 0x08
- #define `RDS_FORMAT_DVDRAM_MEDIA_STAT` 0x09
- #define `RDS_FORMAT_SPARE_AREA` 0x0a
- #define `RDS_FORMAT_RMD_BORDEROUT` 0x0c
- #define `RDS_FORMAT_RMD` 0x0d
- #define `RDS_FORMAT_LEADIN` 0x0e
- #define `RDS_FORMAT_DISC_ID` 0x0f
- #define `RDS_FORMAT_DCB` 0x30
- #define `RDS_FORMAT_WRITE_PROT` 0xc0

- #define [RDS_FORMAT_STRUCTURE_LIST](#) 0xff
- #define [READ_CD_CAPACITY](#) 0x25
- #define [READ_SUBCHANNEL](#) 0x42
- #define [READ_TOC](#) 0x43
- #define [READ_HEADER](#) 0x44
- #define [PLAY_10](#) 0x45
- #define [PLAY_MSF](#) 0x47
- #define [PLAY_TRACK](#) 0x48
- #define [PLAY_TRACK_REL](#) 0x49
- #define [PAUSE](#) 0x4b
- #define [SEND_KEY](#) 0xa3
- #define [REPORT_KEY](#) 0xa4
- #define [PLAY_12](#) 0xa5
- #define [PLAY_TRACK_REL_BIG](#) 0xa9
- #define [READ_DVD_STRUCTURE](#) 0xad
- #define [SET_CD_SPEED](#) 0xbb
- #define [RKD_AGID_MASK](#) 0xc0
- #define [RKD_AGID_SHIFT](#) 6
- #define [RKD_TITLE_CPM](#) 0x80
- #define [RKD_TITLE_CPM_SHIFT](#) 7
- #define [RKD_TITLE_CP_SEC](#) 0x40
- #define [RKD_TITLE_CP_SEC_SHIFT](#) 6
- #define [RKD_TITLE_CMGS_MASK](#) 0x30
- #define [RKD_TITLE_CMGS_SHIFT](#) 4
- #define [RKD_TITLE_CMGS_NO_RST](#) 0x00
- #define [RKD_TITLE_CMGS_RSVD](#) 0x10
- #define [RKD_TITLE_CMGS_1_GEN](#) 0x20
- #define [RKD_TITLE_CMGS_NO_COPY](#) 0x30
- #define [RKD_ASF_SUCCESS](#) 0x01
- #define [RKD_RPC_SCHEME_UNKNOWN](#) 0x00
- #define [RKD_RPC_SCHEME_PHASE_II](#) 0x01
- #define [RKD_RPC_TYPE_MASK](#) 0xC0
- #define [RKD_RPC_TYPE_SHIFT](#) 6
- #define [RKD_RPC_TYPE_NONE](#) 0x00
- #define [RKD_RPC_TYPE_SET](#) 0x40
- #define [RKD_RPC_TYPE_LAST_CHANCE](#) 0x80
- #define [RKD_RPC_TYPE_PERM](#) 0xC0
- #define [RKD_RPC_VENDOR_RESET_MASK](#) 0x38
- #define [RKD_RPC_VENDOR_RESET_SHIFT](#) 3
- #define [RKD_RPC_USER_RESET_MASK](#) 0x07
- #define [RKD_RPC_USER_RESET_SHIFT](#) 0
- #define [RSDS_BOOK_TYPE_DVD_ROM](#) 0x00
- #define [RSDS_BOOK_TYPE_DVD_RAM](#) 0x10
- #define [RSDS_BOOK_TYPE_DVD_R](#) 0x20
- #define [RSDS_BOOK_TYPE_DVD_RW](#) 0x30
- #define [RSDS_BOOK_TYPE_DVD_PRW](#) 0x90
- #define [RSDS_BOOK_TYPE_MASK](#) 0xf0
- #define [RSDS_BOOK_TYPE_SHIFT](#) 4
- #define [RSDS_BOOK_VERSION_MASK](#) 0x0f
- #define [RSDS_DISC_SIZE_120MM](#) 0x00

- #define `RSDS_DISC_SIZE_80MM` 0x10
- #define `RSDS_DISC_SIZE_MASK` 0xf0
- #define `RSDS_DISC_SIZE_SHIFT` 4
- #define `RSDS_MAX_RATE_0252` 0x00
- #define `RSDS_MAX_RATE_0504` 0x01
- #define `RSDS_MAX_RATE_1008` 0x02
- #define `RSDS_MAX_RATE_NOT_SPEC` 0x0f
- #define `RSDS_MAX_RATE_MASK` 0x0f
- #define `RSDS_NUM_LAYERS_MASK` 0x60
- #define `RSDS_NUM_LAYERS_SHIFT` 5
- #define `RSDS_NL_ONE_LAYER` 0x00
- #define `RSDS_NL_TWO_LAYERS` 0x20
- #define `RSDS_TRACK_PATH_MASK` 0x10
- #define `RSDS_TRACK_PATH_SHIFT` 4
- #define `RSDS_TP_PTP` 0x00
- #define `RSDS_TP_OTP` 0x10
- #define `RSDS_LAYER_TYPE_RO` 0x01
- #define `RSDS_LAYER_TYPE_RECORD` 0x02
- #define `RSDS_LAYER_TYPE_RW` 0x04
- #define `RSDS_LAYER_TYPE_MASK` 0x0f
- #define `RSDS_LIN_DENSITY_0267` 0x00
- #define `RSDS_LIN_DENSITY_0293` 0x10
- #define `RSDS_LIN_DENSITY_0409_0435` 0x20
- #define `RSDS_LIN_DENSITY_0280_0291` 0x40
- #define `RSDS_LIN_DENSITY_0353` 0x80
- #define `RSDS_LIN_DENSITY_MASK` 0xf0
- #define `RSDS_LIN_DENSITY_SHIFT` 4
- #define `RSDS_TRACK_DENSITY_074` 0x00
- #define `RSDS_TRACK_DENSITY_080` 0x01
- #define `RSDS_TRACK_DENSITY_0615` 0x02
- #define `RSDS_TRACK_DENSITY_MASK` 0x0f
- #define `RSDS_MAIN_DATA_START_DVD_RO` 0x30000
- #define `RSDS_MAIN_DATA_START_DVD_RW` 0x31000
- #define `RSDS_BCA` 0x80
- #define `RSDS_BCA_MASK` 0x80
- #define `RSDS_BCA_SHIFT` 7
- #define `RSDS_CPS_NOT_PRESENT` 0x00
- #define `RSDS_CPS_DATA_EXISTS` 0x01
- #define `RSDS_CPM_NO_COPYRIGHT` 0x00
- #define `RSDS_CPM_HAS_COPYRIGHT` 0x80
- #define `RSDS_CPM_MASK` 0x80
- #define `RSDS_CMGS_COPY_ALLOWED` 0x00
- #define `RSDS_CMGS_ONE_COPY` 0x20
- #define `RSDS_CMGS_NO_COPIES` 0x30
- #define `RSDS_CMGS_MASK` 0x30
- #define `RSDS_MS_CARTRIDGE` 0x80
- #define `RSDS_MS_OUT` 0x40
- #define `RSDS_MS_MSWI` 0x08
- #define `RSDS_MS_CWP` 0x04
- #define `RSDS_MS_PWP` 0x02

- #define `RDS_D_T_NEED_CARTRIDGE` 0x00
- #define `RDS_D_T_NO_CART_NEEDED` 0x01
- #define `RDS_D_SWI_NO_BARE` 0x01
- #define `RDS_D_SWI_UNSPEC` 0xff
- #define `SCSI_RCB`
- #define `RDS_D_ACTION_RECORDING` 0x0001
- #define `RDS_D_ACTION_READING` 0x0002
- #define `RDS_D_ACTION_FORMAT` 0x0004
- #define `RDS_D_ACTION_MODIFY_DCB` 0x0008
- #define `RDS_D_WPS_MS_WI` 0x08
- #define `RDS_D_WPS_CWP` 0x04
- #define `RDS_D_WPS_PWP` 0x02
- #define `RDS_D_WPS_SWPP` 0x01
- #define `RDS_D_SDS_NOT_WRITEABLE` 0x00
- #define `RDS_D_SDS_WRITEABLE` 0x80
- #define `RDS_D_SDS_MASK` 0x80
- #define `RDS_D_RDS_NOT_READABLE` 0x00
- #define `RDS_D_RDS_READABLE` 0x40
- #define `RDS_D_RDS_MASK` 0x40
- #define `CD_PAGE_CODE` 0x3F
- #define `AUDIO_PAGE` 0x0e
- #define `CD_PAGE_PS` 0x80
- #define `CD_PA_SOTC` 0x02
- #define `CD_PA_IMMED` 0x04
- #define `CD_PA_FORMAT_LBA` 0x0F
- #define `CD_PA_APR_VALID` 0x80
- #define `CHANNEL` 0x0F
- #define `CHANNEL_0` 1
- #define `CHANNEL_1` 2
- #define `CHANNEL_2` 4
- #define `CHANNEL_3` 8
- #define `LEFT_CHANNEL` CHANNEL_0
- #define `RIGHT_CHANNEL` CHANNEL_1
- #define `LEFT_PORT` 0
- #define `RIGHT_PORT` 1

Functions

- `__BEGIN_DECLS` void `scsi_report_key` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcn)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, u_int32_t lba, u_int8_t agid, u_int8_t key_format, u_int8_t *data_ptr, u_int32_t dxfer_len, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_send_key` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcn)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, u_int8_t agid, u_int8_t key_format, u_int8_t *data_ptr, u_int32_t dxfer_len, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_read_dvd_structure` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcn)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, u_int32_t address, u_int8_t layer_number, u_int8_t format, u_int8_t agid, u_int8_t *data_ptr, u_int32_t dxfer_len, u_int8_t sense_len, u_int32_t timeout)

7.19.1 Define Documentation

7.19.1.1 **#define _SCSI_SCSI_CD_H 1**

Definition at line 47 of file scsi_cd.h.

7.19.1.2 **#define AUDIO_PAGE 0x0e**

Definition at line 662 of file scsi_cd.h.

Referenced by cdiocctl().

7.19.1.3 **#define CD_MSF 0x02**

Definition at line 53 of file scsi_cd.h.

Referenced by cdreadsubchannel(), and cdreadtoc().

7.19.1.4 **#define CD_PA_APR_VALID 0x80**

Definition at line 671 of file scsi_cd.h.

7.19.1.5 **#define CD_PA_FORMAT_LBA 0x0F**

Definition at line 670 of file scsi_cd.h.

7.19.1.6 **#define CD_PA_IMMED 0x04**

Definition at line 667 of file scsi_cd.h.

Referenced by cdiocctl().

7.19.1.7 **#define CD_PA_SOTC 0x02**

Definition at line 666 of file scsi_cd.h.

Referenced by cdiocctl().

7.19.1.8 **#define CD_PAGE_CODE 0x3F**

Definition at line 661 of file scsi_cd.h.

7.19.1.9 **#define CD_PAGE_PS 0x80**

Definition at line 663 of file scsi_cd.h.

7.19.1.10 **#define CD_RELADDR 0x01**

Definition at line 52 of file scsi_cd.h.

7.19.1.11 #define CHANNEL 0x0F

Definition at line 676 of file scsi_cd.h.

7.19.1.12 #define CHANNEL_0 1

Definition at line 677 of file scsi_cd.h.

Referenced by cdiocctl().

7.19.1.13 #define CHANNEL_1 2

Definition at line 678 of file scsi_cd.h.

Referenced by cdiocctl().

7.19.1.14 #define CHANNEL_2 4

Definition at line 679 of file scsi_cd.h.

7.19.1.15 #define CHANNEL_3 8

Definition at line 680 of file scsi_cd.h.

7.19.1.16 #define LEFT_CHANNEL CHANNEL_0

Definition at line 681 of file scsi_cd.h.

Referenced by cdiocctl().

7.19.1.17 #define LEFT_PORT 0

Definition at line 685 of file scsi_cd.h.

Referenced by cdiocctl().

7.19.1.18 #define PA_PAUSE 1

Definition at line 67 of file scsi_cd.h.

7.19.1.19 #define PA_RESUME 0

Definition at line 68 of file scsi_cd.h.

7.19.1.20 #define PAUSE 0x4b

Definition at line 258 of file scsi_cd.h.

Referenced by cdpause().

7.19.1.21 #define PLAY_10 0x45

Definition at line 254 of file scsi_cd.h.

Referenced by cdplay().

7.19.1.22 #define PLAY_12 0xa5

Definition at line 261 of file scsi_cd.h.

Referenced by cdplay().

7.19.1.23 #define PLAY_MSF 0x47

Definition at line 255 of file scsi_cd.h.

Referenced by cdplaymsf().

7.19.1.24 #define PLAY_TRACK 0x48

Definition at line 256 of file scsi_cd.h.

Referenced by cdplaytracks().

7.19.1.25 #define PLAY_TRACK_REL 0x49

Definition at line 257 of file scsi_cd.h.

7.19.1.26 #define PLAY_TRACK_REL_BIG 0xa9

Definition at line 262 of file scsi_cd.h.

7.19.1.27 #define RDS_FORMAT_BCA 0x03

Definition at line 227 of file scsi_cd.h.

7.19.1.28 #define RDS_FORMAT_CMGS_CPM 0x05

Definition at line 229 of file scsi_cd.h.

7.19.1.29 #define RDS_FORMAT_COPYRIGHT 0x01

Definition at line 225 of file scsi_cd.h.

7.19.1.30 #define RDS_FORMAT_DCB 0x30

Definition at line 239 of file scsi_cd.h.

7.19.1.31 #define RDS_FORMAT_DDS 0x08

Definition at line 232 of file scsi_cd.h.

7.19.1.32 #define RDS_FORMAT_DISC_ID 0x0f

Definition at line 238 of file scsi_cd.h.

7.19.1.33 #define RDS_FORMAT_DISC_KEY 0x02

Definition at line 226 of file scsi_cd.h.

7.19.1.34 #define RDS_FORMAT_DISC_KEY_BLOCK 0x07

Definition at line 231 of file scsi_cd.h.

7.19.1.35 #define RDS_FORMAT_DVDRAM_MEDIA_STAT 0x09

Definition at line 233 of file scsi_cd.h.

7.19.1.36 #define RDS_FORMAT_LEADIN 0x0e

Definition at line 237 of file scsi_cd.h.

7.19.1.37 #define RDS_FORMAT_MANUFACTURER 0x04

Definition at line 228 of file scsi_cd.h.

7.19.1.38 #define RDS_FORMAT_PHYSICAL 0x00

Definition at line 224 of file scsi_cd.h.

7.19.1.39 #define RDS_FORMAT_PROT_DISCID 0x06

Definition at line 230 of file scsi_cd.h.

7.19.1.40 #define RDS_FORMAT_RMD 0x0d

Definition at line 236 of file scsi_cd.h.

7.19.1.41 #define RDS_FORMAT_RMD_BORDEROUT 0x0c

Definition at line 235 of file scsi_cd.h.

7.19.1.42 #define RDS_FORMAT_SPARE_AREA 0x0a

Definition at line 234 of file scsi_cd.h.

7.19.1.43 #define RDS_FORMAT_STRUCTURE_LIST 0xff

Definition at line 241 of file scsi_cd.h.

7.19.1.44 #define RDS_FORMAT_WRITE_PROT 0xc0

Definition at line 240 of file scsi_cd.h.

7.19.1.45 #define RSD_ACTION_FORMAT 0x0004

Definition at line 601 of file scsi_cd.h.

7.19.1.46 #define RSD_ACTION_MODIFY_DCB 0x0008

Definition at line 602 of file scsi_cd.h.

7.19.1.47 #define RSD_ACTION_READING 0x0002

Definition at line 600 of file scsi_cd.h.

7.19.1.48 #define RSD_ACTION_RECORDING 0x0001

Definition at line 599 of file scsi_cd.h.

7.19.1.49 #define RSD_BCA 0x80

Definition at line 424 of file scsi_cd.h.

7.19.1.50 #define RSD_BCA_MASK 0x80

Definition at line 425 of file scsi_cd.h.

Referenced by cdreadvdstructure().

7.19.1.51 #define RSD_BCA_SHIFT 7

Definition at line 426 of file scsi_cd.h.

Referenced by cdreadvdstructure().

7.19.1.52 #define RSD_BOOK_TYPE_DVD_PRW 0x90

Definition at line 371 of file scsi_cd.h.

7.19.1.53 #define RDS_D_BOOK_TYPE_DVD_R 0x20

Definition at line 369 of file scsi_cd.h.

7.19.1.54 #define RDS_D_BOOK_TYPE_DVD_RAM 0x10

Definition at line 368 of file scsi_cd.h.

7.19.1.55 #define RDS_D_BOOK_TYPE_DVD_ROM 0x00

Definition at line 367 of file scsi_cd.h.

7.19.1.56 #define RDS_D_BOOK_TYPE_DVD_RW 0x30

Definition at line 370 of file scsi_cd.h.

7.19.1.57 #define RDS_D_BOOK_TYPE_MASK 0xf0

Definition at line 372 of file scsi_cd.h.

Referenced by cdreadvdstructure().

7.19.1.58 #define RDS_D_BOOK_TYPE_SHIFT 4

Definition at line 373 of file scsi_cd.h.

Referenced by cdreadvdstructure().

7.19.1.59 #define RDS_D_BOOK_VERSION_MASK 0x0f

Definition at line 374 of file scsi_cd.h.

Referenced by cdreadvdstructure().

7.19.1.60 #define RDS_D_CMGS_COPY_ALLOWED 0x00

Definition at line 477 of file scsi_cd.h.

7.19.1.61 #define RDS_D_CMGS_MASK 0x30

Definition at line 480 of file scsi_cd.h.

7.19.1.62 #define RDS_D_CMGS_NO_COPIES 0x30

Definition at line 479 of file scsi_cd.h.

7.19.1.63 #define RDS_D_CMGS_ONE_COPY 0x20

Definition at line 478 of file scsi_cd.h.

7.19.1.64 #define RDS_D_CPM_HAS_COPYRIGHT 0x80

Definition at line 475 of file scsi_cd.h.

7.19.1.65 #define RDS_D_CPM_MASK 0x80

Definition at line 476 of file scsi_cd.h.

7.19.1.66 #define RDS_D_CPM_NO_COPYRIGHT 0x00

Definition at line 474 of file scsi_cd.h.

7.19.1.67 #define RDS_D_CPS_DATA_EXISTS 0x01

Definition at line 443 of file scsi_cd.h.

7.19.1.68 #define RDS_D_CPS_NOT_PRESENT 0x00

Definition at line 442 of file scsi_cd.h.

7.19.1.69 #define RDS_D_DISC_SIZE_120MM 0x00

Definition at line 380 of file scsi_cd.h.

7.19.1.70 #define RDS_D_DISC_SIZE_80MM 0x10

Definition at line 381 of file scsi_cd.h.

7.19.1.71 #define RDS_D_DISC_SIZE_MASK 0xf0

Definition at line 382 of file scsi_cd.h.

Referenced by cdreadvdstructure().

7.19.1.72 #define RDS_D_DISC_SIZE_SHIFT 4

Definition at line 383 of file scsi_cd.h.

Referenced by cdreadvdstructure().

7.19.1.73 #define RDS_D_DT_NEED_CARTRIDGE 0x00

Definition at line 519 of file scsi_cd.h.

7.19.1.74 #define RDS_D_T_NO_CART_NEEDED 0x01

Definition at line 520 of file scsi_cd.h.

7.19.1.75 #define RDS_D_LAYER_TYPE_MASK 0x0f

Definition at line 401 of file scsi_cd.h.

Referenced by cdreadvdstructure().

7.19.1.76 #define RDS_D_LAYER_TYPE_RECORD 0x02

Definition at line 399 of file scsi_cd.h.

7.19.1.77 #define RDS_D_LAYER_TYPE_RO 0x01

Definition at line 398 of file scsi_cd.h.

7.19.1.78 #define RDS_D_LAYER_TYPE_RW 0x04

Definition at line 400 of file scsi_cd.h.

7.19.1.79 #define RDS_D_LIN_DENSITY_0267 0x00

Definition at line 403 of file scsi_cd.h.

7.19.1.80 #define RDS_D_LIN_DENSITY_0280_0291 0x40

Definition at line 406 of file scsi_cd.h.

7.19.1.81 #define RDS_D_LIN_DENSITY_0293 0x10

Definition at line 404 of file scsi_cd.h.

7.19.1.82 #define RDS_D_LIN_DENSITY_0353 0x80

Definition at line 408 of file scsi_cd.h.

7.19.1.83 #define RDS_D_LIN_DENSITY_0409_0435 0x20

Definition at line 405 of file scsi_cd.h.

7.19.1.84 #define RDS_D_LIN_DENSITY_MASK 0xf0

Definition at line 409 of file scsi_cd.h.

Referenced by cdreadvdstructure().

7.19.1.85 #define RDSD_LIN_DENSITY_SHIFT 4

Definition at line 410 of file scsi_cd.h.

Referenced by cdreaddvdstructure().

7.19.1.86 #define RDSD_MAIN_DATA_START_DVD_RO 0x30000

Definition at line 417 of file scsi_cd.h.

7.19.1.87 #define RDSD_MAIN_DATA_START_DVD_RW 0x31000

Definition at line 418 of file scsi_cd.h.

7.19.1.88 #define RDSD_MAX_RATE_0252 0x00

Definition at line 384 of file scsi_cd.h.

7.19.1.89 #define RDSD_MAX_RATE_0504 0x01

Definition at line 385 of file scsi_cd.h.

7.19.1.90 #define RDSD_MAX_RATE_1008 0x02

Definition at line 386 of file scsi_cd.h.

7.19.1.91 #define RDSD_MAX_RATE_MASK 0x0f

Definition at line 388 of file scsi_cd.h.

Referenced by cdreaddvdstructure().

7.19.1.92 #define RDSD_MAX_RATE_NOT_SPEC 0x0f

Definition at line 387 of file scsi_cd.h.

7.19.1.93 #define RDSD_MS_CARTRIDGE 0x80

Definition at line 513 of file scsi_cd.h.

7.19.1.94 #define RDSD_MS_CWP 0x04

Definition at line 516 of file scsi_cd.h.

7.19.1.95 #define RDSD_MS_MSWI 0x08

Definition at line 515 of file scsi_cd.h.

7.19.1.96 #define RDSD_MS_OUT 0x40

Definition at line 514 of file scsi_cd.h.

7.19.1.97 #define RDSD_MS_PWP 0x02

Definition at line 517 of file scsi_cd.h.

7.19.1.98 #define RDSD_NL_ONE_LAYER 0x00

Definition at line 392 of file scsi_cd.h.

7.19.1.99 #define RDSD_NL_TWO_LAYERS 0x20

Definition at line 393 of file scsi_cd.h.

7.19.1.100 #define RDSD_NUM_LAYERS_MASK 0x60

Definition at line 390 of file scsi_cd.h.

Referenced by cdreaddvdstructure().

7.19.1.101 #define RDSD_NUM_LAYERS_SHIFT 5

Definition at line 391 of file scsi_cd.h.

Referenced by cdreaddvdstructure().

7.19.1.102 #define RDSD_RDS_MASK 0x40

Definition at line 635 of file scsi_cd.h.

7.19.1.103 #define RDSD_RDS_NOT_READABLE 0x00

Definition at line 633 of file scsi_cd.h.

7.19.1.104 #define RDSD_RDS_READABLE 0x40

Definition at line 634 of file scsi_cd.h.

7.19.1.105 #define RDSD_SDS_MASK 0x80

Definition at line 632 of file scsi_cd.h.

7.19.1.106 #define RDSD_SDS_NOT_WRITEABLE 0x00

Definition at line 630 of file scsi_cd.h.

7.19.1.107 #define RDSO_SDS_WRITEABLE 0x80

Definition at line 631 of file scsi_cd.h.

7.19.1.108 #define RDSO_SWI_NO_BARE 0x01

Definition at line 523 of file scsi_cd.h.

7.19.1.109 #define RDSO_SWI_UNSPEC 0xff

Definition at line 524 of file scsi_cd.h.

7.19.1.110 #define RDSO_TP_OTP 0x10

Definition at line 397 of file scsi_cd.h.

7.19.1.111 #define RDSO_TP_PTP 0x00

Definition at line 396 of file scsi_cd.h.

7.19.1.112 #define RDSO_TRACK_DENSITY_0615 0x02

Definition at line 413 of file scsi_cd.h.

7.19.1.113 #define RDSO_TRACK_DENSITY_074 0x00

Definition at line 411 of file scsi_cd.h.

7.19.1.114 #define RDSO_TRACK_DENSITY_080 0x01

Definition at line 412 of file scsi_cd.h.

7.19.1.115 #define RDSO_TRACK_DENSITY_MASK 0x0f

Definition at line 414 of file scsi_cd.h.

Referenced by cdreadvdstructure().

7.19.1.116 #define RDSO_TRACK_PATH_MASK 0x10

Definition at line 394 of file scsi_cd.h.

Referenced by cdreadvdstructure().

7.19.1.117 #define RDS_D_TRACK_PATH_SHIFT 4

Definition at line 395 of file scsi_cd.h.

Referenced by cdreaddvdstructure().

7.19.1.118 #define RDS_D_WPS_CWP 0x04

Definition at line 620 of file scsi_cd.h.

7.19.1.119 #define RDS_D_WPS_MS WI 0x08

Definition at line 619 of file scsi_cd.h.

7.19.1.120 #define RDS_D_WPS_PWP 0x02

Definition at line 621 of file scsi_cd.h.

7.19.1.121 #define RDS_D_WPS_SWPP 0x01

Definition at line 622 of file scsi_cd.h.

7.19.1.122 #define READ_CD_CAPACITY 0x25

Definition at line 250 of file scsi_cd.h.

7.19.1.123 #define READ_DVD_STRUCTURE 0xad

Definition at line 263 of file scsi_cd.h.

Referenced by scsi_read_dvd_structure().

7.19.1.124 #define READ_HEADER 0x44

Definition at line 253 of file scsi_cd.h.

7.19.1.125 #define READ_SUBCHANNEL 0x42

Definition at line 251 of file scsi_cd.h.

Referenced by cdreadsubchannel().

7.19.1.126 #define READ_TOC 0x43

Definition at line 252 of file scsi_cd.h.

Referenced by cdreadtoc().

7.19.1.127 #define REPORT_KEY 0xa4

Definition at line 260 of file scsi_cd.h.

Referenced by scsi_report_key().

7.19.1.128 #define RF_KF_ASF 0x05

Definition at line 198 of file scsi_cd.h.

7.19.1.129 #define RF_KF_INV_AGID 0x3f

Definition at line 201 of file scsi_cd.h.

7.19.1.130 #define RF_KF_KEY1 0x02

Definition at line 195 of file scsi_cd.h.

7.19.1.131 #define RF_KF_RPC_REPORT 0x08

Definition at line 200 of file scsi_cd.h.

7.19.1.132 #define RF_KF_TITLE 0x04

Definition at line 197 of file scsi_cd.h.

7.19.1.133 #define RIGHT_CHANNEL CHANNEL_1

Definition at line 682 of file scsi_cd.h.

Referenced by cdiocctl().

7.19.1.134 #define RIGHT_PORT 1

Definition at line 686 of file scsi_cd.h.

Referenced by cdiocctl().

7.19.1.135 #define RK_KF_AGID 0x00

Definition at line 193 of file scsi_cd.h.

7.19.1.136 #define RK_KF_AGID_MASK 0xc0

Definition at line 190 of file scsi_cd.h.

7.19.1.137 #define RK_KF_AGID_SHIFT 6

Definition at line 191 of file scsi_cd.h.

Referenced by scsi_report_key(), and scsi_send_key().

7.19.1.138 #define RK_KF_CHALLENGE 0x01

Definition at line 194 of file scsi_cd.h.

7.19.1.139 #define RK_KF_KEY2 0x03

Definition at line 196 of file scsi_cd.h.

7.19.1.140 #define RK_KF_KEYFORMAT_MASK 0x3f

Definition at line 192 of file scsi_cd.h.

Referenced by scsi_report_key(), and scsi_send_key().

7.19.1.141 #define RK_KF_RPC_SET 0x06

Definition at line 199 of file scsi_cd.h.

7.19.1.142 #define RKD_AGID_MASK 0xc0

Definition at line 277 of file scsi_cd.h.

Referenced by cdreportkey().

7.19.1.143 #define RKD_AGID_SHIFT 6

Definition at line 278 of file scsi_cd.h.

Referenced by cdreportkey().

7.19.1.144 #define RKD_ASF_SUCCESS 0x01

Definition at line 321 of file scsi_cd.h.

Referenced by cdreportkey().

7.19.1.145 #define RKD_RPC_SCHEME_PHASE_II 0x01

Definition at line 329 of file scsi_cd.h.

7.19.1.146 #define RKD_RPC_SCHEME_UNKNOWN 0x00

Definition at line 328 of file scsi_cd.h.

7.19.1.147 #define RKD_RPC_TYPE_LAST_CHANCE 0x80

Definition at line 336 of file scsi_cd.h.

7.19.1.148 #define RKD_RPC_TYPE_MASK 0xC0

Definition at line 332 of file scsi_cd.h.

Referenced by cdreportkey().

7.19.1.149 #define RKD_RPC_TYPE_NONE 0x00

Definition at line 334 of file scsi_cd.h.

7.19.1.150 #define RKD_RPC_TYPE_PERM 0xC0

Definition at line 337 of file scsi_cd.h.

7.19.1.151 #define RKD_RPC_TYPE_SET 0x40

Definition at line 335 of file scsi_cd.h.

7.19.1.152 #define RKD_RPC_TYPE_SHIFT 6

Definition at line 333 of file scsi_cd.h.

Referenced by cdreportkey().

7.19.1.153 #define RKD_RPC_USER_RESET_MASK 0x07

Definition at line 340 of file scsi_cd.h.

Referenced by cdreportkey().

7.19.1.154 #define RKD_RPC_USER_RESET_SHIFT 0

Definition at line 341 of file scsi_cd.h.

7.19.1.155 #define RKD_RPC_VENDOR_RESET_MASK 0x38

Definition at line 338 of file scsi_cd.h.

Referenced by cdreportkey().

7.19.1.156 #define RKD_RPC_VENDOR_RESET_SHIFT 3

Definition at line 339 of file scsi_cd.h.

Referenced by cdreportkey().

7.19.1.157 #define RKD_TITLE_CMGS_1_GEN 0x20

Definition at line 310 of file scsi_cd.h.

7.19.1.158 #define RKD_TITLE_CMGS_MASK 0x30

Definition at line 306 of file scsi_cd.h.

Referenced by cdreportkey().

7.19.1.159 #define RKD_TITLE_CMGS_NO_COPY 0x30

Definition at line 311 of file scsi_cd.h.

7.19.1.160 #define RKD_TITLE_CMGS_NO_RST 0x00

Definition at line 308 of file scsi_cd.h.

7.19.1.161 #define RKD_TITLE_CMGS_RSVD 0x10

Definition at line 309 of file scsi_cd.h.

7.19.1.162 #define RKD_TITLE_CMGS_SHIFT 4

Definition at line 307 of file scsi_cd.h.

Referenced by cdreportkey().

7.19.1.163 #define RKD_TITLE_CP_SEC 0x40

Definition at line 304 of file scsi_cd.h.

Referenced by cdreportkey().

7.19.1.164 #define RKD_TITLE_CP_SEC_SHIFT 6

Definition at line 305 of file scsi_cd.h.

Referenced by cdreportkey().

7.19.1.165 #define RKD_TITLE_CPM 0x80

Definition at line 302 of file scsi_cd.h.

Referenced by cdreportkey().

7.19.1.166 #define RKD_TITLE_CPM_SHIFT 7

Definition at line 303 of file scsi_cd.h.

Referenced by cdreportkey().

7.19.1.167 #define SCSI_RCB

Definition at line 597 of file scsi_cd.h.

7.19.1.168 #define SEND_KEY 0xa3

Definition at line 259 of file scsi_cd.h.

Referenced by scsi_send_key().

7.19.1.169 #define SET_CD_SPEED 0xbb

Definition at line 264 of file scsi_cd.h.

Referenced by cdsetspeed().

7.19.1.170 #define SRS_SUBQ 0x40

Definition at line 142 of file scsi_cd.h.

Referenced by cdreadsubchannel().

7.19.2 Function Documentation

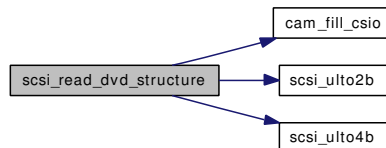
7.19.2.1 void `scsi_read_dvd_structure` (struct `ccb_scsiio` * *csio*, u_int32_t *retries*, void(*) (struct `cam_periph` *, union `ccb` *) *cbfcnp*, u_int8_t *tag_action*, u_int32_t *address*, u_int8_t *layer_number*, u_int8_t *format*, u_int8_t *agid*, u_int8_t * *data_ptr*, u_int32_t *dxfer_len*, u_int8_t *sense_len*, u_int32_t *timeout*)

Definition at line 4198 of file scsi_cd.c.

References CAM_DIR_IN, `cam_fill_csio`(), `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, READ_DVD_STRUCTURE, `scsi_ulto2b`(), and `scsi_ulto4b`().

Referenced by `cdreaddvdstructure`().

Here is the call graph for this function:



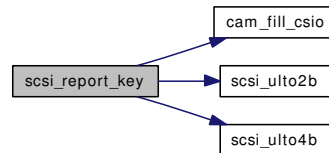
7.19.2.2 __BEGIN_DECLS void `scsi_report_key` (struct `ccb_scsiio` * *csio*, u_int32_t *retries*, void(*) (struct `cam_periph` *, union `ccb` *) *cbfcnp*, u_int8_t *tag_action*, u_int32_t *lba*, u_int8_t *agid*, u_int8_t *key_format*, u_int8_t * *data_ptr*, u_int32_t *dxfer_len*, u_int8_t *sense_len*, u_int32_t *timeout*)

Definition at line 4139 of file scsi_cd.c.

References CAM_DIR_IN, CAM_DIR_NONE, cam_fill_csio(), cdb_t::cdb_bytes, ccb_scsiio::cdb_io, REPORT_KEY, RK_KF_AGID_SHIFT, RK_KF_KEYFORMAT_MASK, scsi_ulto2b(), and scsi_ulto4b().

Referenced by cdreportkey().

Here is the call graph for this function:



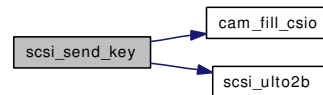
7.19.2.3 void `scsi_send_key` (struct `ccb_scsiio` * `csio`, `u_int32_t` `retries`, void(*) (struct `cam_periph` *, union `ccb` *) `cbfcn`, `u_int8_t` `tag_action`, `u_int8_t` `agid`, `u_int8_t` `key_format`, `u_int8_t` * `data_ptr`, `u_int32_t` `dxfer_len`, `u_int8_t` `sense_len`, `u_int32_t` `timeout`)

Definition at line 4168 of file `scsi_cd.c`.

References CAM_DIR_OUT, cam_fill_csio(), cdb_t::cdb_bytes, ccb_scsiio::cdb_io, RK_KF_AGID_SHIFT, RK_KF_KEYFORMAT_MASK, scsi_ulto2b(), and SEND_KEY.

Referenced by cdsendkey().

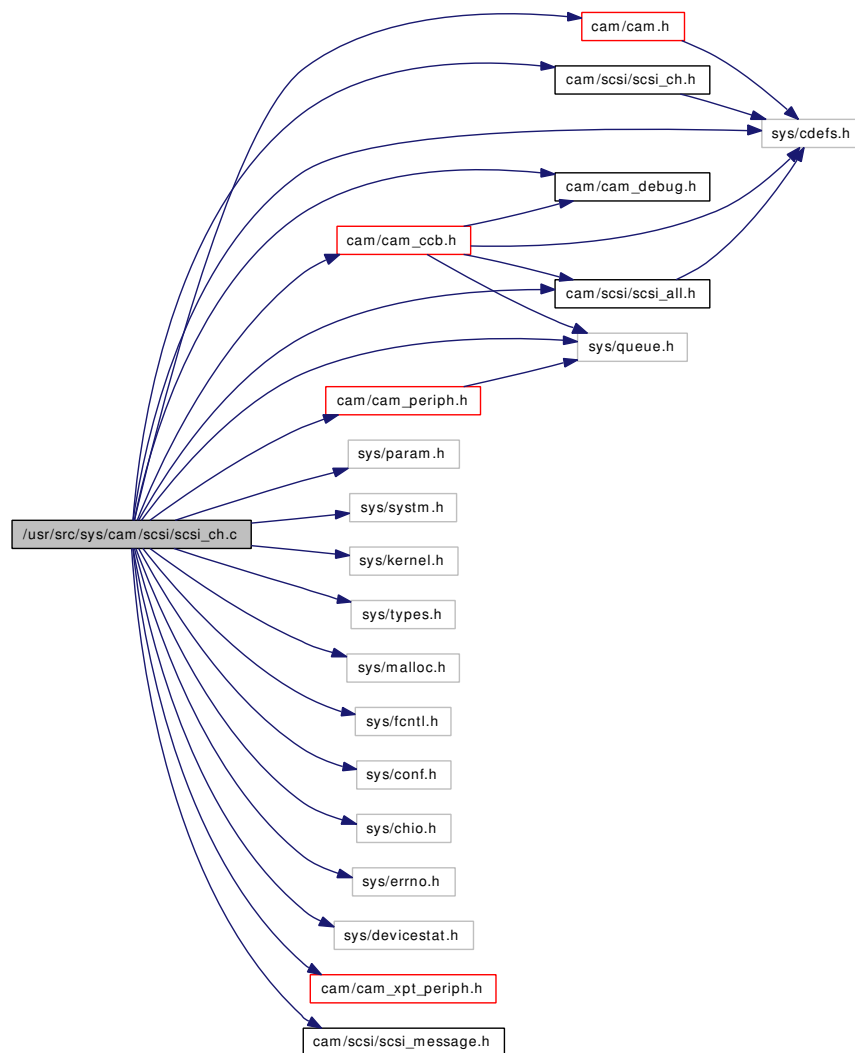
Here is the call graph for this function:



7.20 /usr/src/sys/cam/scsi/scsi_ch.c File Reference

```
#include <sys/cdefs.h>
#include <sys/param.h>
#include <sys/queue.h>
#include <sys/system.h>
#include <sys/kernel.h>
#include <sys/types.h>
#include <sys/malloc.h>
#include <sys/fcntl.h>
#include <sys/conf.h>
#include <sys/chio.h>
#include <sys/errno.h>
#include <sys/devicestat.h>
#include <cam/cam.h>
#include <cam/cam_ccb.h>
#include <cam/cam_periph.h>
#include <cam/cam_xpt_periph.h>
#include <cam/cam_debug.h>
#include <cam/scsi/scsi_all.h>
#include <cam/scsi/scsi_message.h>
#include <cam/scsi/scsi_ch.h>
```

Include dependency graph for scsi_ch.c:



Data Structures

- struct [scsi_mode_sense_data](#)
- struct [ch_softc](#)

Defines

- #define [ccb_state](#) ppriv_field0
- #define [ccb_bp](#) ppriv_ptr1
- #define [CHUNIT\(x\)](#) (minor((x)))
- #define [PLURAL\(c\)](#) (c) == 1 ? "" : "s"

Enumerations

- enum [ch_flags](#) { [CH_FLAG_INVALID](#) = 0x001, [CH_FLAG_OPEN](#) = 0x002 }
- enum [ch_state](#) { [CH_STATE_PROBE](#), [CH_STATE_NORMAL](#) }

- enum `ch_ccb_types` { `CH_CCB_PROBE`, `CH_CCB_WAITING` }
- enum `ch_quirks` { `CH_Q_NONE` = 0x00, `CH_Q_NO_DBD` = 0x01 }

Functions

- `__FBSDID` ("\$FreeBSD: src/sys/cam/scsi/scsi_ch.c,v 1.43 2006/12/05 07:45:28 mjacob Exp \$")
- static void `chasync` (void *callback_arg, u_int32_t code, struct `cam_path` *path, void *arg)
- static void `chdone` (struct `cam_periph` *periph, union `ccb` *done_ccb)
- static int `cherror` (union `ccb` *ccb, u_int32_t `cam_flags`, u_int32_t sense_flags)
- static int `chmove` (struct `cam_periph` *periph, struct `changer_move` *cm)
- static int `chexchange` (struct `cam_periph` *periph, struct `changer_exchange` *ce)
- static int `chposition` (struct `cam_periph` *periph, struct `changer_position` *cp)
- static int `chgetelemstatus` (struct `cam_periph` *periph, struct `changer_element_status_request` *csr)
- static int `chsetvoltage` (struct `cam_periph` *periph, struct `changer_set_voltage_request` *csvr)
- static int `chielem` (struct `cam_periph` *periph, unsigned int timeout)
- static int `chgetparams` (struct `cam_periph` *periph)
- `PERIPHDRIIVER_DECLARE` (ch, chdriver)
- static void `chinit` (void)
- static void `choninvalidate` (struct `cam_periph` *periph)
- static void `chcleanup` (struct `cam_periph` *periph)
- static `cam_status` `chregister` (struct `cam_periph` *periph, void *arg)
- static int `chopen` (struct `cdev` *dev, int flags, int fmt, struct `thread` *td)
- static int `chclose` (struct `cdev` *dev, int flag, int fmt, struct `thread` *td)
- static void `chstart` (struct `cam_periph` *periph, union `ccb` *start_ccb)
- static int `chioctl` (struct `cdev` *dev, u_long cmd, `caddr_t` addr, int flag, struct `thread` *td)
- static void `copy_voltage` (struct `changer_voltage` *uvoltage, struct `volume_tag` *voltage)
- static void `copy_element_status` (struct `ch_softc` *softc, u_int16_t flags, struct `read_element_status_descriptor` *desc, struct `changer_element_status` *ces)
- void `scsi_move_medium` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcn)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, u_int32_t tea, u_int32_t src, u_int32_t dst, int invert, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_exchange_medium` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcn)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, u_int32_t tea, u_int32_t src, u_int32_t dst1, u_int32_t dst2, int invert1, int invert2, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_position_to_element` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcn)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, u_int32_t tea, u_int32_t dst, int invert, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_read_element_status` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcn)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, int voltage, u_int32_t sea, u_int32_t count, u_int8_t *data_ptr, u_int32_t dxfer_len, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_initialize_element_status` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcn)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_send_volume_tag` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcn)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, u_int16_t element_address, u_int8_t send_action_code, struct `scsi_send_volume_tag_parameters` *parameters, u_int8_t sense_len, u_int32_t timeout)

Variables

- static const u_int32_t [CH_TIMEOUT_MODE_SENSE](#) = 6000
- static const u_int32_t [CH_TIMEOUT_MOVE_MEDIUM](#) = 100000
- static const u_int32_t [CH_TIMEOUT_EXCHANGE_MEDIUM](#) = 100000
- static const u_int32_t [CH_TIMEOUT_POSITION_TO_ELEMENT](#) = 100000
- static const u_int32_t [CH_TIMEOUT_READ_ELEMENT_STATUS](#) = 10000
- static const u_int32_t [CH_TIMEOUT_SEND_VOLTAGE](#) = 10000
- static const u_int32_t [CH_TIMEOUT_INITIALIZE_ELEMENT_STATUS](#) = 500000
- static d_open_t [chopen](#)
- static d_close_t [chclose](#)
- static d_ioctl_t [chioctl](#)
- static [periph_init_t](#) [chinit](#)
- static [periph_ctor_t](#) [chregister](#)
- static [periph_oninv_t](#) [choninvalidate](#)
- static [periph_dtor_t](#) [chcleanup](#)
- static [periph_start_t](#) [chstart](#)
- static struct [periph_driver](#) [chdriver](#)
- static struct cdevsw [ch_cdevsw](#)

7.20.1 Define Documentation

7.20.1.1 #define ccb_bp ppriv_ptr1

Definition at line 130 of file scsi_ch.c.

7.20.1.2 #define ccb_state ppriv_field0

Definition at line 129 of file scsi_ch.c.

7.20.1.3 #define CHUNIT(x) (minor((x)))

Definition at line 176 of file scsi_ch.c.

7.20.1.4 #define PLURAL(c) (c) == 1 ? "" : "s"

Referenced by [chdone\(\)](#).

7.20.2 Enumeration Type Documentation

7.20.2.1 enum [ch_ccb_types](#)

Enumerator:

CH_CCB_PROBE
CH_CCB_WAITING

Definition at line 119 of file scsi_ch.c.

7.20.2.2 enum [ch_flags](#)

Enumerator:

CH_FLAG_INVALID

CH_FLAG_OPEN

Definition at line 109 of file `scsi_ch.c`.

7.20.2.3 enum [ch_quirks](#)

Enumerator:

CH_Q_NONE

CH_Q_NO_DBD

Definition at line 124 of file `scsi_ch.c`.

7.20.2.4 enum [ch_state](#)

Enumerator:

CH_STATE_PROBE

CH_STATE_NORMAL

Definition at line 114 of file `scsi_ch.c`.

7.20.3 Function Documentation

7.20.3.1 `__FBSDID` ("FreeBSD: src/sys/cam/scsi/scsi_ch. c, v 1.43 2006/12/05 07:45:28 mjacob Exp \$")

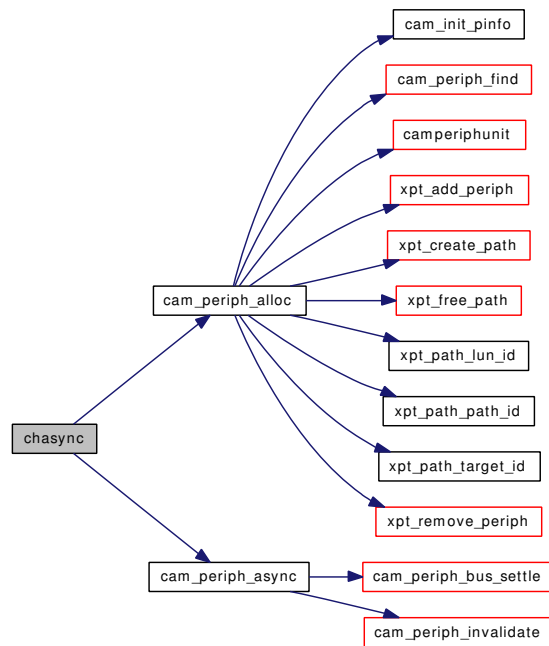
7.20.3.2 `static void chasync` (`void * callback_arg`, `u_int32_t code`, `struct cam_path * path`, `void * arg`) [`static`]

Definition at line 294 of file `scsi_ch.c`.

References `AC_FOUND_DEVICE`, `cam_periph_alloc()`, `cam_periph_async()`, `CAM_PERIPH_BIO`, `CAM_REQ_CMP`, `CAM_REQ_INPROG`, `ccb_getdev::ccb_h`, `chcleanup`, `choninvalidate`, `chregister`, `chstart`, `ccb_getdev::inq_data`, `ccb_hdr::path`, `SID_TYPE`, and `T_CHANGER`.

Referenced by `chinit()`, `choninvalidate()`, and `chregister()`.

Here is the call graph for this function:

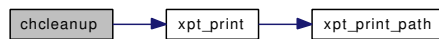


7.20.3.3 `static void chcleanup (struct cam_periph * periph)` [static]

Definition at line 281 of file `scsi_ch.c`.

References `ch_softc::dev`, `ch_softc::device_stats`, `cam_periph::path`, `cam_periph::softc`, and `xpt_print()`.

Here is the call graph for this function:

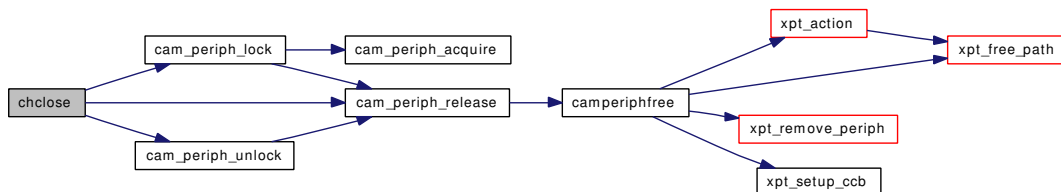


7.20.3.4 `static int chclose (struct cdev * dev, int flag, int fmt, struct thread * td)` [static]

Definition at line 454 of file `scsi_ch.c`.

References `cam_periph_lock()`, `cam_periph_release()`, `cam_periph_unlock()`, `CH_FLAG_OPEN`, `ch_softc::flags`, and `cam_periph::softc`.

Here is the call graph for this function:



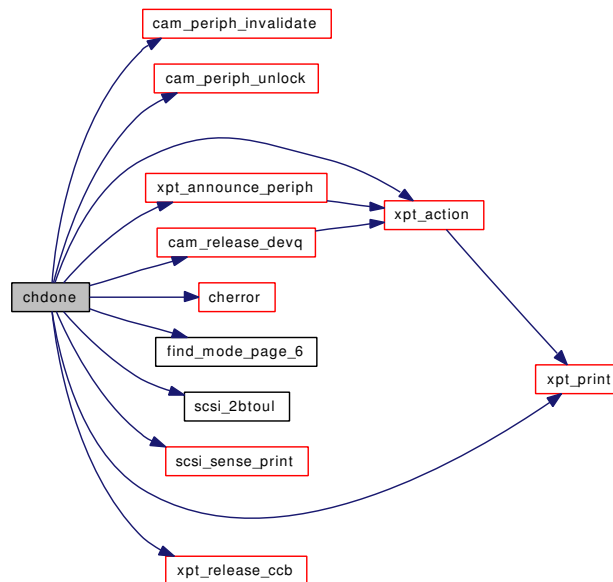
7.20.3.5 static void chdone (struct **cam_periph** * *periph*, union **ccb** * *done_ccb*) [static]

Definition at line 549 of file scsi_ch.c.

References `scsi_mode_sense_6::byte2`, `cam_periph_invalidate()`, `cam_periph_unlock()`, `cam_release_devq()`, `CAM_REQ_CMP`, `CAM_RETRY_SELTO`, `CAM_SCSI_STATUS_ERROR`, `CAM_STATUS_MASK`, `ccb_hdr::cbfnp`, `ccb::ccb_h`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `CH_CCB_PROBE`, `CH_CCB_WAITING`, `CH_Q_NO_DBD`, `CH_STATE_NORMAL`, `cherror()`, `ccb::csio`, `ccb_scsiio::data_ptr`, `page_element_address_assignment::fdtea`, `page_element_address_assignment::fieea`, `find_mode_page_6()`, `page_element_address_assignment::fsea`, `page_element_address_assignment::mtea`, `page_element_address_assignment::ndte`, `page_element_address_assignment::niee`, `page_element_address_assignment::nmte`, `page_element_address_assignment::nse`, `cam_periph::path`, `ccb_hdr::path`, `PLURAL`, `ch_softc::quirks`, `ch_softc::sc_counts`, `ch_softc::sc_firsts`, `ch_softc::sc_picker`, `scsi_2btoul()`, `scsi_sense_print()`, `SF_NO_PRINT`, `SF_RETRY_UA`, `SMS_DBD`, `cam_periph::softc`, `ch_softc::state`, `ccb_hdr::status`, `xpt_action()`, `xpt_announce_periph()`, `xpt_print()`, and `xpt_release_ccb()`.

Referenced by `chexchange()`, `chgetelemstatus()`, `chgetparams()`, `chielem()`, `chmove()`, `chposition()`, and `chstart()`.

Here is the call graph for this function:



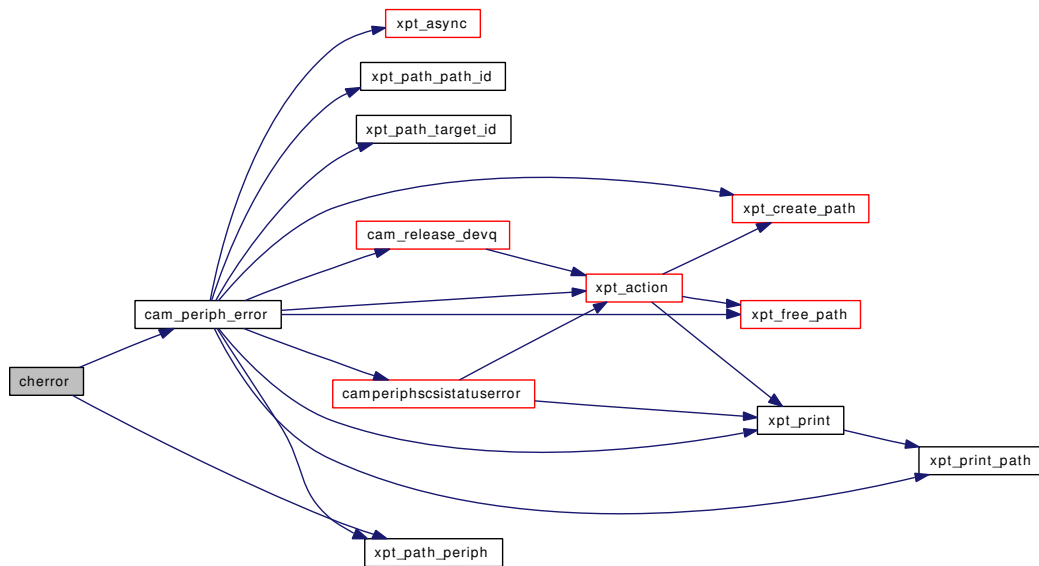
7.20.3.6 static int cherror (union **ccb** * *ccb*, u_int32_t *cam_flags*, u_int32_t *sense_flags*) [static]

Definition at line 689 of file scsi_ch.c.

References `cam_periph_error()`, `ccb::ccb_h`, `ccb_hdr::path`, `ch_softc::saved_ccb`, `cam_periph::softc`, and `xpt_path_periph()`.

Referenced by `chdone()`, `chexchange()`, `chgetelemstatus()`, `chgetparams()`, `chielem()`, `chioctl()`, `chmove()`, and `chposition()`.

Here is the call graph for this function:



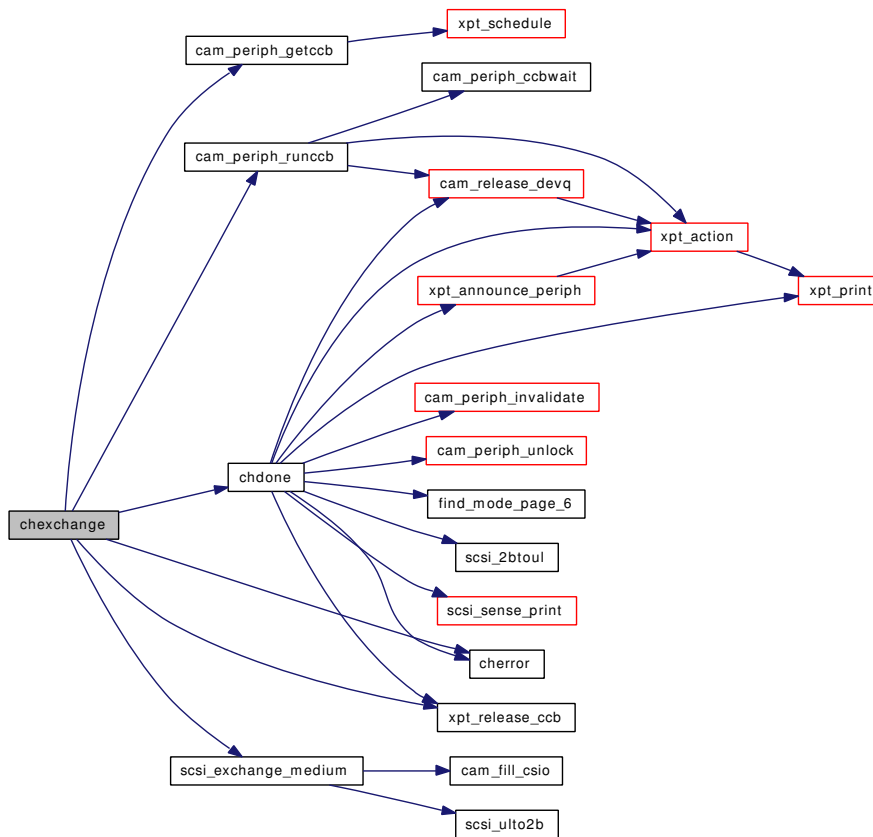
7.20.3.7 static int chexchange (struct [cam_periph](#) * *periph*, struct [changer_exchange](#) * *ce*) [static]

Definition at line 855 of file [scsi_ch.c](#).

References [cam_periph_getccb\(\)](#), [cam_periph_runccb\(\)](#), [CAM_RETRY_SELTO](#), [CH_TIMEOUT_EXCHANGE_MEDIUM](#), [chdone\(\)](#), [cherror\(\)](#), [ccb::csio](#), [ch_softc::device_stats](#), [MSG_SIMPLE_Q_TAG](#), [ch_softc::sc_counts](#), [ch_softc::sc_exchangemask](#), [ch_softc::sc_firsts](#), [ch_softc::sc_picker](#), [scsi_exchange_medium\(\)](#), [SF_RETRY_UA](#), [cam_periph::softc](#), [SSD_FULL_SIZE](#), and [xpt_release_ccb\(\)](#).

Referenced by [chioctl\(\)](#).

Here is the call graph for this function:



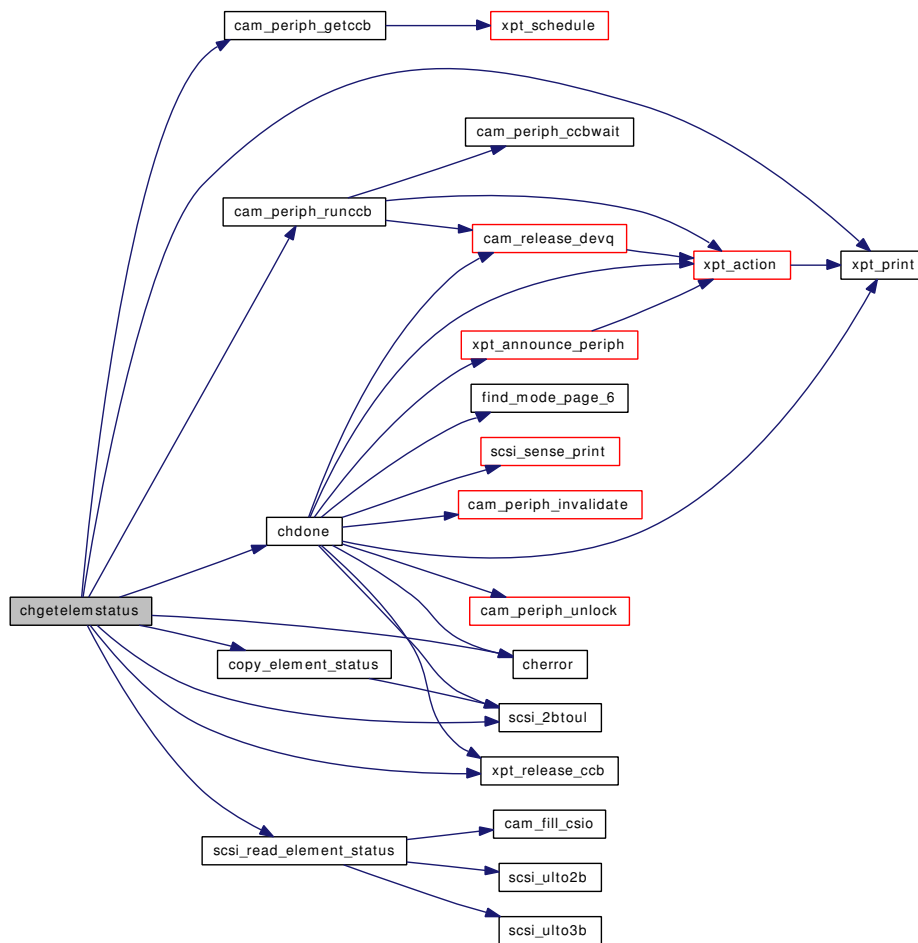
7.20.3.8 static int chgetelemstatus (struct [cam_periph](#) * *periph*, struct [changer_element_status_request](#) * *csr*) [static]

Definition at line 1060 of file `scsi_ch.c`.

References `cam_periph_getccb()`, `cam_periph_runccb()`, `CAM_RETRY_SELTO`, `CH_TIMEOUT_READ_ELEMENT_STATUS`, `chdone()`, `cherror()`, `copy_element_status()`, `read_element_status_header::count`, `ccb::csio`, `ch_softc::device_stats`, `read_element_status_page_header::edl`, `read_element_status_page_header::flags`, `MSG_SIMPLE_Q_TAG`, `cam_periph::path`, `ch_softc::sc_counts`, `ch_softc::sc_firsts`, `scsi_2btoul()`, `scsi_read_element_status()`, `SF_RETRY_UA`, `cam_periph::softc`, `SSD_FULL_SIZE`, `xpt_print()`, and `xpt_release_ccb()`.

Referenced by `chioctl()`.

Here is the call graph for this function:



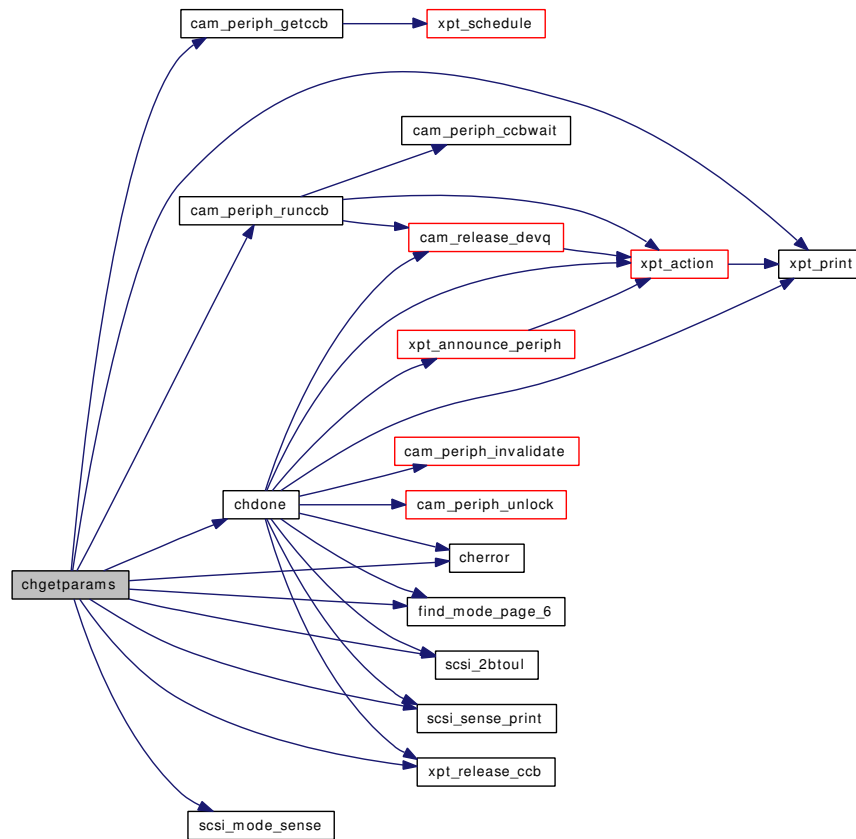
7.20.3.9 static int chgetparams (struct [cam_periph](#) * *periph*) [static]

Definition at line 1325 of file `scsi_ch.c`.

References `scsi_mode_sense_6::byte2`, `cam_periph_getccb()`, `cam_periph_runccb()`, `CAM_RETRY_SELTO`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `CH_DEVICE_CAP_PAGE`, `CH_ELEMENT_ADDR_ASSIGN_PAGE`, `CH_Q_NO_DBD`, `CH_TIMEOUT_MODE_SENSE`, `chdone()`, `cherror()`, `ccb::csio`, `ch_softc::device_stats`, `page_device_capabilities::exchange_with`, `page_element_address_assignment::fdtea`, `page_element_address_assignment::fieea`, `find_mode_page_6()`, `page_element_address_assignment::fsea`, `page_device_capabilities::move_from`, `MSG_SIMPLE_Q_TAG`, `page_element_address_assignment::mtea`, `page_element_address_assignment::ndte`, `page_element_address_assignment::niee`, `page_element_address_assignment::nmte`, `page_element_address_assignment::nse`, `cam_periph::path`, `ch_softc::quirks`, `ch_softc::sc_counts`, `ch_softc::sc_exchangemask`, `ch_softc::sc_firsts`, `ch_softc::sc_movemask`, `scsi_2btoul()`, `scsi_mode_sense()`, `scsi_sense_print()`, `SF_NO_PRINT`, `SF_RETRY_UA`, `SMS_DBD`, `SMS_PAGE_CTRL_CURRENT`, `cam_periph::softc`, `SSD_FULL_SIZE`, `xpt_print()`, and `xpt_release_ccb()`.

Referenced by `chopen()`.

Here is the call graph for this function:



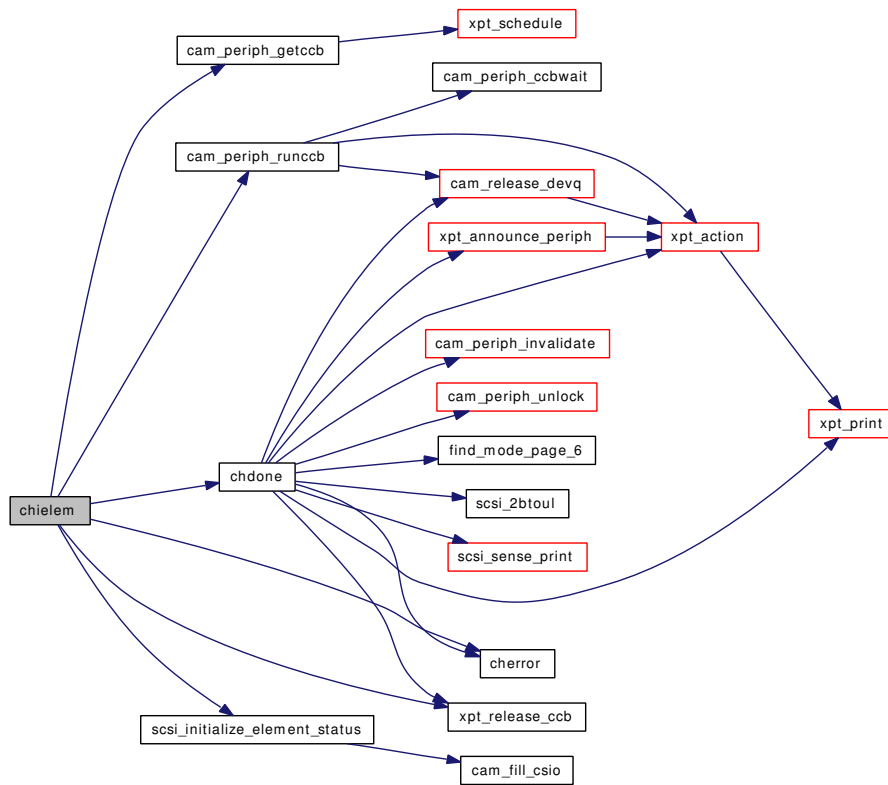
7.20.3.10 static int chielem (struct **cam_periph** * *periph*, unsigned int *timeout*) [static]

Definition at line 1202 of file `scsi_ch.c`.

References `cam_periph_getccb()`, `cam_periph_runccb()`, `CAM_RETRY_SELTO`, `CH_TIMEOUT_INITIALIZE_ELEMENT_STATUS`, `chdone()`, `cherror()`, `ccb::csio`, `ch_softc::device_stats`, `MSG_SIMPLE_Q_TAG`, `scsi_initialize_element_status()`, `SF_RETRY_UA`, `cam_periph::softc`, `SSD_FULL_SIZE`, and `xpt_release_ccb()`.

Referenced by `chioctl()`.

Here is the call graph for this function:

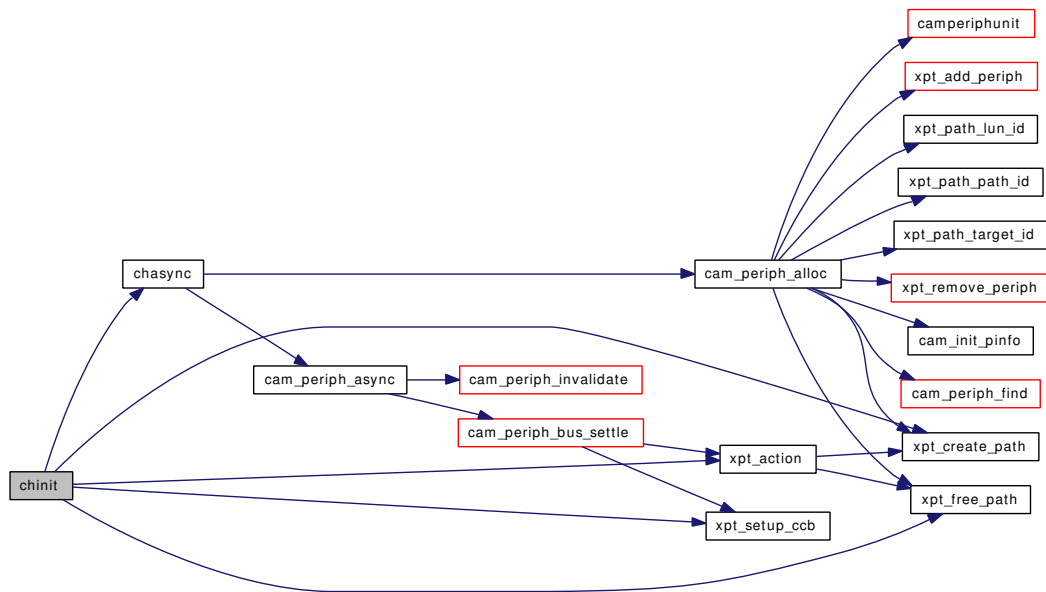


7.20.3.11 static void chinit (void) [static]

Definition at line 224 of file scsi_ch.c.

References AC_FOUND_DEVICE, ccb_setasync::callback, ccb_setasync::callback_arg, CAM_LUN_WILDCARD, CAM_REQ_CMP, CAM_TARGET_WILDCARD, CAM_XPT_PATH_ID, ccb::ccb_h, ccb_setasync::ccb_h, chasync(), ccb_setasync::event_enable, ccb_hdr::func_code, ccb_hdr::status, xpt_action(), xpt_create_path(), xpt_free_path(), XPT_SASYNC_CB, and xpt_setup_ccb().

Here is the call graph for this function:

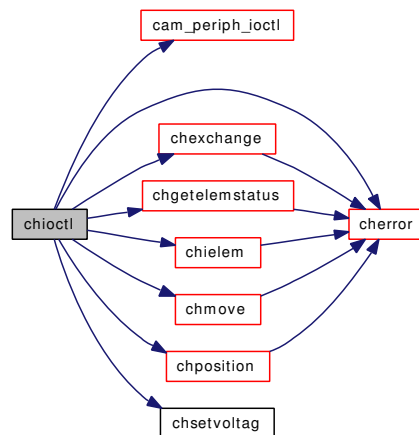


7.20.3.12 `static int chioctl (struct cdev * dev, u_long cmd, caddr_t addr, int flag, struct thread * td)`
 [static]

Definition at line 702 of file scsi_ch.c.

References CAM_DEBUG, CAM_DEBUG_TRACE, cam_periph_ioctl(), cherror(), chexchange(), chgetelemstatus(), chielem(), chmove(), chposition(), chsetvoltage(), cam_periph::path, ch_softc::sc_counts, ch_softc::sc_firsts, ch_softc::sc_picker, and cam_periph::softc.

Here is the call graph for this function:



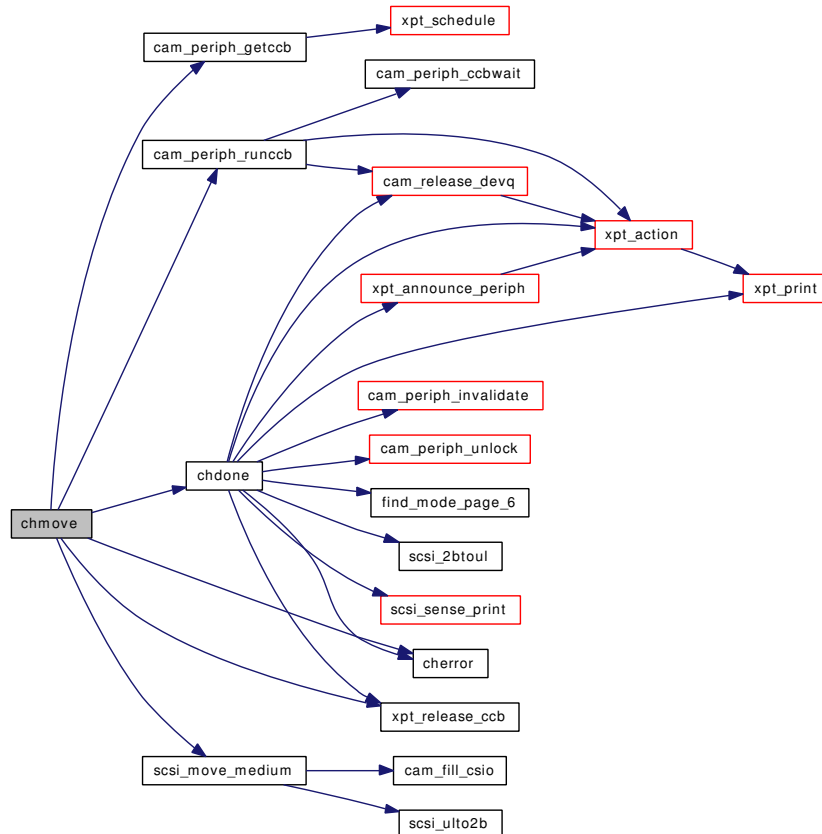
7.20.3.13 `static int chmove (struct cam_periph * periph, struct changer_move * cm)` [static]

Definition at line 801 of file scsi_ch.c.

References `cam_periph_getccb()`, `cam_periph_runccb()`, `CAM_RETRY_SELTO`, `CH_TIMEOUT_MOVE_MEDIUM`, `chdone()`, `cherror()`, `ccb::csio`, `ch_softc::device_stats`, `MSG_SIMPLE_Q_TAG`, `ch_softc::sc_counts`, `ch_softc::sc_firsts`, `ch_softc::sc_movemask`, `ch_softc::sc_picker`, `scsi_move_medium()`, `SF_RETRY_UA`, `cam_periph::softc`, `SSD_FULL_SIZE`, and `xpt_release_ccb()`.

Referenced by `chioctl()`.

Here is the call graph for this function:

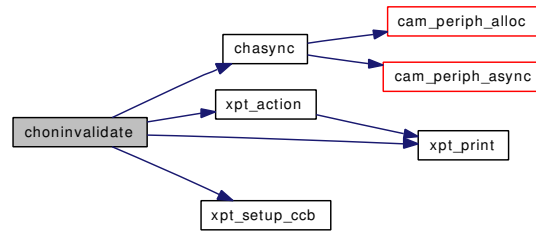


7.20.3.14 static void choninvalidate (struct `cam_periph` * *periph*) [static]

Definition at line 256 of file `scsi_ch.c`.

References `ccb_setasync::callback`, `ccb_setasync::callback_arg`, `ccb_setasync::ccb_h`, `CH_FLAG_INVALID`, `chasync()`, `ccb_setasync::event_enable`, `ccb_hdr::func_code`, `cam_periph::path`, `cam_periph::softc`, `xpt_action()`, `xpt_print()`, `XPT_SASYNC_CB`, and `xpt_setup_ccb()`.

Here is the call graph for this function:

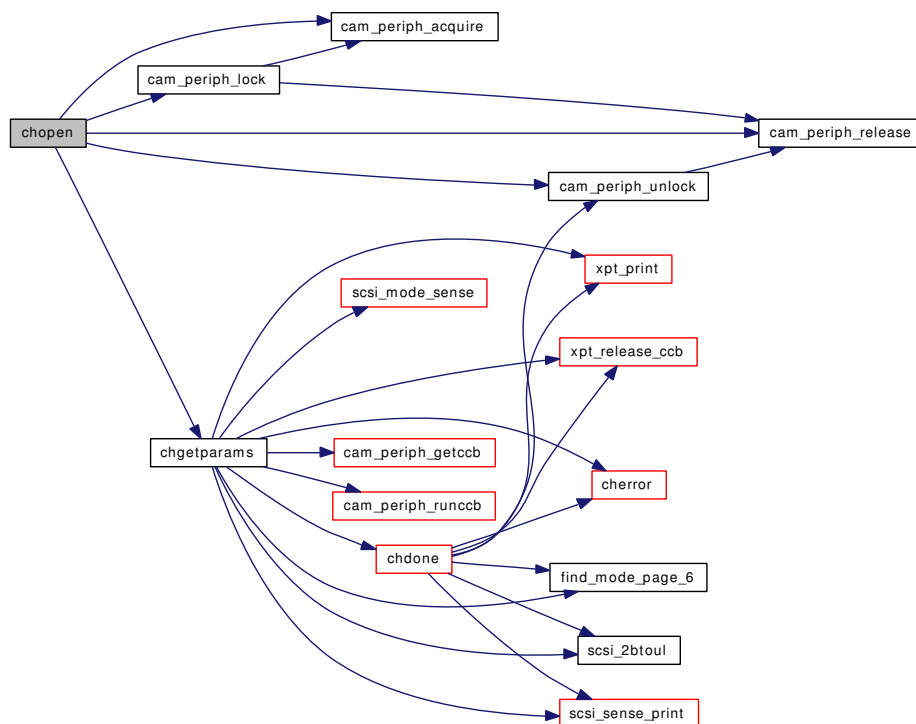


7.20.3.15 static int chopen (struct cdev * dev, int flags, int fmt, struct thread * td) [static]

Definition at line 406 of file scsi_ch.c.

References cam_periph_acquire(), cam_periph_lock(), cam_periph_release(), cam_periph_unlock(), CAM_REQ_CMP, CH_FLAG_INVALID, CH_FLAG_OPEN, chgetparams(), ch_softc::flags, and cam_periph::softc.

Here is the call graph for this function:



7.20.3.16 static int chposition (struct cam_periph * periph, struct changer_position * cp) [static]

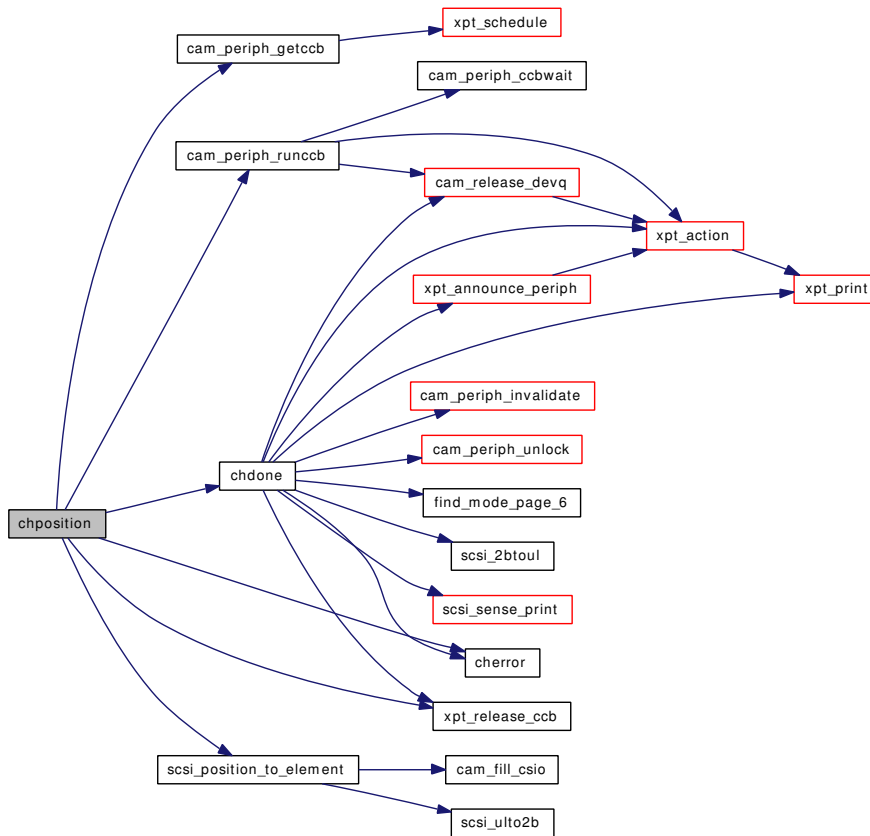
Definition at line 918 of file scsi_ch.c.

References cam_periph_getccb(), cam_periph_runccb(), CAM_RETRY_SELTO, CH_TIMEOUT_POSITION_TO_ELEMENT, chdone(), cherror(), ccb::csio, ch_softc::device_stats, MSG_SIMPLE_

Q_TAG, ch_softc::sc_counts, ch_softc::sc_firsts, ch_softc::sc_picker, scsi_position_to_element(), SF_RETRY-UA, cam_periph::softc, SSD_FULL_SIZE, and xpt_release_ccb().

Referenced by chioctl().

Here is the call graph for this function:

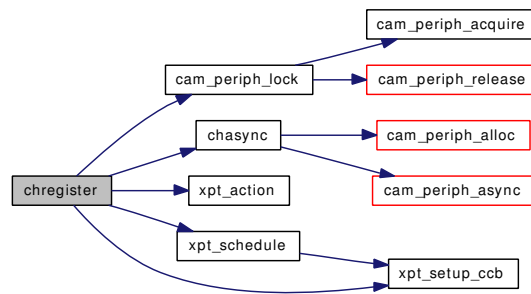


7.20.3.17 static `cam_status` `chregister` (`struct cam_periph *periph`, `void *arg`) [static]

Definition at line 338 of file `scsi_ch.c`.

References `AC_LOST_DEVICE`, `ccb_setasync::callback`, `ccb_setasync::callback_arg`, `cam_periph_unlock()`, `CAM_REQ_CMP`, `CAM_REQ_CMP_ERR`, `ccb_setasync::ccb_h`, `ch_cdevsw`, `CH_Q_NONE`, `CH_STATE_PROBE`, `chasync()`, `ccb_setasync::event_enable`, `ccb_hdr::func_code`, `ccb_getdev::inq_data`, `cam_periph::path`, `cam_periph::periph_name`, `SID_TYPE`, `cam_periph::softc`, `cam_periph::unit_number`, `xpt_action()`, `XPT_SASYNC_CB`, `xpt_schedule()`, and `xpt_setup_ccb()`.

Here is the call graph for this function:



7.20.3.18 `static int chsetvoltag (struct cam_periph * periph, struct changer_set_voltag_request * csvr)` [static]

Definition at line 1237 of file `scsi_ch.c`.

References `ch_softc::sc_counts`, `ch_softc::sc_firsts`, `SEND_VOLUME_TAG_ASSERT_ALTERNATE`, `SEND_VOLUME_TAG_REPLACE_ALTERNATE`, `SEND_VOLUME_TAG_UNDEFINED_ALTERNATE`, and `cam_periph::softc`.

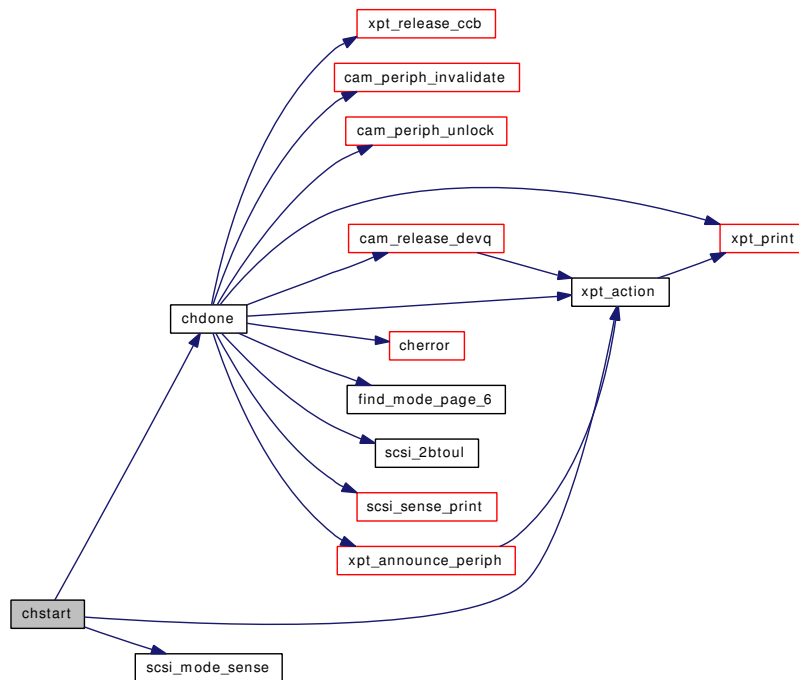
Referenced by `chioctl()`.

7.20.3.19 `static void chstart (struct cam_periph * periph, union ccb * start_ccb)` [static]

Definition at line 480 of file `scsi_ch.c`.

References `CAM_PRIORITY_NONE`, `ccb::ccb_h`, `CH_CCB_PROBE`, `CH_CCB_WAITING`, `CH_ELEMENT_ADDR_ASSIGN_PAGE`, `CH_Q_NO_DBD`, `CH_STATE_NORMAL`, `CH_STATE_PROBE`, `CH_TIMEOUT_MODE_SENSE`, `chdone()`, `ccb::csio`, `cam_periph::immediate_priority`, `MSG_SIMPLE_Q_TAG`, `cam_periph::pinfo`, `cam_pinfo::priority`, `ch_softc::quirks`, `scsi_mode_sense()`, `SMS_PAGE_CTRL_CURRENT`, `cam_periph::softc`, `SSD_FULL_SIZE`, `ch_softc::state`, and `xpt_action()`.

Here is the call graph for this function:



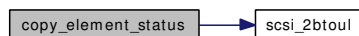
7.20.3.20 `static void copy_element_status (struct ch_softc * softc, u_int16_t flags, struct read_element_status_descriptor * desc, struct changer_element_status * ces)`
`[static]`

Definition at line 990 of file `scsi_ch.c`.

References `read_element_status_descriptor::eaddr`, `ch_softc::sc_counts`, `ch_softc::sc_firsts`, and `scsi_2btoul()`.

Referenced by `chgetelemstatus()`.

Here is the call graph for this function:



7.20.3.21 `static void copy_voltag (struct changer_voltag * uvoltag, struct volume_tag * voltag)`
`[static]`

Definition at line 971 of file `scsi_ch.c`.

References `c`, `scsi_2btoul()`, `volume_tag::vif`, and `volume_tag::vsn`.

Here is the call graph for this function:



7.20.3.22 PERIPHDRIIVER_DECLARE (ch, chdriver)

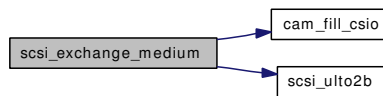
7.20.3.23 void **scsi_exchange_medium** (struct **ccb_scsiio** * *csio*, u_int32_t *retries*, void(*) (struct **cam_periph** *, union **ccb** *) *cbfcnp*, u_int8_t *tag_action*, u_int32_t *tea*, u_int32_t *src*, u_int32_t *dst1*, u_int32_t *dst2*, int *invert1*, int *invert2*, u_int8_t *sense_len*, u_int32_t *timeout*)

Definition at line 1528 of file `scsi_ch.c`.

References `CAM_DIR_NONE`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `EXCHANGE_MEDIUM`, `EXCHANGE_MEDIUM_INV1`, `EXCHANGE_MEDIUM_INV2`, and `scsi_ulto2b()`.

Referenced by `chexchange()`.

Here is the call graph for this function:



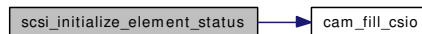
7.20.3.24 void **scsi_initialize_element_status** (struct **ccb_scsiio** * *csio*, u_int32_t *retries*, void(*) (struct **cam_periph** *, union **ccb** *) *cbfcnp*, u_int8_t *tag_action*, u_int8_t *sense_len*, u_int32_t *timeout*)

Definition at line 1630 of file `scsi_ch.c`.

References `CAM_DIR_NONE`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, and `INITIALIZE_ELEMENT_STATUS`.

Referenced by `chielem()`.

Here is the call graph for this function:



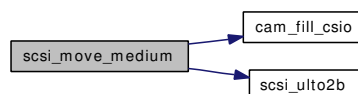
7.20.3.25 void **scsi_move_medium** (struct **ccb_scsiio** * *csio*, u_int32_t *retries*, void(*) (struct **cam_periph** *, union **ccb** *) *cbfcnp*, u_int8_t *tag_action*, u_int32_t *tea*, u_int32_t *src*, u_int32_t *dst*, int *invert*, u_int8_t *sense_len*, u_int32_t *timeout*)

Definition at line 1495 of file `scsi_ch.c`.

References `CAM_DIR_NONE`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `MOVE_MEDIUM`, `MOVE_MEDIUM_INVERT`, and `scsi_ulto2b()`.

Referenced by `chmove()`.

Here is the call graph for this function:



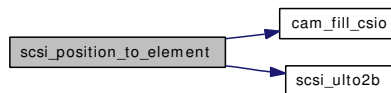
7.20.3.26 void `scsi_position_to_element` (struct `ccb_scsiio` * *csio*, u_int32_t *retries*, void(*) (struct `cam_periph` *, union `ccb` *) *cbfcnp*, u_int8_t *tag_action*, u_int32_t *tea*, u_int32_t *dst*, int *invert*, u_int8_t *sense_len*, u_int32_t *timeout*)

Definition at line 1565 of file `scsi_ch.c`.

References `CAM_DIR_NONE`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `POSITION_TO_ELEMENT`, `POSITION_TO_ELEMENT_INVERT`, and `scsi_ulto2b()`.

Referenced by `chposition()`.

Here is the call graph for this function:



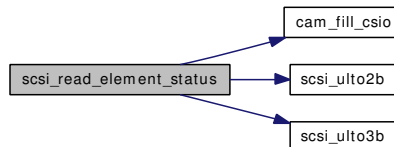
7.20.3.27 void `scsi_read_element_status` (struct `ccb_scsiio` * *csio*, u_int32_t *retries*, void(*) (struct `cam_periph` *, union `ccb` *) *cbfcnp*, u_int8_t *tag_action*, int *voltag*, u_int32_t *sea*, u_int32_t *count*, u_int8_t * *data_ptr*, u_int32_t *dxfer_len*, u_int8_t *sense_len*, u_int32_t *timeout*)

Definition at line 1596 of file `scsi_ch.c`.

References `CAM_DIR_IN`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `READ_ELEMENT_STATUS`, `READ_ELEMENT_STATUS_VOLTAG`, `scsi_ulto2b()`, and `scsi_ulto3b()`.

Referenced by `chgetelemstatus()`.

Here is the call graph for this function:

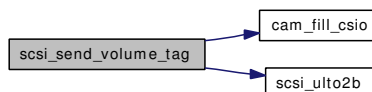


7.20.3.28 void `scsi_send_volume_tag` (struct `ccb_scsiio` * *csio*, u_int32_t *retries*, void(*) (struct `cam_periph` *, union `ccb` *) *cbfcnp*, u_int8_t *tag_action*, u_int16_t *element_address*, u_int8_t *send_action_code*, struct `scsi_send_volume_tag_parameters` * *parameters*, u_int8_t *sense_len*, u_int32_t *timeout*)

Definition at line 1656 of file `scsi_ch.c`.

References `CAM_DIR_OUT`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `scsi_ulto2b()`, and `SEND_VOLUME_TAG`.

Here is the call graph for this function:



7.20.4 Variable Documentation

7.20.4.1 struct cdevsw [ch_cdevsw](#) [static]

Initial value:

```
{
    .d_version =    D_VERSION,
    .d_flags =     D_NEEDGIANT,
    .d_open =      chopen,
    .d_close =     chclose,
    .d_ioctl =     chioctl,
    .d_name =      "ch",
}
```

Definition at line 214 of file `scsi_ch.c`.

Referenced by `chregister()`.

7.20.4.2 const u_int32_t [CH_TIMEOUT_EXCHANGE_MEDIUM = 100000](#) [static]

Definition at line 103 of file `scsi_ch.c`.

Referenced by `chexchange()`.

7.20.4.3 const u_int32_t [CH_TIMEOUT_INITIALIZE_ELEMENT_STATUS = 500000](#) [static]

Definition at line 107 of file `scsi_ch.c`.

Referenced by `chielem()`.

7.20.4.4 const u_int32_t [CH_TIMEOUT_MODE_SENSE = 6000](#) [static]

Definition at line 101 of file `scsi_ch.c`.

Referenced by `chgetparams()`, and `chstart()`.

7.20.4.5 const u_int32_t [CH_TIMEOUT_MOVE_MEDIUM = 100000](#) [static]

Definition at line 102 of file `scsi_ch.c`.

Referenced by `chmove()`.

7.20.4.6 const u_int32_t [CH_TIMEOUT_POSITION_TO_ELEMENT = 100000](#) [static]

Definition at line 104 of file `scsi_ch.c`.

Referenced by `chposition()`.

7.20.4.7 const u_int32_t [CH_TIMEOUT_READ_ELEMENT_STATUS = 10000](#) [static]

Definition at line 105 of file `scsi_ch.c`.

Referenced by `chgetelemstatus()`.

7.20.4.8 `const u_int32_t CH_TIMEOUT_SEND_VOLTAG = 10000` [static]

Definition at line 106 of file scsi_ch.c.

7.20.4.9 `periph_dtor_t chcleanup` [static]

Definition at line 184 of file scsi_ch.c.

Referenced by chasync().

7.20.4.10 `d_close_t chclose` [static]

Definition at line 179 of file scsi_ch.c.

7.20.4.11 `struct periph_driver chdriver` [static]**Initial value:**

```
{
    chinit, "ch",
    TAILQ_HEAD_INITIALIZER(chdriver.units), 0
}
```

Definition at line 206 of file scsi_ch.c.

7.20.4.12 `periph_init_t chinit` [static]

Definition at line 181 of file scsi_ch.c.

7.20.4.13 `d_ioctl_t chioctl` [static]

Definition at line 180 of file scsi_ch.c.

7.20.4.14 `periph_oninv_t choninvalidate` [static]

Definition at line 183 of file scsi_ch.c.

Referenced by chasync().

7.20.4.15 `d_open_t chopen` [static]

Definition at line 178 of file scsi_ch.c.

7.20.4.16 `periph_ctor_t chregister` [static]

Definition at line 182 of file scsi_ch.c.

Referenced by chasync().

7.20.4.17 `periph_start_t chstart` `[static]`

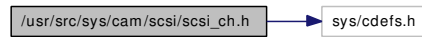
Definition at line 185 of file `scsi_ch.c`.

Referenced by `chasync()`.

7.21 /usr/src/sys/cam/scsi/scsi_ch.h File Reference

```
#include <sys/cdefs.h>
```

Include dependency graph for scsi_ch.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- struct [scsi_exchange_medium](#)
- struct [scsi_initialize_element_status](#)
- struct [scsi_move_medium](#)
- struct [scsi_position_to_element](#)
- struct [scsi_read_element_status](#)
- struct [scsi_request_volume_element_address](#)
- struct [read_element_status_header](#)
- struct [read_element_status_page_header](#)
- struct [volume_tag](#)
- struct [read_element_status_descriptor](#)
- struct [scsi_send_volume_tag](#)
- struct [scsi_send_volume_tag_parameters](#)
- struct [page_device_capabilities](#)
- struct [page_element_address_assignment](#)
- struct [page_transport_geometry_parameters](#)

Defines

- #define [_SCSI_SCSI_CH_H](#) 1
- #define [EXCHANGE_MEDIUM](#) 0xa6
- #define [EXCHANGE_MEDIUM_INV1](#) 0x01
- #define [EXCHANGE_MEDIUM_INV2](#) 0x02
- #define [INITIALIZE_ELEMENT_STATUS](#) 0x07
- #define [MOVE_MEDIUM](#) 0xa5
- #define [MOVE_MEDIUM_INVERT](#) 0x01
- #define [POSITION_TO_ELEMENT](#) 0x2b
- #define [POSITION_TO_ELEMENT_INVERT](#) 0x01
- #define [READ_ELEMENT_STATUS](#) 0xb8
- #define [READ_ELEMENT_STATUS_VOLTAG](#) 0x10
- #define [REQUEST_VOLUME_ELEMENT_ADDRESS](#) 0xb5
- #define [REQUEST_VOLUME_ELEMENT_ADDRESS_VOLTAG](#) 0x10
- #define [CH_ELEMENT_ADDR_ASSIGN_PAGE](#) 0x1D
- #define [CH_TRANS_GEOM_PARAMS_PAGE](#) 0x1E

- #define CH_DEVICE_CAP_PAGE 0x1F
- #define READ_ELEMENT_STATUS_AVOLTAG 0x40
- #define READ_ELEMENT_STATUS_PVOLTAG 0x80
- #define READ_ELEMENT_STATUS_FULL 0x01
- #define READ_ELEMENT_STATUS_IMPEXP 0x02
- #define READ_ELEMENT_STATUS_EXCEPT 0x04
- #define READ_ELEMENT_STATUS_ACCESS 0x08
- #define READ_ELEMENT_STATUS_EXENAB 0x10
- #define READ_ELEMENT_STATUS_INENAB 0x20
- #define READ_ELEMENT_STATUS_MT_MASK1 0x05
- #define READ_ELEMENT_STATUS_ST_MASK1 0x0c
- #define READ_ELEMENT_STATUS_IE_MASK1 0x3f
- #define READ_ELEMENT_STATUS_DT_MASK1 0x0c
- #define READ_ELEMENT_STATUS_DT_LUNMASK 0x07
- #define READ_ELEMENT_STATUS_DT_LUVALID 0x10
- #define READ_ELEMENT_STATUS_DT_IDVALID 0x20
- #define READ_ELEMENT_STATUS_DT_NOTBUS 0x80
- #define READ_ELEMENT_STATUS_INVERT 0x40
- #define READ_ELEMENT_STATUS_SVALID 0x80
- #define ELEMENT_TYPE_MASK 0x0f
- #define ELEMENT_TYPE_ALL 0x00
- #define ELEMENT_TYPE_MT 0x01
- #define ELEMENT_TYPE_ST 0x02
- #define ELEMENT_TYPE_IE 0x03
- #define ELEMENT_TYPE_DT 0x04
- #define PGCODE_MASK 0x3f
- #define PGCODE_PS 0x80
- #define SEND_VOLUME_TAG 0xb6
- #define SEND_VOLUME_TAG_ASSERT_PRIMARY 0x08
- #define SEND_VOLUME_TAG_ASSERT_ALTERNATE 0x09
- #define SEND_VOLUME_TAG_REPLACE_PRIMARY 0x0a
- #define SEND_VOLUME_TAG_REPLACE_ALTERNATE 0x0b
- #define SEND_VOLUME_TAG_UNDEFINED_PRIMARY 0x0c
- #define SEND_VOLUME_TAG_UNDEFINED_ALTERNATE 0x0d
- #define STOR_MT 0x01
- #define STOR_ST 0x02
- #define STOR_IE 0x04
- #define STOR_DT 0x08
- #define MOVE_TO_MT 0x01
- #define MOVE_TO_ST 0x02
- #define MOVE_TO_IE 0x04
- #define MOVE_TO_DT 0x08
- #define EXCHANGE_WITH_MT 0x01
- #define EXCHANGE_WITH_ST 0x02
- #define EXCHANGE_WITH_IE 0x04
- #define EXCHANGE_WITH_DT 0x08
- #define CAN_ROTATE 0x01

Functions

- `__BEGIN_DECLS` void `scsi_move_medium` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, u_int32_t tea, u_int32_t src, u_int32_t dst, int invert, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_exchange_medium` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, u_int32_t tea, u_int32_t src, u_int32_t dst1, u_int32_t dst2, int invert1, int invert2, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_position_to_element` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, u_int32_t tea, u_int32_t dst, int invert, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_read_element_status` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, int voltag, u_int32_t sea, u_int32_t count, u_int8_t *data_ptr, u_int32_t dxfer_len, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_initialize_element_status` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_send_volume_tag` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, u_int16_t element_address, u_int8_t send_action_code, struct `scsi_send_volume_tag_parameters` *parameters, u_int8_t sense_len, u_int32_t timeout)

7.21.1 Define Documentation

7.21.1.1 #define SCSI SCSI_CH_H 1

Definition at line 64 of file `scsi_ch.h`.

7.21.1.2 #define CAN_ROTATE 0x01

Definition at line 436 of file `scsi_ch.h`.

7.21.1.3 #define CH_DEVICE_CAP_PAGE 0x1F

Definition at line 173 of file `scsi_ch.h`.

Referenced by `chgetparams()`.

7.21.1.4 #define CH_ELEMENT_ADDR_ASSIGN_PAGE 0x1D

Definition at line 171 of file `scsi_ch.h`.

Referenced by `chgetparams()`, and `chstart()`.

7.21.1.5 #define CH_TRANS_GEOM_PARAMS_PAGE 0x1E

Definition at line 172 of file `scsi_ch.h`.

7.21.1.6 #define ELEMENT_TYPE_ALL 0x00

Definition at line 256 of file `scsi_ch.h`.

7.21.1.7 #define ELEMENT_TYPE_DT 0x04

Definition at line 260 of file scsi_ch.h.

7.21.1.8 #define ELEMENT_TYPE_IE 0x03

Definition at line 259 of file scsi_ch.h.

7.21.1.9 #define ELEMENT_TYPE_MASK 0x0f

Definition at line 255 of file scsi_ch.h.

7.21.1.10 #define ELEMENT_TYPE_MT 0x01

Definition at line 257 of file scsi_ch.h.

7.21.1.11 #define ELEMENT_TYPE_ST 0x02

Definition at line 258 of file scsi_ch.h.

7.21.1.12 #define EXCHANGE_MEDIUM 0xa6

Definition at line 78 of file scsi_ch.h.

Referenced by scsi_exchange_medium().

7.21.1.13 #define EXCHANGE_MEDIUM_INV1 0x01

Definition at line 85 of file scsi_ch.h.

Referenced by scsi_exchange_medium().

7.21.1.14 #define EXCHANGE_MEDIUM_INV2 0x02

Definition at line 86 of file scsi_ch.h.

Referenced by scsi_exchange_medium().

7.21.1.15 #define EXCHANGE_WITH_DT 0x08

Definition at line 357 of file scsi_ch.h.

7.21.1.16 #define EXCHANGE_WITH_IE 0x04

Definition at line 356 of file scsi_ch.h.

7.21.1.17 #define EXCHANGE_WITH_MT 0x01

Definition at line 354 of file scsi_ch.h.

7.21.1.18 #define EXCHANGE_WITH_ST 0x02

Definition at line 355 of file scsi_ch.h.

7.21.1.19 #define INITIALIZE_ELEMENT_STATUS 0x07

Definition at line 96 of file scsi_ch.h.

Referenced by scsi_initialize_element_status().

7.21.1.20 #define MOVE_MEDIUM 0xa5

Definition at line 108 of file scsi_ch.h.

7.21.1.21 #define MOVE_MEDIUM_INVERT 0x01

Definition at line 115 of file scsi_ch.h.

Referenced by scsi_move_medium().

7.21.1.22 #define MOVE_TO_DT 0x08

Definition at line 346 of file scsi_ch.h.

7.21.1.23 #define MOVE_TO_IE 0x04

Definition at line 345 of file scsi_ch.h.

7.21.1.24 #define MOVE_TO_MT 0x01

Definition at line 343 of file scsi_ch.h.

7.21.1.25 #define MOVE_TO_ST 0x02

Definition at line 344 of file scsi_ch.h.

7.21.1.26 #define PGCODE_MASK 0x3f

Definition at line 265 of file scsi_ch.h.

7.21.1.27 #define PGCODE_PS 0x80

Definition at line 266 of file scsi_ch.h.

7.21.1.28 #define POSITION_TO_ELEMENT 0x2b

Definition at line 125 of file scsi_ch.h.

7.21.1.29 #define POSITION_TO_ELEMENT_INVERT 0x01

Definition at line 131 of file scsi_ch.h.

Referenced by scsi_position_to_element().

7.21.1.30 #define READ_ELEMENT_STATUS 0xb8

Definition at line 140 of file scsi_ch.h.

7.21.1.31 #define READ_ELEMENT_STATUS_ACCESS 0x08

Definition at line 213 of file scsi_ch.h.

7.21.1.32 #define READ_ELEMENT_STATUS_AVOLTAG 0x40

Definition at line 189 of file scsi_ch.h.

7.21.1.33 #define READ_ELEMENT_STATUS_DT_IDVALID 0x20

Definition at line 234 of file scsi_ch.h.

7.21.1.34 #define READ_ELEMENT_STATUS_DT_LUNMASK 0x07

Definition at line 232 of file scsi_ch.h.

7.21.1.35 #define READ_ELEMENT_STATUS_DT_LUVALID 0x10

Definition at line 233 of file scsi_ch.h.

7.21.1.36 #define READ_ELEMENT_STATUS_DT_MASK1 0x0c

Definition at line 220 of file scsi_ch.h.

7.21.1.37 #define READ_ELEMENT_STATUS_DT_NOTBUS 0x80

Definition at line 235 of file scsi_ch.h.

7.21.1.38 #define READ_ELEMENT_STATUS_EXCEPT 0x04

Definition at line 212 of file scsi_ch.h.

7.21.1.39 #define READ_ELEMENT_STATUS_EXENAB 0x10

Definition at line 214 of file scsi_ch.h.

7.21.1.40 #define READ_ELEMENT_STATUS_FULL 0x01

Definition at line 210 of file scsi_ch.h.

7.21.1.41 #define READ_ELEMENT_STATUS_IE_MASK1 0x3f

Definition at line 219 of file scsi_ch.h.

7.21.1.42 #define READ_ELEMENT_STATUS_IMPEXP 0x02

Definition at line 211 of file scsi_ch.h.

7.21.1.43 #define READ_ELEMENT_STATUS_INENAB 0x20

Definition at line 215 of file scsi_ch.h.

7.21.1.44 #define READ_ELEMENT_STATUS_INVERT 0x40

Definition at line 242 of file scsi_ch.h.

7.21.1.45 #define READ_ELEMENT_STATUS_MT_MASK1 0x05

Definition at line 217 of file scsi_ch.h.

7.21.1.46 #define READ_ELEMENT_STATUS_PVOLTAG 0x80

Definition at line 190 of file scsi_ch.h.

7.21.1.47 #define READ_ELEMENT_STATUS_ST_MASK1 0x0c

Definition at line 218 of file scsi_ch.h.

7.21.1.48 #define READ_ELEMENT_STATUS_SVALID 0x80

Definition at line 243 of file scsi_ch.h.

7.21.1.49 #define READ_ELEMENT_STATUS_VOLTAG 0x10

Definition at line 142 of file scsi_ch.h.

Referenced by scsi_read_element_status().

7.21.1.50 #define REQUEST_VOLUME_ELEMENT_ADDRESS 0xb5

Definition at line 154 of file scsi_ch.h.

7.21.1.51 #define REQUEST_VOLUME_ELEMENT_ADDRESS_VOLTAG 0x10

Definition at line 156 of file scsi_ch.h.

7.21.1.52 #define SEND_VOLUME_TAG 0xb6

Definition at line 274 of file scsi_ch.h.

Referenced by scsi_send_volume_tag().

7.21.1.53 #define SEND_VOLUME_TAG_ASSERT_ALTERNATE 0x09

Definition at line 281 of file scsi_ch.h.

Referenced by chsetvoltag().

7.21.1.54 #define SEND_VOLUME_TAG_ASSERT_PRIMARY 0x08

Definition at line 280 of file scsi_ch.h.

7.21.1.55 #define SEND_VOLUME_TAG_REPLACE_ALTERNATE 0x0b

Definition at line 283 of file scsi_ch.h.

Referenced by chsetvoltag().

7.21.1.56 #define SEND_VOLUME_TAG_REPLACE_PRIMARY 0x0a

Definition at line 282 of file scsi_ch.h.

7.21.1.57 #define SEND_VOLUME_TAG_UNDEFINED_ALTERNATE 0x0d

Definition at line 285 of file scsi_ch.h.

Referenced by chsetvoltag().

7.21.1.58 #define SEND_VOLUME_TAG_UNDEFINED_PRIMARY 0x0c

Definition at line 284 of file scsi_ch.h.

7.21.1.59 #define STOR_DT 0x08

Definition at line 331 of file scsi_ch.h.

7.21.1.60 #define STOR_IE 0x04

Definition at line 330 of file scsi_ch.h.

7.21.1.61 #define STOR_MT 0x01

Definition at line 328 of file scsi_ch.h.

7.21.1.62 #define STOR_ST 0x02

Definition at line 329 of file scsi_ch.h.

7.21.2 Function Documentation

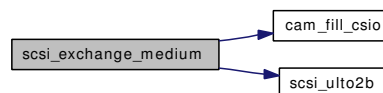
7.21.2.1 void [scsi_exchange_medium](#) (struct [ccb_scsiio](#) * *csio*, u_int32_t *retries*, void(*) (struct [cam_periph](#) *, union [ccb](#) *) *cbfcnp*, u_int8_t *tag_action*, u_int32_t *tea*, u_int32_t *src*, u_int32_t *dst1*, u_int32_t *dst2*, int *invert1*, int *invert2*, u_int8_t *sense_len*, u_int32_t *timeout*)

Definition at line 1528 of file scsi_ch.c.

References [CAM_DIR_NONE](#), [cam_fill_csio\(\)](#), [cdb_t::cdb_bytes](#), [ccb_scsiio::cdb_io](#), [EXCHANGE_MEDIUM](#), [EXCHANGE_MEDIUM_INV1](#), [EXCHANGE_MEDIUM_INV2](#), and [scsi_ulto2b\(\)](#).

Referenced by [chexchange\(\)](#).

Here is the call graph for this function:



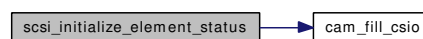
7.21.2.2 void [scsi_initialize_element_status](#) (struct [ccb_scsiio](#) * *csio*, u_int32_t *retries*, void(*) (struct [cam_periph](#) *, union [ccb](#) *) *cbfcnp*, u_int8_t *tag_action*, u_int8_t *sense_len*, u_int32_t *timeout*)

Definition at line 1630 of file scsi_ch.c.

References [CAM_DIR_NONE](#), [cam_fill_csio\(\)](#), [cdb_t::cdb_bytes](#), [ccb_scsiio::cdb_io](#), and [INITIALIZE_ELEMENT_STATUS](#).

Referenced by [chielem\(\)](#).

Here is the call graph for this function:



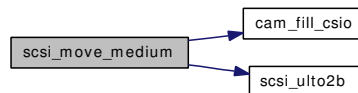
7.21.2.3 `__BEGIN_DECLS void scsi_move_medium (struct ccb_scsiio * csio, u_int32_t retries, void(*)(struct cam_periph *, union ccb *) cbfcnp, u_int8_t tag_action, u_int32_t tea, u_int32_t src, u_int32_t dst, int invert, u_int8_t sense_len, u_int32_t timeout)`

Definition at line 1495 of file scsi_ch.c.

References CAM_DIR_NONE, cam_fill_csio(), cdb_t::cdb_bytes, ccb_scsiio::cdb_io, MOVE_MEDIUM, MOVE_MEDIUM_INVERT, and scsi_ulto2b().

Referenced by chmove().

Here is the call graph for this function:



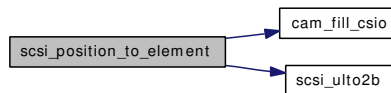
7.21.2.4 `void scsi_position_to_element (struct ccb_scsiio * csio, u_int32_t retries, void(*)(struct cam_periph *, union ccb *) cbfcnp, u_int8_t tag_action, u_int32_t tea, u_int32_t dst, int invert, u_int8_t sense_len, u_int32_t timeout)`

Definition at line 1565 of file scsi_ch.c.

References CAM_DIR_NONE, cam_fill_csio(), cdb_t::cdb_bytes, ccb_scsiio::cdb_io, POSITION_TO_ELEMENT, POSITION_TO_ELEMENT_INVERT, and scsi_ulto2b().

Referenced by chposition().

Here is the call graph for this function:



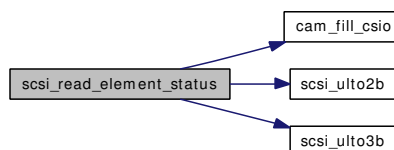
7.21.2.5 `void scsi_read_element_status (struct ccb_scsiio * csio, u_int32_t retries, void(*)(struct cam_periph *, union ccb *) cbfcnp, u_int8_t tag_action, int voltag, u_int32_t sea, u_int32_t count, u_int8_t * data_ptr, u_int32_t dxfer_len, u_int8_t sense_len, u_int32_t timeout)`

Definition at line 1596 of file scsi_ch.c.

References CAM_DIR_IN, cam_fill_csio(), cdb_t::cdb_bytes, ccb_scsiio::cdb_io, READ_ELEMENT_STATUS, READ_ELEMENT_STATUS_VOLTAG, scsi_ulto2b(), and scsi_ulto3b().

Referenced by chgetelemstatus().

Here is the call graph for this function:

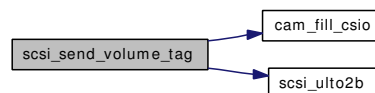


7.21.2.6 void **scsi_send_volume_tag** (struct **ccb_scsiio** * *csio*, u_int32_t *retries*, void(*) (struct **cam_periph** *, union **ccb** *) *cbfcnp*, u_int8_t *tag_action*, u_int16_t *element_address*, u_int8_t *send_action_code*, struct **scsi_send_volume_tag_parameters** * *parameters*, u_int8_t *sense_len*, u_int32_t *timeout*)

Definition at line 1656 of file `scsi_ch.c`.

References `CAM_DIR_OUT`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `scsi_ulito2b()`, and `SEND_VOLUME_TAG`.

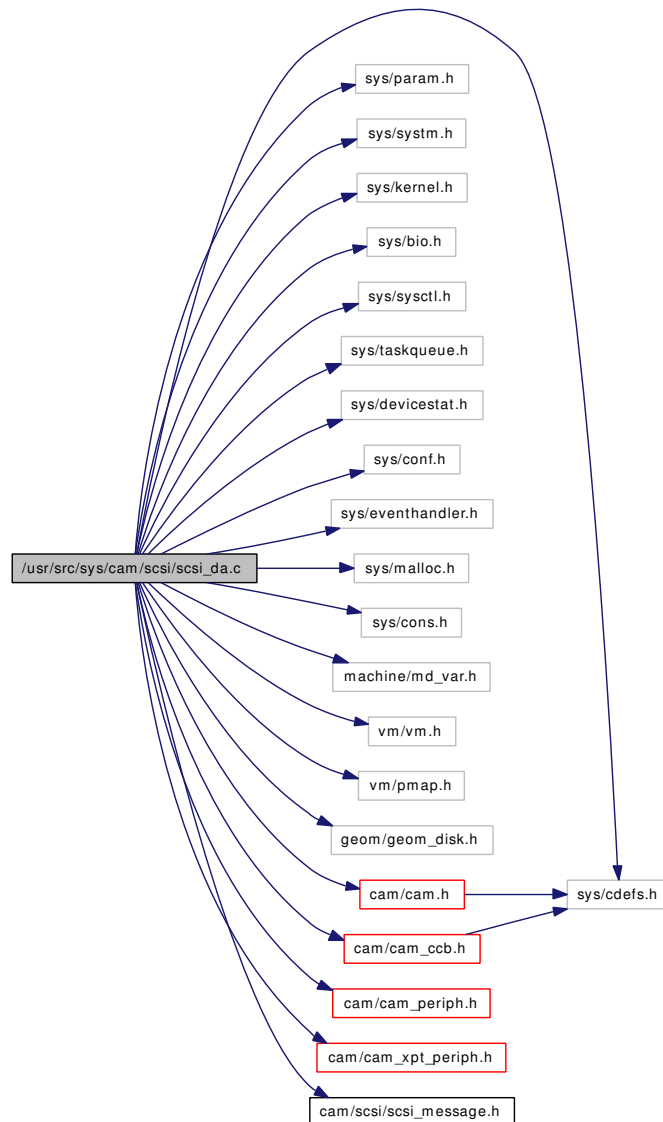
Here is the call graph for this function:



7.22 /usr/src/sys/cam/scsi/scsi_da.c File Reference

```
#include <sys/cdefs.h>
#include <sys/param.h>
#include <sys/system.h>
#include <sys/kernel.h>
#include <sys/bio.h>
#include <sys/sysctl.h>
#include <sys/taskqueue.h>
#include <sys/devicestat.h>
#include <sys/conf.h>
#include <sys/eventhandler.h>
#include <sys/malloc.h>
#include <sys/cons.h>
#include <machine/md_var.h>
#include <vm/vm.h>
#include <vm/pmap.h>
#include <geom/geom_disk.h>
#include <cam/cam.h>
#include <cam/cam_ccb.h>
#include <cam/cam_periph.h>
#include <cam/cam_xpt_periph.h>
#include <cam/scsi/scsi_message.h>
```

Include dependency graph for scsi_da.c:



Data Structures

- struct [disk_params](#)
- struct [da_softc](#)
- struct [da_quirk_entry](#)

Defines

- #define [ccb_state](#) ppriv_field0
- #define [ccb_bp](#) ppriv_ptr1
- #define [DA_DEFAULT_TIMEOUT](#) 60
- #define [DA_DEFAULT_RETRY](#) 4
- #define [DA_DEFAULT_SEND_ORDERED](#) 1
- #define [DA_ORDEREDTAG_INTERVAL](#) 4

Enumerations

- enum `da_state` { `DA_STATE_PROBE`, `DA_STATE_PROBE2`, `DA_STATE_NORMAL` }
- enum `da_flags` {
`DA_FLAG_PACK_INVALID` = 0x001, `DA_FLAG_NEW_PACK` = 0x002, `DA_FLAG_PACK_LOCKED` = 0x004, `DA_FLAG_PACK_REMOVABLE` = 0x008,
`DA_FLAG_TAGGED_QUEUEING` = 0x010, `DA_FLAG_NEED_OTAG` = 0x020, `DA_FLAG_WENT_IDLE` = 0x040, `DA_FLAG_RETRY_UA` = 0x080,
`DA_FLAG_OPEN` = 0x100, `DA_FLAG_SCTX_INIT` = 0x200 }
- enum `da_quirks` { `DA_Q_NONE` = 0x00, `DA_Q_NO_SYNC_CACHE` = 0x01, `DA_Q_NO_6_BYTE` = 0x02, `DA_Q_NO_PREVENT` = 0x04 }
- enum `da_ccb_state` {
`DA_CCB_PROBE` = 0x01, `DA_CCB_PROBE2` = 0x02, `DA_CCB_BUFFER_IO` = 0x03, `DA_CCB_WAITING` = 0x04,
`DA_CCB_DUMP` = 0x05, `DA_CCB_TYPE_MASK` = 0x0F, `DA_CCB_RETRY_UA` = 0x10 }

Functions

- `__FBSDDID` ("FreeBSD: src/sys/cam/scsi/scsi_da.c,v 1.203 2007/02/21 07:45:01 n_hibma Exp \$")
- static void `dasync` (void *callback_arg, u_int32_t code, struct `cam_path` *path, void *arg)
- static void `dasysctlnit` (void *context, int pending)
- static int `dacmdsizesysctl` (SYSCTL_HANDLER_ARGS)
- static void `dadone` (struct `cam_periph` *periph, union `ccb` *done_ccb)
- static int `daerror` (union `ccb` *ccb, u_int32_t `cam_flags`, u_int32_t `sense_flags`)
- static void `daprevent` (struct `cam_periph` *periph, int action)
- static int `dagetcapacity` (struct `cam_periph` *periph)
- static void `dasetgeom` (struct `cam_periph` *periph, uint32_t `block_len`, uint64_t `maxsector`)
- static void `dashutdown` (void *arg, int howto)
- `SYSCTL_NODE` (`_kern_cam`, `OID_AUTO`, `da`, `CTLFLAG_RD`, 0, "CAM Direct Access Disk driver")
- `SYSCTL_INT` (`_kern_cam_da`, `OID_AUTO`, `retry_count`, `CTLFLAG_RW`, &`da_retry_count`, 0, "Normal I/O retry count")
- `TUNABLE_INT` ("kern.cam.da.retry_count",&`da_retry_count`)
- `SYSCTL_INT` (`_kern_cam_da`, `OID_AUTO`, `default_timeout`, `CTLFLAG_RW`, &`da_default_timeout`, 0, "Normal I/O timeout (in seconds)")
- `TUNABLE_INT` ("kern.cam.da.default_timeout",&`da_default_timeout`)
- `SYSCTL_INT` (`_kern_cam_da`, `OID_AUTO`, `da_send_ordered`, `CTLFLAG_RW`, &`da_send_ordered`, 0, "Send Ordered Tags")
- `TUNABLE_INT` ("kern.cam.da.da_send_ordered",&`da_send_ordered`)
- `PERIPHDRIVER_DECLARE` (`da`, `dadriver`)
- static `SLIST_HEAD` (`da_softc`)
- static int `daclose` (struct `disk` *dp)
- static void `dastrategy` (struct `bio` *bp)
- static int `dadump` (void *arg, void *virtual, vm_offset_t `physical`, off_t `offset`, size_t `length`)
- static void `dainit` (void)
- static void `daoninvalidate` (struct `cam_periph` *periph)
- static void `dacleanup` (struct `cam_periph` *periph)
- static `cam_status` `daregister` (struct `cam_periph` *periph, void *arg)
- static void `dastart` (struct `cam_periph` *periph, union `ccb` *start_ccb)
- static int `cmd6workaround` (union `ccb` *ccb)
- static void `dasendorderedtag` (void *arg)

Variables

- static const char `quantum` [] = "QUANTUM"
- static const char `microp` [] = "MICROP"
- static struct `da_quirk_entry` `da_quirk_table` []
- static `disk_strategy_t` `dastrategy`
- static `dumper_t` `dadump`
- static `periph_init_t` `dainit`
- static `periph_ctor_t` `daregister`
- static `periph_dtor_t` `dacleanup`
- static `periph_start_t` `dastart`
- static `periph_oninv_t` `daoninvalidate`
- static `timeout_t` `dasendorderedtag`
- static int `da_retry_count` = DA_DEFAULT_RETRY
- static int `da_default_timeout` = DA_DEFAULT_TIMEOUT
- static int `da_send_ordered` = DA_DEFAULT_SEND_ORDERED
- static struct `periph_driver` `dadriver`

7.22.1 Define Documentation

7.22.1.1 #define ccb_bp ppriv_ptr1

Definition at line 110 of file `scsi_da.c`.

7.22.1.2 #define ccb_state ppriv_field0

Definition at line 109 of file `scsi_da.c`.

7.22.1.3 #define DA_DEFAULT_RETRY 4

Definition at line 506 of file `scsi_da.c`.

7.22.1.4 #define DA_DEFAULT_SEND_ORDERED 1

Definition at line 510 of file `scsi_da.c`.

7.22.1.5 #define DA_DEFAULT_TIMEOUT 60

Definition at line 502 of file `scsi_da.c`.

Referenced by `dadump()`, and `dainit()`.

7.22.1.6 #define DA_ORDEREDTAG_INTERVAL 4

Definition at line 543 of file `scsi_da.c`.

Referenced by `dainit()`, and `dasendorderedtag()`.

7.22.2 Enumeration Type Documentation

7.22.2.1 enum [da_ccb_state](#)

Enumerator:

DA_CCB_PROBE
DA_CCB_PROBE2
DA_CCB_BUFFER_IO
DA_CCB_WAITING
DA_CCB_DUMP
DA_CCB_TYPE_MASK
DA_CCB_RETRY_UA

Definition at line 98 of file `scsi_da.c`.

7.22.2.2 enum [da_flags](#)

Enumerator:

DA_FLAG_PACK_INVALID
DA_FLAG_NEW_PACK
DA_FLAG_PACK_LOCKED
DA_FLAG_PACK_REMOVABLE
DA_FLAG_TAGGED_QUEUEING
DA_FLAG_NEED_OTAG
DA_FLAG_WENT_IDLE
DA_FLAG_RETRY_UA
DA_FLAG_OPEN
DA_FLAG_SCTX_INIT

Definition at line 78 of file `scsi_da.c`.

7.22.2.3 enum [da_quirks](#)

Enumerator:

DA_Q_NONE
DA_Q_NO_SYNC_CACHE
DA_Q_NO_6_BYTE
DA_Q_NO_PREVENT

Definition at line 91 of file `scsi_da.c`.

7.22.2.4 enum `da_state`

Enumerator:

`DA_STATE_PROBE`
`DA_STATE_PROBE2`
`DA_STATE_NORMAL`

Definition at line 72 of file `scsi_da.c`.

7.22.3 Function Documentation

7.22.3.1 `__FBSDID("$FreeBSD: src/sys/cam/scsi/scsi_da.c, v 1.203 2007/02/21 07:45:01 n_hibma Exp $")`

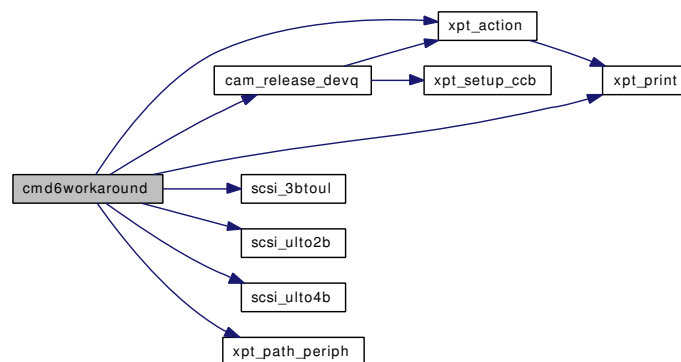
7.22.3.2 `static int cmd6workaround (union ccb * ccb)` `[static]`

Definition at line 1393 of file `scsi_da.c`.

References `scsi_rw_10::addr`, `scsi_rw_6::addr`, `scsi_rw_10::byte2`, `CAM_CDB_POINTER`, `CAM_DEV_QFRZN`, `cam_release_devq()`, `CAM_QUEUE_REQ`, `ccb::ccb_h`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `ccb_scsiio::cdb_len`, `scsi_rw_6::control`, `scsi_rw_10::control`, `ccb::csio`, `ccb_hdr::flags`, `scsi_rw_10::length`, `scsi_rw_6::length`, `scsi_rw_6::opcode`, `scsi_rw_10::opcode`, `ccb_hdr::path`, `READ_10`, `READ_6`, `scsi_rw_10::reserved`, `scsi_3btoul()`, `scsi_ulto2b()`, `scsi_ulto4b()`, `cam_periph::softc`, `ccb_hdr::status`, `WRITE_10`, `WRITE_6`, `xpt_action()`, `xpt_path_periph()`, and `xpt_print()`.

Referenced by `daerror()`.

Here is the call graph for this function:



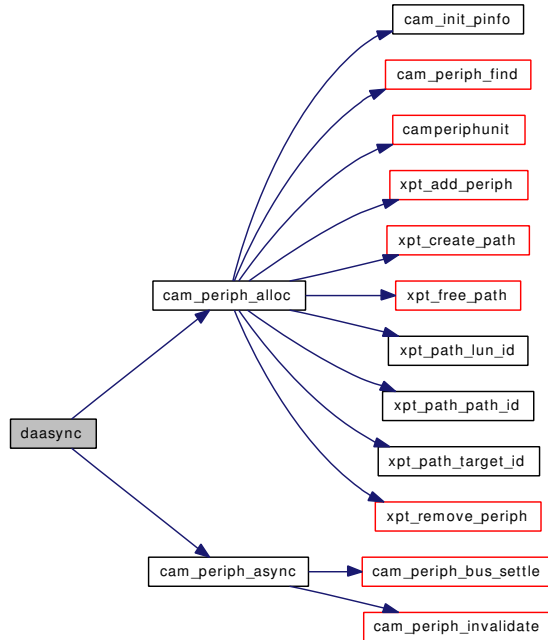
7.22.3.3 `static void daasync (void * callback_arg, u_int32_t code, struct cam_path * path, void * arg)` `[static]`

Definition at line 956 of file `scsi_da.c`.

References `AC_BUS_RESET`, `AC_FOUND_DEVICE`, `AC_SENT_BDR`, `cam_periph_alloc()`, `cam_periph_async()`, `CAM_PERIPH_BIO`, `CAM_REQ_CMP`, `CAM_REQ_INPROG`, `ccb_getdev::ccb_h`, `DA_CCB_RETRY_UA`, `DA_FLAG_RETRY_UA`, `dacleanup`, `daoninvalidate`, `daregister`, `dastart`, `ccb_getdev::inq_data`, `ccb_hdr::path`, `SID_TYPE`, `cam_periph::softc`, `T_DIRECT`, `T_OPTICAL`, and `T_RBC`.

Referenced by `dainit()`, `daoninvalidate()`, and `daregister()`.

Here is the call graph for this function:

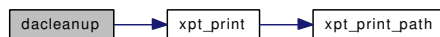


7.22.3.4 `static void dacleanup (struct cam_periph * periph)` [static]

Definition at line 937 of file `scsi_da.c`.

References `DA_FLAG_SCTX_INIT`, `cam_periph::path`, `cam_periph::softc`, and `xpt_print()`.

Here is the call graph for this function:



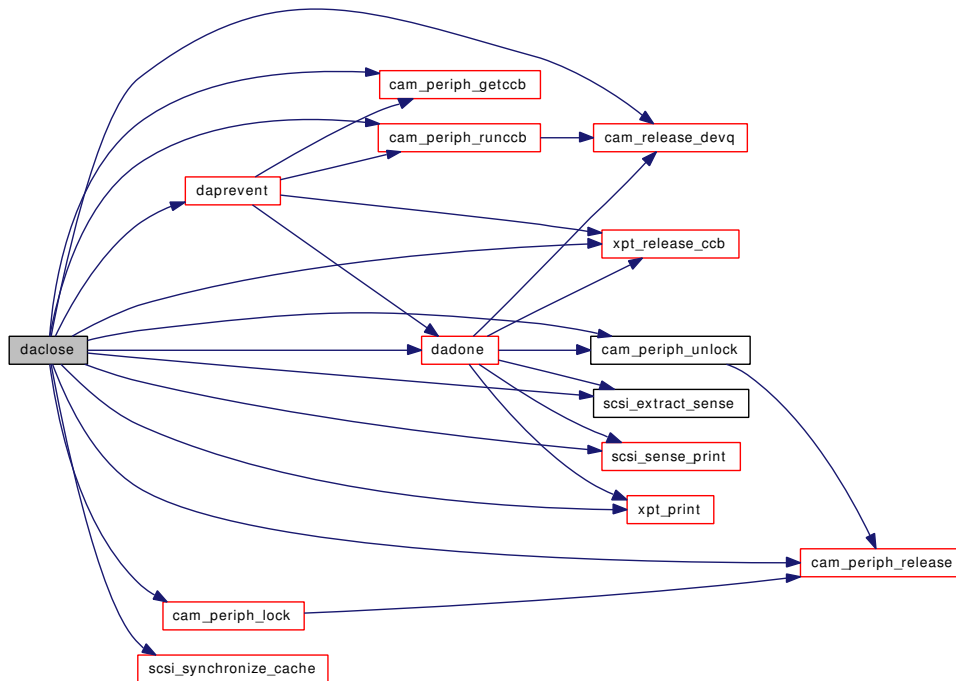
7.22.3.5 `static int daclose (struct disk * dp)` [static]

Definition at line 618 of file `scsi_da.c`.

References `CAM_DEV_QFRZN`, `cam_periph_getccb()`, `cam_periph_lock()`, `cam_periph_release()`, `cam_periph_runccb()`, `cam_periph_unlock()`, `cam_release_devq()`, `CAM_REQ_CMP`, `CAM_SCSI_STATUS_ERROR`, `CAM_STATUS_MASK`, `ccb_scsiio::ccb_h`, `ccb::ccb_h`, `ccb::csio`, `DA_FLAG_OPEN`, `DA_FLAG_PACK_REMOVABLE`, `DA_Q_NO_PREVENT`, `DA_Q_NO_SYNC_CACHE`, `dadone()`, `daprevent()`, `MSG_SIMPLE_Q_TAG`, `ccb_hdr::path`, `cam_periph::path`, `PR_ALLOW`, `scsi_extract_sense()`, `scsi_sense_print()`, `ccb_scsiio::scsi_status`, `scsi_synchronize_cache()`, `ccb_scsiio::sense_data`, `SF_RETRY_UA`, `cam_periph::softc`, `SSD_FULL_SIZE`, `SSD_KEY_ILLEGAL_REQUEST`, `ccb_hdr::status`, `xpt_print()`, and `xpt_release_ccb()`.

Referenced by `daregister()`.

Here is the call graph for this function:



7.22.3.6 static int dacmdsizesysctl (SYSCTL_HANDLER_ARGS) [static]

Definition at line 1057 of file scsi_da.c.

Referenced by dasysctlinit().

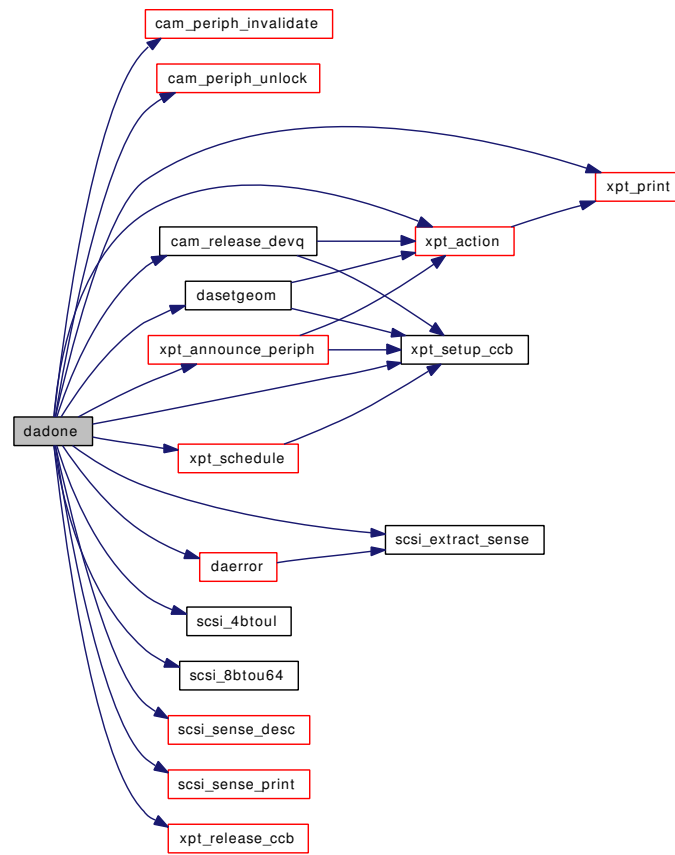
7.22.3.7 static void dadone (struct cam_periph * periph, union ccb * done_ccb) [static]

Definition at line 1438 of file scsi_da.c.

References `scsi_read_capacity_data_long::addr`, `scsi_read_capacity_data::addr`, `da_softc::bio_queue`, `CAM_AUTOSNS_VALID`, `CAM_DEV_QFRZN`, `cam_periph_invalidate()`, `cam_periph_unlock()`, `cam_release_devq()`, `CAM_REQ_CMP`, `CAM_RETRY_SELTO`, `CAM_SENSE_PHYS`, `CAM_SENSE_PTR`, `CAM_STATUS_MASK`, `ccb_hdr::cbfcnp`, `ccb_getdev::ccb_h`, `ccb::ccb_h`, `ccb_scsiio::ccb_h`, `ccb::csio`, `disk_params::cylinders`, `DA_CCB_BUFFER_IO`, `DA_CCB_DUMP`, `DA_CCB_PROBE`, `DA_CCB_PROBE2`, `DA_CCB_RETRY_UA`, `DA_CCB_TYPE_MASK`, `DA_CCB_WAITING`, `DA_FLAG_PACK_INVALID`, `DA_FLAG_WENT_IDLE`, `DA_STATE_NORMAL`, `DA_STATE_PROBE`, `DA_STATE_PROBE2`, `daerror()`, `dasetgeom()`, `ccb_scsiio::data_ptr`, `scsi_sense_data::error_code`, `ccb_hdr::flags`, `ccb_hdr::func_code`, `disk_params::heads`, `scsi_read_capacity_data_long::length`, `scsi_read_capacity_data::length`, `ccb_hdr::path`, `cam_periph::path`, `ccb_scsiio::resid`, `scsi_4btoul()`, `scsi_8btoul64()`, `scsi_extract_sense()`, `scsi_sense_desc()`, `scsi_sense_print()`, `disk_params::secs_per_track`, `disk_params::secsize`, `disk_params::sectors`, `ccb_scsiio::sense_data`, `SF_NO_PRINT`, `SF_RETRY_UA`, `cam_periph::softc`, `SSD_CURRENT_ERROR`, `ccb_hdr::status`, `xpt_action()`, `xpt_announce_periph()`, `XPT_GDEV_TYPE`, `xpt_print()`, `xpt_release_ccb()`, `xpt_schedule()`, and `xpt_setup_ccb()`.

Referenced by `daclose()`, `dadump()`, `dagetcapacity()`, `daprevent()`, `dashutdown()`, and `dastart()`.

Here is the call graph for this function:

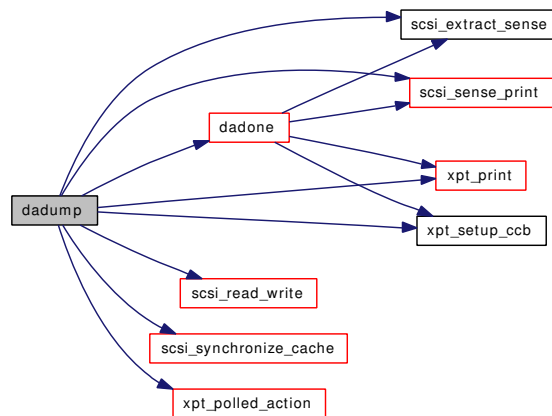


7.22.3.8 static int dadump (void * arg, void * virtual, vm_offset_t physical, off_t offset, size_t length) [static]

Definition at line 757 of file scsi_da.c.

References CAM_REQ_CMP, CAM SCSI_STATUS_ERROR, CAM_STATUS_MASK, ccb::ccb_h, ccb_scsiio::ccb_h, DA_CCB_DUMP, DA_DEFAULT_TIMEOUT, DA_FLAG_PACK_INVALID, DA_Q_NO_SYNC_CACHE, dadone(), MSG_ORDERED_Q_TAG, MSG_SIMPLE_Q_TAG, cam_periph::path, scsi_extract_sense(), scsi_read_write(), scsi_sense_print(), scsi_synchronize_cache(), cam_periph::softc, SSD_FULL_SIZE, SSD_KEY_ILLEGAL_REQUEST, ccb_hdr::status, xpt_polled_action(), xpt_print(), and xpt_setup_ccb().

Here is the call graph for this function:



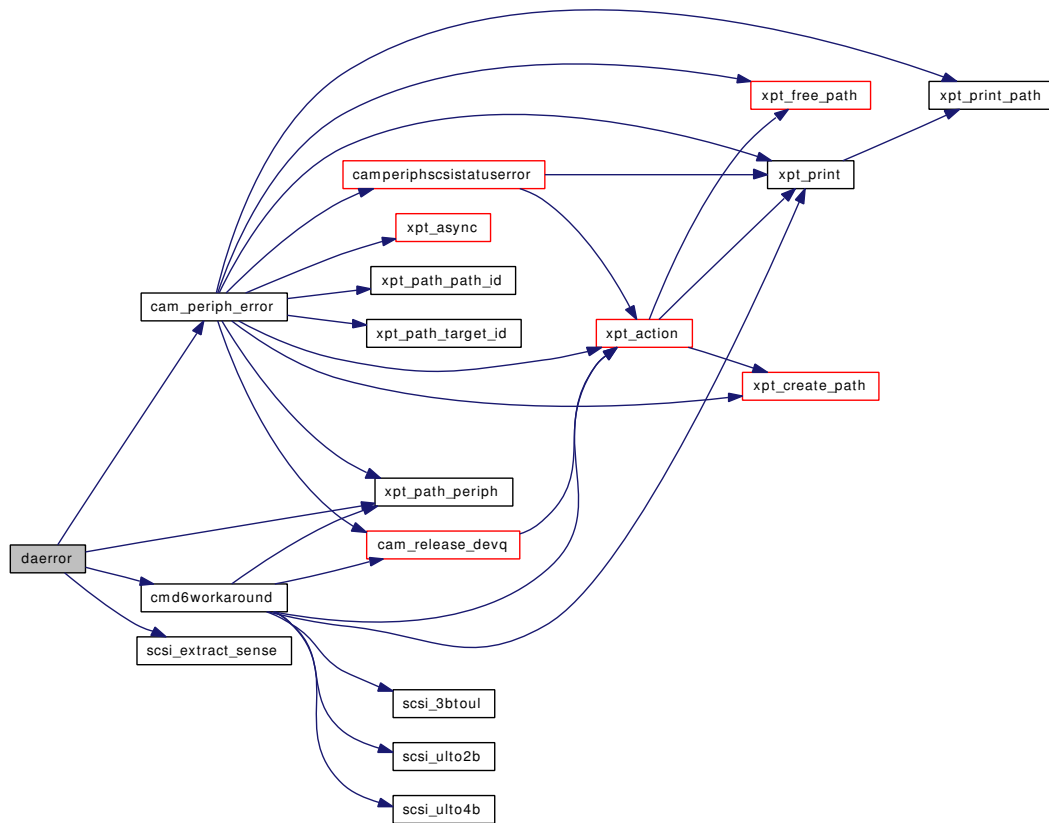
7.22.3.9 static int daerror (union **ccb** * *ccb*, u_int32_t *cam_flags*, u_int32_t *sense_flags*) [static]

Definition at line 1732 of file scsi_da.c.

References CAM_AUTOSNS_VALID, cam_periph_error(), CAM_REQ_INVALID, CAM_SCSI_STATUS_ERROR, CAM_SENSE_PHYS, CAM_SENSE_PTR, CAM_STATUS_MASK, ccb::ccb_h, cmd6workaround(), ccb::csio, ccb_hdr::flags, ccb_hdr::path, scsi_extract_sense(), ccb_scsiio::scsi_status, SCSI_STATUS_CHECK_COND, ccb_scsiio::sense_data, SF_RETRY_UA, cam_periph::softc, SSD_KEY_ILLEGAL_REQUEST, ccb_hdr::status, and xpt_path_periph().

Referenced by dadone(), and dagetcapacity().

Here is the call graph for this function:



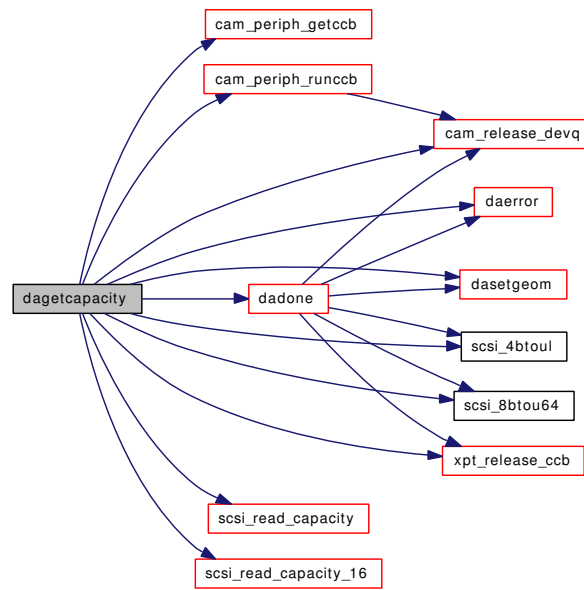
7.22.3.10 static int dagetcapacity (struct [cam_periph](#) * *periph*) [static]

Definition at line 1814 of file `scsi_da.c`.

References `scsi_read_capacity_data::addr`, `CAM_DEV_QFRZN`, `cam_periph_getccb()`, `cam_periph_runccb()`, `cam_release_devq()`, `CAM_RETRY_SELTO`, `ccb::ccb_h`, `ccb::csio`, `DA_FLAG_PACK_REMOVABLE`, `dadone()`, `daerror()`, `dasetgeom()`, `scsi_read_capacity_data::length`, `MSG_SIMPLE_QTAG`, `ccb_hdr::path`, `scsi_4btoul()`, `scsi_8btou64()`, `scsi_read_capacity()`, `scsi_read_capacity_16()`, `SF_NO_PRINT`, `SF_RETRY_UA`, `cam_periph::softc`, `SSD_FULL_SIZE`, `ccb_hdr::status`, and `xpt_release_ccb()`.

Referenced by `SLIST_HEAD()`.

Here is the call graph for this function:

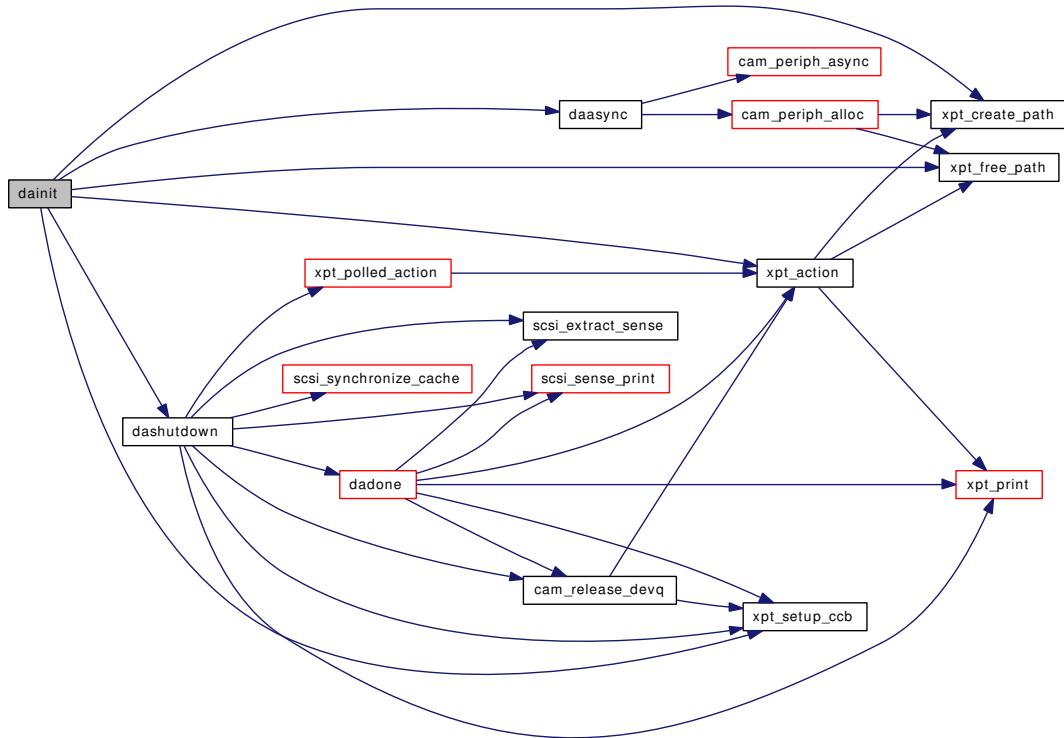


7.22.3.11 static void dainit (void) [static]

Definition at line 847 of file scsi_da.c.

References AC_FOUND_DEVICE, ccb_setasync::callback, ccb_setasync::callback_arg, CAM_LUN_WILDCARD, CAM_REQ_CMP, CAM_TARGET_WILDCARD, CAM_XPT_PATH_ID, ccb::ccb_h, ccb_setasync::ccb_h, DA_DEFAULT_TIMEOUT, DA_ORDEREDTAG_INTERVAL, da_send_ordered, daasync(), dasendorderedtag, dashutdown(), ccb_setasync::event_enable, ccb_hdr::func_code, ccb_hdr::status, xpt_action(), xpt_create_path(), xpt_free_path(), XPT_SASYNC_CB, and xpt_setup_ccb().

Here is the call graph for this function:

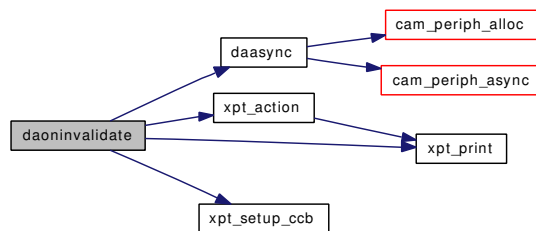


7.22.3.12 static void daoinvalidate (struct **cam_periph** * *periph*) [static]

Definition at line 894 of file scsi_da.c.

References ccb_setasync::callback, ccb_setasync::callback_arg, ccb_setasync::ccb_h, DA_FLAG_PACK_INVALID, daasync(), ccb_setasync::event_enable, ccb_hdr::func_code, cam_periph::path, cam_periph::softc, xpt_action(), xpt_print(), XPT_SASYNC_CB, and xpt_setup_ccb().

Here is the call graph for this function:



7.22.3.13 static void daprevent (struct **cam_periph** * *periph*, int *action*) [static]

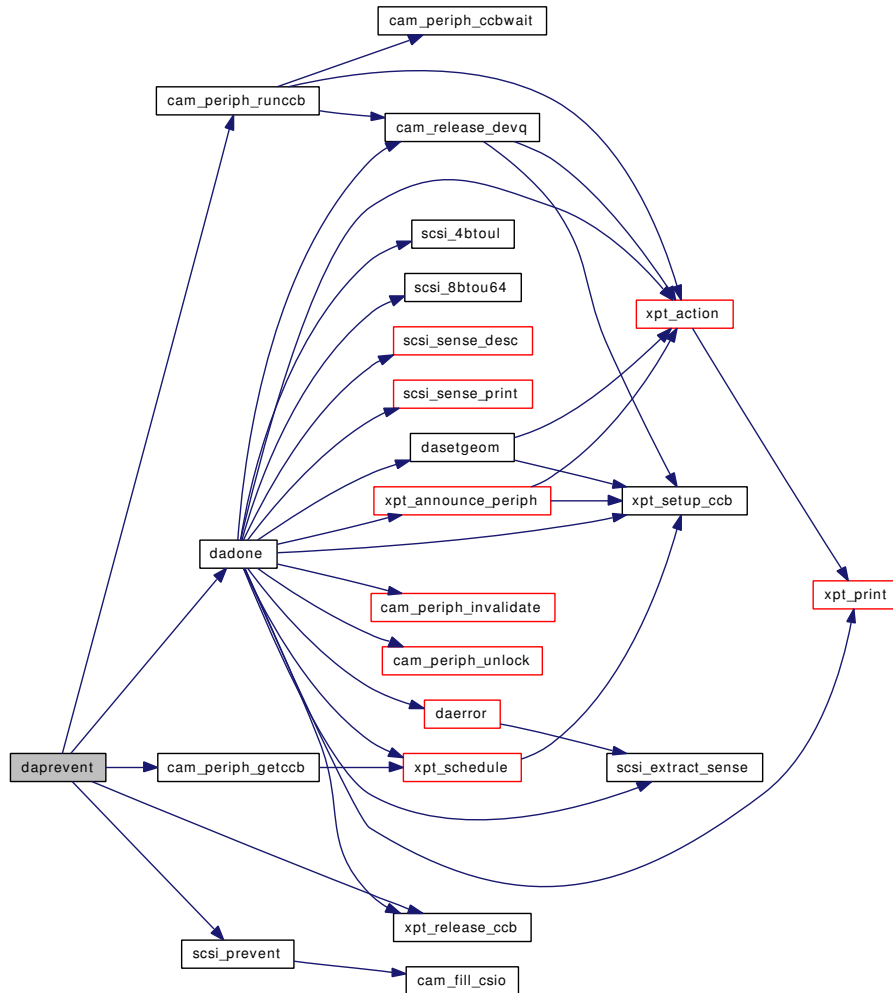
Definition at line 1775 of file scsi_da.c.

References cam_periph_getccb(), cam_periph_runccb(), CAM_RETRY_SELTO, ccb::csio, DA_FLAG_PACK_LOCKED, dadone(), MSG_SIMPLE_Q_TAG, PR_ALLOW, PR_PREVENT, scsi_prevent(), SF_

RETRY_UA, cam_periph::softc, SSD_FULL_SIZE, and xpt_release_ccb().

Referenced by daclose(), and SLIST_HEAD().

Here is the call graph for this function:

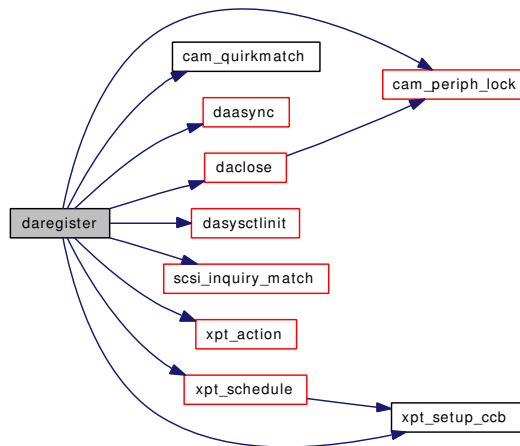


7.22.3.14 static `cam_status` `daregister (struct cam_periph * periph, void * arg)` [static]

Definition at line 1089 of file `scsi_da.c`.

References `AC_BUS_RESET`, `AC_LOST_DEVICE`, `AC_SENT_BDR`, `ccb_setasync::callback`, `ccb_setasync::callback_arg`, `cam_periph_lock()`, `cam_quirkmatch()`, `CAM_REQ_CMP`, `CAM_REQ_CMP_ERR`, `ccb_setasync::ccb_h`, `ccb::ccb_h`, `ccb_pathinq::ccb_h`, `DA_FLAG_PACK_REMOVABLE`, `DA_FLAG_TAGGED_QUEUEING`, `DA_Q_NO_6_BYTE`, `DA_Q_NO_SYNC_CACHE`, `DA_Q_NONE`, `da_quirk_table`, `DA_STATE_PROBE`, `daasync()`, `daclose()`, `dadump`, `dastrategy`, `dasysctlimit()`, `ccb_setasync::event_enable`, `scsi_inquiry_data::flags`, `ccb_hdr::func_code`, `ccb_getdev::inq_data`, `cam_periph::path`, `PIM_NO_6_BYTE`, `scsi_inquiry_match()`, `SID_CmdQue`, `SID_IS_REMOVABLE`, `SID_TYPE`, `cam_periph::softc`, `ccb_hdr::status`, `T_RBC`, `cam_periph::unit_number`, `xpt_action()`, `XPT_PATH_INQ`, `XPT_SASYNC_CB`, `xpt_schedule()`, and `xpt_setup_ccb()`.

Here is the call graph for this function:



7.22.3.15 static void dasendorderedtag (void * arg) [static]

Definition at line 1960 of file scsi_da.c.

References da_default_timeout, DA_FLAG_NEED_OTAG, DA_FLAG_WENT_IDLE, DA_ORDEREDTAG_INTERVAL, da_send_ordered, and dasendorderedtag.

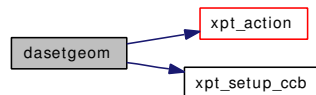
7.22.3.16 static void dasetgeom (struct cam_periph * periph, uint32_t block_len, uint64_t maxsector) [static]

Definition at line 1913 of file scsi_da.c.

References ccb_calc_geometry::block_size, CAM_REQ_CMP, CAM_STATUS_MASK, ccb::ccb_h, ccb_calc_geometry::ccb_h, disk_params::cylinders, ccb_calc_geometry::cylinders, ccb_hdr::func_code, disk_params::heads, ccb_calc_geometry::heads, cam_periph::path, disk_params::secs_per_track, ccb_calc_geometry::secs_per_track, disk_params::secsize, disk_params::sectors, cam_periph::softc, ccb_hdr::status, ccb_calc_geometry::volume_size, xpt_action(), XPT_CALC_GEOMETRY, and xpt_setup_ccb().

Referenced by dadone(), and dagetcapacity().

Here is the call graph for this function:



7.22.3.17 static void dashutdown (void * arg, int howto) [static]

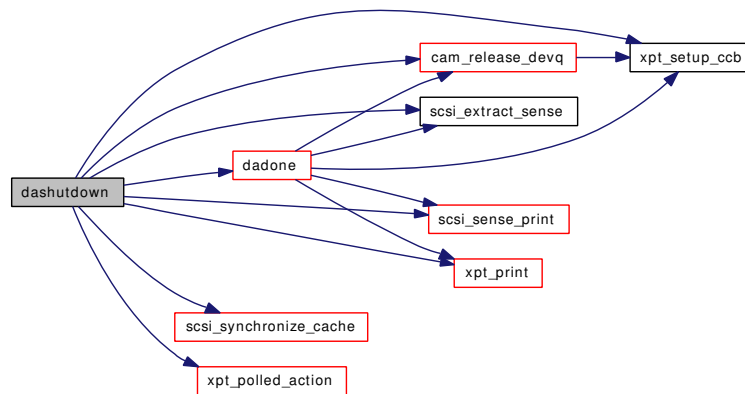
Definition at line 1990 of file scsi_da.c.

References CAM_DEV_QFRZN, cam_release_devq(), CAM_REQ_CMP, CAM_SCSI_STATUS_ERROR, CAM_STATUS_MASK, ccb::ccb_h, ccb::csio, DA_CCB_DUMP, DA_FLAG_OPEN, DA_Q_NO_SYNC_CACHE, dadone(), dadriver, MSG_SIMPLE_Q_TAG, ccb_hdr::path, cam_periph::path,

scsi_extract_sense(), scsi_sense_print(), ccb_scsiio::scsi_status, SCSI_STATUS_CHECK_COND, scsi_synchronize_cache(), ccb_scsiio::sense_data, cam_periph::softc, SSD_FULL_SIZE, SSD_KEY_ILLEGAL_REQUEST, ccb_hdr::status, xpt_polled_action(), xpt_print(), and xpt_setup_ccb().

Referenced by dainit().

Here is the call graph for this function:

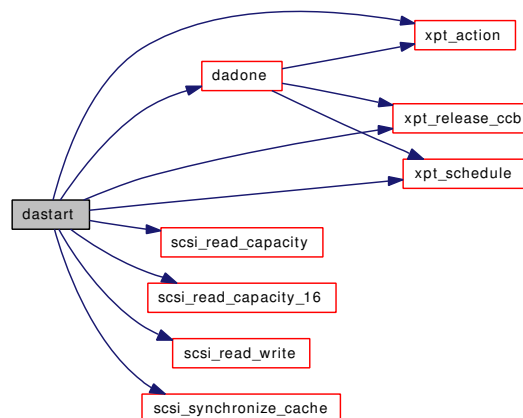


7.22.3.18 static void dastart (struct cam_periph * periph, union ccb * start_ccb) [static]

Definition at line 1231 of file scsi_da.c.

References da_softc::bio_queue, CAM_DEBUG_PRINT, CAM_DEBUG_SUBTRACE, CAM_PRIORITY_NONE, ccb::ccb_h, ccb::csio, DA_CCB_BUFFER_IO, DA_CCB_PROBE, DA_CCB_PROBE2, DA_CCB_RETRY_UA, DA_CCB_WAITING, da_default_timeout, DA_FLAG_NEED_OTAG, DA_FLAG_RETRY_UA, da_retry_count, DA_STATE_NORMAL, DA_STATE_PROBE, DA_STATE_PROBE2, dadone(), cam_periph::immediate_priority, MSG_ORDERED_Q_TAG, MSG_SIMPLE_Q_TAG, cam_periph::pinfo, cam_pinfo::priority, scsi_read_capacity(), scsi_read_capacity_16(), scsi_read_write(), scsi_synchronize_cache(), cam_periph::softc, SSD_FULL_SIZE, xpt_action(), xpt_release_ccb(), and xpt_schedule().

Here is the call graph for this function:

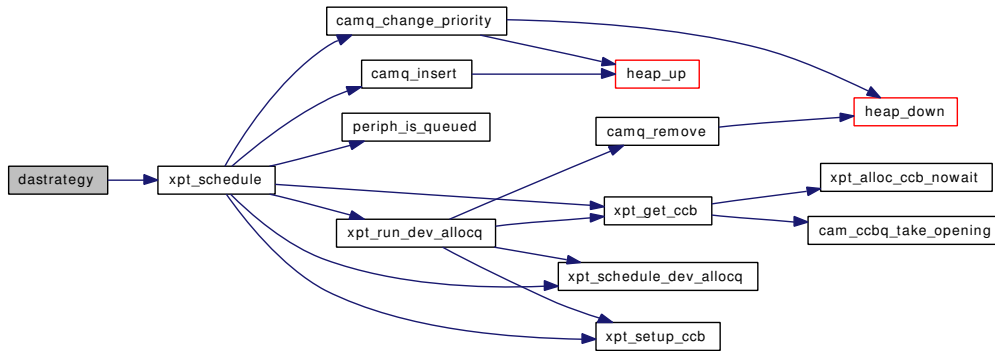


7.22.3.19 static void dastrategy (struct bio * bp) [static]

Definition at line 706 of file scsi_da.c.

References da_softc::bio_queue, DA_FLAG_PACK_INVALID, cam_periph::softc, and xpt_schedule().

Here is the call graph for this function:



7.22.3.20 static void dasysctlinit (void * context, int pending) [static]

Definition at line 1020 of file scsi_da.c.

References DA_FLAG_SCTX_INIT, dacmdsizesysctl(), cam_periph::softc, and cam_periph::unit_number.

Referenced by daregister().

Here is the call graph for this function:



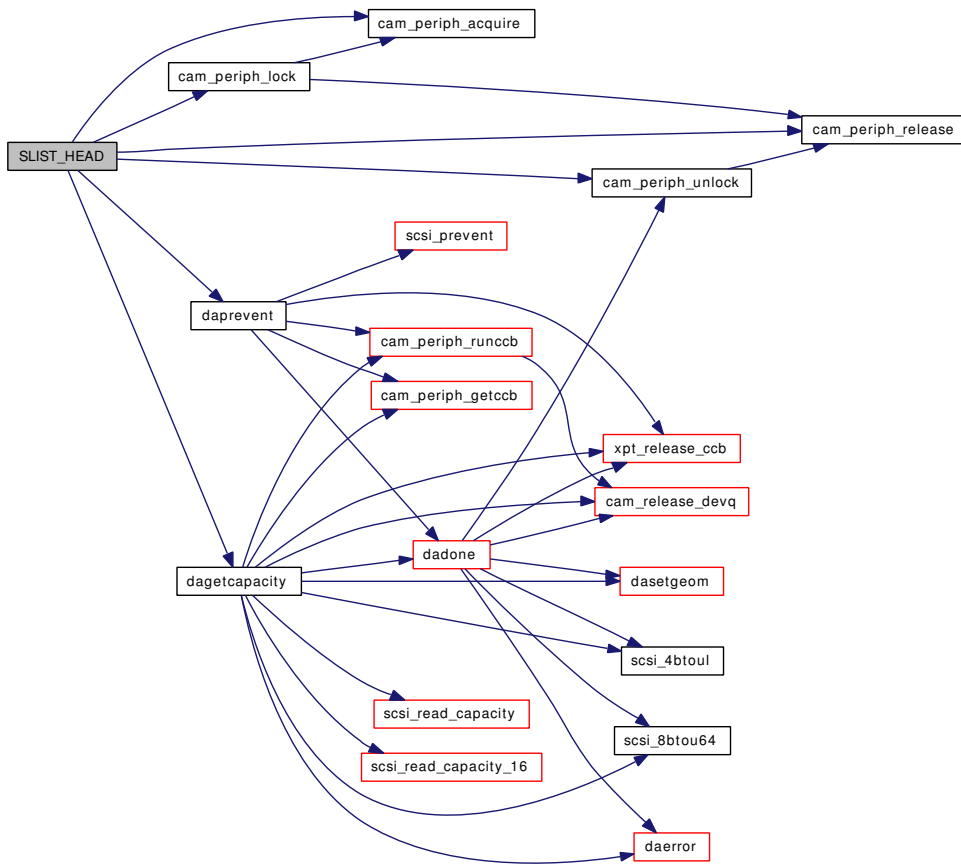
7.22.3.21 PERIPHDRIVER_DECLARE (da, dadriver)

7.22.3.22 static SLIST_HEAD (da_softc) [static]

Definition at line 554 of file scsi_da.c.

References CAM_DEBUG, CAM_DEBUG_TRACE, cam_periph_acquire(), cam_periph_lock(), cam_periph_release(), cam_periph_unlock(), CAM_REQ_CMP, DA_FLAG_OPEN, DA_FLAG_PACK_INVALID, DA_FLAG_PACK_REMOVABLE, DA_Q_NO_PREVENT, dagetcapacity(), daprevent(), cam_periph::path, PR_PREVENT, cam_periph::softc, and cam_periph::unit_number.

Here is the call graph for this function:



- 7.22.3.23 `SYSCTL_INT (_kern_cam_da, OID_AUTO, da_send_ordered, CTLFLAG_RW, & da_send_ordered, 0, "Send Ordered Tags")`
- 7.22.3.24 `SYSCTL_INT (_kern_cam_da, OID_AUTO, default_timeout, CTLFLAG_RW, & da_default_timeout, 0, "Normal I/O timeout (in seconds)")`
- 7.22.3.25 `SYSCTL_INT (_kern_cam_da, OID_AUTO, retry_count, CTLFLAG_RW, & da_retry_count, 0, "Normal I/O retry count")`
- 7.22.3.26 `SYSCTL_NODE (_kern_cam, OID_AUTO, da, CTLFLAG_RD, 0, "CAM Direct Access Disk driver")`
- 7.22.3.27 `TUNABLE_INT ("kern.cam.da.da_send_ordered", & da_send_ordered)`
- 7.22.3.28 `TUNABLE_INT ("kern.cam.da.default_timeout", & da_default_timeout)`
- 7.22.3.29 `TUNABLE_INT ("kern.cam.da.retry_count", & da_retry_count)`

7.22.4 Variable Documentation

7.22.4.1 `int da_default_timeout = DA_DEFAULT_TIMEOUT [static]`

Definition at line 515 of file `scsi_da.c`.

Referenced by `dasendorderedtag()`, and `dastart()`.

7.22.4.2 `struct da_quirk_entry da_quirk_table[] [static]`

Definition at line 146 of file `scsi_da.c`.

Referenced by `daregister()`.

7.22.4.3 `int da_retry_count = DA_DEFAULT_RETRY [static]`

Definition at line 514 of file `scsi_da.c`.

Referenced by `dastart()`.

7.22.4.4 `int da_send_ordered = DA_DEFAULT_SEND_ORDERED [static]`

Definition at line 516 of file `scsi_da.c`.

Referenced by `dainit()`, and `dasendorderedtag()`.

7.22.4.5 `periph_dtor_t dacleanup [static]`

Definition at line 487 of file `scsi_da.c`.

Referenced by `daasync()`.

7.22.4.6 struct `periph_driver dadriver` [static]**Initial value:**

```
{
    dainit, "da",
    TAILQ_HEAD_INITIALIZER(dadriver.units), 0
}
```

Definition at line 546 of file `scsi_da.c`.

Referenced by `dashutdown()`.

7.22.4.7 dumper_t `dadump` [static]

Definition at line 480 of file `scsi_da.c`.

Referenced by `daregister()`.

7.22.4.8 `periph_init_t dainit` [static]

Definition at line 481 of file `scsi_da.c`.

7.22.4.9 `periph_oninv_t daoninvalidate` [static]

Definition at line 489 of file `scsi_da.c`.

Referenced by `daasync()`.

7.22.4.10 `periph_ctor_t daregister` [static]

Definition at line 486 of file `scsi_da.c`.

Referenced by `daasync()`.

7.22.4.11 `timeout_t dasendorderedtag` [static]

Definition at line 498 of file `scsi_da.c`.

Referenced by `dainit()`, and `dasendorderedtag()`.

7.22.4.12 `periph_start_t dastart` [static]

Definition at line 488 of file `scsi_da.c`.

Referenced by `daasync()`.

7.22.4.13 `disk_strategy_t dastrategy` [static]

Definition at line 479 of file `scsi_da.c`.

Referenced by `daregister()`.

7.22.4.14 `const char microp[] = "MICROP" [static]`

Definition at line 144 of file scsi_da.c.

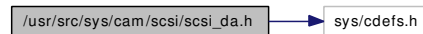
7.22.4.15 `const char quantum[] = "QUANTUM" [static]`

Definition at line 143 of file scsi_da.c.

7.23 /usr/src/sys/cam/scsi/scsi_da.h File Reference

```
#include <sys/cdefs.h>
```

Include dependency graph for scsi_da.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- struct [scsi_rezero_unit](#)
- struct [scsi_format_unit](#)
- struct [scsi_reassign_blocks](#)
- struct [scsi_read_defect_data_10](#)
- struct [scsi_read_defect_data_12](#)
- struct [format_defect_list_header](#)
- struct [format_ipat_descriptor](#)
- struct [scsi_read_format_capacities](#)
- struct [scsi_verify](#)
- struct [scsi_write_and_verify](#)
- struct [format_capacity_list_header](#)
- struct [format_capacity_descriptor](#)
- struct [scsi_reassign_blocks_data](#)
- struct [scsi_read_defect_data_hdr_10](#)
- struct [scsi_defect_desc_block](#)
- struct [scsi_defect_desc_bytes_from_index](#)
- struct [scsi_defect_desc_phys_sector](#)
- struct [scsi_read_defect_data_hdr_12](#)
- union [disk_pages](#)
- struct [disk_pages::format_device_page](#)
- struct [disk_pages::rigid_geometry_page](#)
- struct [disk_pages::flexible_disk_page](#)
- struct [scsi_da_rw_recovery_page](#)

Defines

- #define [_SCSI_SCSI_DA_H](#) 1
- #define [SRZU_LUN_MASK](#) 0xE0
- #define [FU_FORMAT_MASK](#) SRDD10_DLIST_FORMAT_MASK
- #define [FU_BLOCK_FORMAT](#) SRDD10_BLOCK_FORMAT
- #define [FU_BFI_FORMAT](#) SRDD10_BYTES_FROM_INDEX_FORMAT
- #define [FU_PHYS_FORMAT](#) SRDD10_PHYSICAL_SECTOR_FORMAT
- #define [FU_CMPLST](#) 0x08

- #define FU_FMT_DATA 0x10
- #define SRDD10_LUN_MASK 0xE0
- #define SRDD10_GLIST 0x08
- #define SRDD10_PLIST 0x10
- #define SRDD10_DLIST_FORMAT_MASK 0x07
- #define SRDD10_BLOCK_FORMAT 0x00
- #define SRDD10_BYTES_FROM_INDEX_FORMAT 0x04
- #define SRDD10_PHYSICAL_SECTOR_FORMAT 0x05
- #define SRDD12_LUN_MASK 0xE0
- #define SRDD12_GLIST 0x08
- #define SRDD12_PLIST 0x10
- #define SRDD12_DLIST_FORMAT_MASK 0x07
- #define SRDD12_BLOCK_FORMAT 0x00
- #define SRDD12_BYTES_FROM_INDEX_FORMAT 0x04
- #define SRDD12_PHYSICAL_SECTOR_FORMAT 0x05
- #define REZERO_UNIT 0x01
- #define FORMAT_UNIT 0x04
- #define REASSIGN_BLOCKS 0x07
- #define MODE_SELECT 0x15
- #define MODE_SENSE 0x1a
- #define READ_FORMAT_CAPACITIES 0x23
- #define WRITE_AND_VERIFY 0x2e
- #define VERIFY 0x2f
- #define READ_DEFECT_DATA_10 0x37
- #define READ_DEFECT_DATA_12 0xb7
- #define FU_DLH_VS 0x01
- #define FU_DLH_IMMED 0x02
- #define FU_DLH_DSP 0x04
- #define FU_DLH_IP 0x08
- #define FU_DLH_STPF 0x10
- #define FU_DLH_DCRT 0x20
- #define FU_DLH_DPRY 0x40
- #define FU_DLH_FOV 0x80
- #define FU_INIT_NO_HDR 0x00
- #define FU_INIT_LBA_MSB 0x40
- #define FU_INIT_LBA_EACH 0x80
- #define FU_INIT_SI 0x20
- #define FU_INIT_PAT_DEFAULT 0x00
- #define FU_INIT_PAT_REPEAT 0x01
- #define SRFC_LUN_MASK 0xE0
- #define SVFY_LUN_MASK 0xE0
- #define SVFY_RELADR 0x01
- #define SVFY_BYTECHK 0x02
- #define SVFY_DPO 0x10
- #define SWVY_LUN_MASK 0xE0
- #define SWVY_RELADR 0x01
- #define SWVY_BYTECHK 0x02
- #define SWVY_DPO 0x10
- #define FCD_CODE_MASK 0x03
- #define FCD_UNFORMATTED 0x01

- #define `FCD_FORMATTED` 0x02
- #define `FCD_NOMEDIA` 0x03
- #define `SRDDH10_GLIST` 0x08
- #define `SRDDH10_PLIST` 0x10
- #define `SRDDH10_DLIST_FORMAT_MASK` 0x07
- #define `SRDDH10_BLOCK_FORMAT` 0x00
- #define `SRDDH10_BYTES_FROM_INDEX_FORMAT` 0x04
- #define `SRDDH10_PHYSICAL_SECTOR_FORMAT` 0x05
- #define `SRDDH12_GLIST` 0x08
- #define `SRDDH12_PLIST` 0x10
- #define `SRDDH12_DLIST_FORMAT_MASK` 0x07
- #define `SRDDH12_BLOCK_FORMAT` 0x00
- #define `SRDDH12_BYTES_FROM_INDEX_FORMAT` 0x04
- #define `SRDDH12_PHYSICAL_SECTOR_FORMAT` 0x05
- #define `SMS_FORMAT_DEVICE_PAGE` 0x03
- #define `SMS_FORMAT_DEVICE_PLEN` 0x16
- #define `DISK_FMT_SURF` 0x10
- #define `DISK_FMT_RMB` 0x20
- #define `DISK_FMT_HSEC` 0x40
- #define `DISK_FMT_SSEC` 0x80
- #define `SMS_RIGID_GEOMETRY_PAGE` 0x04
- #define `SMS_RIGID_GEOMETRY_PLEN` 0x16
- #define `SMS_FLEXIBLE_GEOMETRY_PAGE` 0x05
- #define `SMS_FLEXIBLE_GEOMETRY_PLEN` 0x1E
- #define `SMS_RW_ERROR_RECOVERY_PAGE` 0x01
- #define `SMS_RWER_AWRE` 0x80
- #define `SMS_RWER_ARRE` 0x40
- #define `SMS_RWER_TB` 0x20
- #define `SMS_RWER_RC` 0x10
- #define `SMS_RWER_EER` 0x08
- #define `SMS_RWER_PER` 0x04
- #define `SMS_RWER_DTE` 0x02
- #define `SMS_RWER_DCR` 0x01

7.23.1 Define Documentation

7.23.1.1 #define `_SCSI_SCSI_DA_H` 1

Definition at line 53 of file `scsi_da.h`.

7.23.1.2 #define `DISK_FMT_HSEC` 0x40

Definition at line 352 of file `scsi_da.h`.

7.23.1.3 #define `DISK_FMT_RMB` 0x20

Definition at line 351 of file `scsi_da.h`.

7.23.1.4 #define DISK_FMT_SSEC 0x80

Definition at line 353 of file scsi_da.h.

7.23.1.5 #define DISK_FMT_SURF 0x10

Definition at line 350 of file scsi_da.h.

7.23.1.6 #define FCD_CODE_MASK 0x03

Definition at line 254 of file scsi_da.h.

7.23.1.7 #define FCD_FORMATTED 0x02

Definition at line 257 of file scsi_da.h.

7.23.1.8 #define FCD_NOMEDIA 0x03

Definition at line 259 of file scsi_da.h.

7.23.1.9 #define FCD_UNFORMATTED 0x01

Definition at line 255 of file scsi_da.h.

7.23.1.10 #define FORMAT_UNIT 0x04

Definition at line 150 of file scsi_da.h.

7.23.1.11 #define FU_BFI_FORMAT SRDD10_BYTES_FROM_INDEX_FORMAT

Definition at line 76 of file scsi_da.h.

7.23.1.12 #define FU_BLOCK_FORMAT SRDD10_BLOCK_FORMAT

Definition at line 75 of file scsi_da.h.

7.23.1.13 #define FU_CMPLST 0x08

Definition at line 78 of file scsi_da.h.

7.23.1.14 #define FU_DLH_DCRT 0x20

Definition at line 169 of file scsi_da.h.

7.23.1.15 #define FU_DLH_DPRY 0x40

Definition at line 170 of file scsi_da.h.

7.23.1.16 #define FU_DLH_DSP 0x04

Definition at line 166 of file scsi_da.h.

7.23.1.17 #define FU_DLH_FOV 0x80

Definition at line 171 of file scsi_da.h.

7.23.1.18 #define FU_DLH_IMMED 0x02

Definition at line 165 of file scsi_da.h.

7.23.1.19 #define FU_DLH_IP 0x08

Definition at line 167 of file scsi_da.h.

7.23.1.20 #define FU_DLH_STPF 0x10

Definition at line 168 of file scsi_da.h.

7.23.1.21 #define FU_DLH_VS 0x01

Definition at line 164 of file scsi_da.h.

7.23.1.22 #define FU_FMT_DATA 0x10

Definition at line 79 of file scsi_da.h.

7.23.1.23 #define FU_FORMAT_MASK SRDD10_DLIST_FORMAT_MASK

Definition at line 74 of file scsi_da.h.

7.23.1.24 #define FU_INIT_LBA_EACH 0x80

Definition at line 180 of file scsi_da.h.

7.23.1.25 #define FU_INIT_LBA_MSB 0x40

Definition at line 179 of file scsi_da.h.

7.23.1.26 #define FU_INIT_NO_HDR 0x00

Definition at line 178 of file scsi_da.h.

7.23.1.27 #define FU_INIT_PAT_DEFAULT 0x00

Definition at line 183 of file scsi_da.h.

7.23.1.28 #define FU_INIT_PAT_REPEAT 0x01

Definition at line 184 of file scsi_da.h.

7.23.1.29 #define FU_INIT_SI 0x20

Definition at line 181 of file scsi_da.h.

7.23.1.30 #define FU_PHYS_FORMAT SRDD10_PHYSICAL_SECTOR_FORMAT

Definition at line 77 of file scsi_da.h.

7.23.1.31 #define MODE_SELECT 0x15

Definition at line 152 of file scsi_da.h.

7.23.1.32 #define MODE_SENSE 0x1a

Definition at line 153 of file scsi_da.h.

7.23.1.33 #define READ_DEFECT_DATA_10 0x37

Definition at line 157 of file scsi_da.h.

7.23.1.34 #define READ_DEFECT_DATA_12 0xb7

Definition at line 158 of file scsi_da.h.

7.23.1.35 #define READ_FORMAT_CAPACITIES 0x23

Definition at line 154 of file scsi_da.h.

7.23.1.36 #define REASSIGN_BLOCKS 0x07

Definition at line 151 of file scsi_da.h.

7.23.1.37 #define REZERO_UNIT 0x01

Definition at line 149 of file scsi_da.h.

7.23.1.38 #define SMS_FLEXIBLE_GEOMETRY_PAGE 0x05

Definition at line 388 of file scsi_da.h.

7.23.1.39 #define SMS_FLEXIBLE_GEOMETRY_PLEN 0x1E

Definition at line 390 of file scsi_da.h.

7.23.1.40 #define SMS_FORMAT_DEVICE_PAGE 0x03

Definition at line 328 of file scsi_da.h.

7.23.1.41 #define SMS_FORMAT_DEVICE_PLEN 0x16

Definition at line 330 of file scsi_da.h.

7.23.1.42 #define SMS_RIGID_GEOMETRY_PAGE 0x04

Definition at line 360 of file scsi_da.h.

7.23.1.43 #define SMS_RIGID_GEOMETRY_PLEN 0x16

Definition at line 362 of file scsi_da.h.

7.23.1.44 #define SMS_RW_ERROR_RECOVERY_PAGE 0x01

Definition at line 426 of file scsi_da.h.

7.23.1.45 #define SMS_RWER_ARRE 0x40

Definition at line 430 of file scsi_da.h.

7.23.1.46 #define SMS_RWER_AWRE 0x80

Definition at line 429 of file scsi_da.h.

7.23.1.47 #define SMS_RWER_DCR 0x01

Definition at line 436 of file scsi_da.h.

7.23.1.48 #define SMS_RWER_DTE 0x02

Definition at line 435 of file scsi_da.h.

7.23.1.49 #define SMS_RWER_EER 0x08

Definition at line 433 of file scsi_da.h.

7.23.1.50 #define SMS_RWER_PER 0x04

Definition at line 434 of file scsi_da.h.

7.23.1.51 #define SMS_RWER_RC 0x10

Definition at line 432 of file scsi_da.h.

7.23.1.52 #define SMS_RWER_TB 0x20

Definition at line 431 of file scsi_da.h.

7.23.1.53 #define SRDD10_BLOCK_FORMAT 0x00

Definition at line 106 of file scsi_da.h.

7.23.1.54 #define SRDD10_BYTES_FROM_INDEX_FORMAT 0x04

Definition at line 107 of file scsi_da.h.

7.23.1.55 #define SRDD10_DLIST_FORMAT_MASK 0x07

Definition at line 105 of file scsi_da.h.

7.23.1.56 #define SRDD10_GLIST 0x08

Definition at line 103 of file scsi_da.h.

7.23.1.57 #define SRDD10_LUN_MASK 0xE0

Definition at line 101 of file scsi_da.h.

7.23.1.58 #define SRDD10_PHYSICAL_SECTOR_FORMAT 0x05

Definition at line 108 of file scsi_da.h.

7.23.1.59 #define SRDD10_PLIST 0x10

Definition at line 104 of file scsi_da.h.

7.23.1.60 #define SRDD12_BLOCK_FORMAT 0x00

Definition at line 132 of file scsi_da.h.

7.23.1.61 #define SRDD12_BYTES_FROM_INDEX_FORMAT 0x04

Definition at line 133 of file scsi_da.h.

7.23.1.62 #define SRDD12_DLIST_FORMAT_MASK 0x07

Definition at line 131 of file scsi_da.h.

7.23.1.63 #define SRDD12_GLIST 0x08

Definition at line 129 of file scsi_da.h.

7.23.1.64 #define SRDD12_LUN_MASK 0xE0

Definition at line 126 of file scsi_da.h.

7.23.1.65 #define SRDD12_PHYSICAL_SECTOR_FORMAT 0x05

Definition at line 134 of file scsi_da.h.

7.23.1.66 #define SRDD12_PLIST 0x10

Definition at line 130 of file scsi_da.h.

7.23.1.67 #define SRDDH10_BLOCK_FORMAT 0x00

Definition at line 285 of file scsi_da.h.

7.23.1.68 #define SRDDH10_BYTES_FROM_INDEX_FORMAT 0x04

Definition at line 286 of file scsi_da.h.

7.23.1.69 #define SRDDH10_DLIST_FORMAT_MASK 0x07

Definition at line 284 of file scsi_da.h.

7.23.1.70 #define SRDDH10_GLIST 0x08

Definition at line 282 of file scsi_da.h.

7.23.1.71 #define SRDDH10_PHYSICAL_SECTOR_FORMAT 0x05

Definition at line 287 of file scsi_da.h.

7.23.1.72 #define SRDDH10_PLIST 0x10

Definition at line 283 of file scsi_da.h.

7.23.1.73 #define SRDDH12_BLOCK_FORMAT 0x00

Definition at line 317 of file scsi_da.h.

7.23.1.74 #define SRDDH12_BYTES_FROM_INDEX_FORMAT 0x04

Definition at line 318 of file scsi_da.h.

7.23.1.75 #define SRDDH12_DLIST_FORMAT_MASK 0x07

Definition at line 316 of file scsi_da.h.

7.23.1.76 #define SRDDH12_GLIST 0x08

Definition at line 314 of file scsi_da.h.

7.23.1.77 #define SRDDH12_PHYSICAL_SECTOR_FORMAT 0x05

Definition at line 319 of file scsi_da.h.

7.23.1.78 #define SRDDH12_PLIST 0x10

Definition at line 315 of file scsi_da.h.

7.23.1.79 #define SRFC_LUN_MASK 0xE0

Definition at line 192 of file scsi_da.h.

7.23.1.80 #define SRZU_LUN_MASK 0xE0

Definition at line 60 of file scsi_da.h.

7.23.1.81 #define SVFY_BYTECHK 0x02

Definition at line 204 of file scsi_da.h.

7.23.1.82 #define SVFY_DPO 0x10

Definition at line 205 of file scsi_da.h.

7.23.1.83 #define SVFY_LUN_MASK 0xE0

Definition at line 202 of file scsi_da.h.

7.23.1.84 #define SVFY_RELADR 0x01

Definition at line 203 of file scsi_da.h.

7.23.1.85 #define SWVY_BYTECHK 0x02

Definition at line 218 of file scsi_da.h.

7.23.1.86 #define SWVY_DPO 0x10

Definition at line 219 of file scsi_da.h.

7.23.1.87 #define SWVY_LUN_MASK 0xE0

Definition at line 216 of file scsi_da.h.

7.23.1.88 #define SWVY_RELADR 0x01

Definition at line 217 of file scsi_da.h.

7.23.1.89 #define VERIFY 0x2f

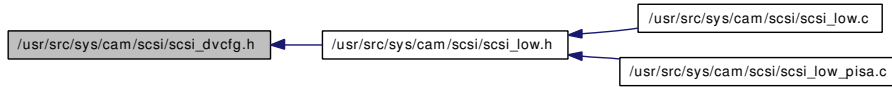
Definition at line 156 of file scsi_da.h.

7.23.1.90 #define WRITE_AND_VERIFY 0x2e

Definition at line 155 of file scsi_da.h.

7.24 /usr/src/sys/cam/scsi/scsi_dvcfg.h File Reference

This graph shows which files directly or indirectly include this file:



Defines

- #define `DVF SCSI_SYNC` 0x01
- #define `DVF SCSI_DISC` 0x02
- #define `DVF SCSI_WAIT` 0x04
- #define `DVF SCSI_LINK` 0x08
- #define `DVF SCSI_QTAG` 0x10
- #define `DVF SCSI_SP0` 0x100
- #define `DVF SCSI_NOPARITY` 0x200
- #define `DVF SCSI_SAVESP` 0x400
- #define `DVF SCSI_SPI` 0x800
- #define `DVF SCSI_PERIOD`(XXX) (((XXX) >> 24) & 0xff)
- #define `DVF SCSI_OFFSET`(XXX) (((XXX) >> 16) & 0xff)
- #define `DVF SCSI_SYNCMASK` 0xffff0000
- #define `DVF SCSI_DEF CFG` (DVF SCSI_SYNC | DVF SCSI_NOPARITY | DVF SCSI_SYNCMASK)
- #define `DVF SCSI_BITS` "\020\13fssp\12noparity\11nosat\005qtag\004cmdlnk\003wait\002disc\001sync"

7.24.1 Define Documentation

7.24.1.1 #define DVF SCSI_-BITS "\020\13fssp\12noparity\11nosat\005qtag\004cmdlnk\003wait\002disc\001sync"

Definition at line 56 of file scsi_dvcfg.h.

7.24.1.2 #define DVF SCSI_DEF CFG (DVF SCSI_SYNC | DVF SCSI_NOPARITY | DVF SCSI_SYNCMASK)

Definition at line 54 of file scsi_dvcfg.h.

7.24.1.3 #define DVF SCSI_DISC 0x02

Definition at line 42 of file scsi_dvcfg.h.

7.24.1.4 #define DVF SCSI_LINK 0x08

Definition at line 44 of file scsi_dvcfg.h.

7.24.1.5 #define DVF SCSI_NOPARITY 0x200

Definition at line 47 of file scsi_dvcfg.h.

7.24.1.6 #define DVF SCSI_OFFSET(XXX) (((XXX) >> 16) & 0xff)

Definition at line 51 of file scsi_dvcfg.h.

7.24.1.7 #define DVF SCSI_PERIOD(XXX) (((XXX) >> 24) & 0xff)

Definition at line 50 of file scsi_dvcfg.h.

7.24.1.8 #define DVF SCSI_QTAG 0x10

Definition at line 45 of file scsi_dvcfg.h.

7.24.1.9 #define DVF SCSI_SAVESP 0x400

Definition at line 48 of file scsi_dvcfg.h.

7.24.1.10 #define DVF SCSI_SP0 0x100

Definition at line 46 of file scsi_dvcfg.h.

7.24.1.11 #define DVF SCSI_SP1 0x800

Definition at line 49 of file scsi_dvcfg.h.

7.24.1.12 #define DVF SCSI_SYNC 0x01

Definition at line 41 of file scsi_dvcfg.h.

7.24.1.13 #define DVF SCSI_SYNCMASK 0xffff0000

Definition at line 52 of file scsi_dvcfg.h.

7.24.1.14 #define DVF SCSI_WAIT 0x04

Definition at line 43 of file scsi_dvcfg.h.

7.25 /usr/src/sys/cam/scsi/scsi_iu.h File Reference

Data Structures

- struct [scsi_status_iu_header](#)

Defines

- #define [_SCSI_SCSI_IU_H](#) 1
- #define [SIU_SNSVALID](#) 0x2
- #define [SIU_RSPVALID](#) 0x1
- #define [SIU_PKTFAIL_OFFSET](#)(siu) 12
- #define [SIU_PKTFAIL_CODE](#)(siu) (scsi_4btoul((siu) → pkt_failures) & 0xFF)
- #define [SIU_PFC_NONE](#) 0
- #define [SIU_PFC_CIU_FIELDS_INVALID](#) 2
- #define [SIU_PFC_TMF_NOT_SUPPORTED](#) 4
- #define [SIU_PFC_TMF_FAILED](#) 5
- #define [SIU_PFC_INVALID_TYPE_CODE](#) 6
- #define [SIU_PFC_ILLEGAL_REQUEST](#) 7
- #define [SIU_SENSE_OFFSET](#)(siu)
- #define [SIU_TASKMGMT_NONE](#) 0x00
- #define [SIU_TASKMGMT_ABORT_TASK](#) 0x01
- #define [SIU_TASKMGMT_ABORT_TASK_SET](#) 0x02
- #define [SIU_TASKMGMT_CLEAR_TASK_SET](#) 0x04
- #define [SIU_TASKMGMT_LUN_RESET](#) 0x08
- #define [SIU_TASKMGMT_TARGET_RESET](#) 0x20
- #define [SIU_TASKMGMT_CLEAR_ACA](#) 0x40

7.25.1 Define Documentation

7.25.1.1 #define [_SCSI_SCSI_IU_H](#) 1

Definition at line 6 of file [scsi_iu.h](#).

7.25.1.2 #define [SIU_PFC_CIU_FIELDS_INVALID](#) 2

Definition at line 23 of file [scsi_iu.h](#).

7.25.1.3 #define [SIU_PFC_ILLEGAL_REQUEST](#) 7

Definition at line 27 of file [scsi_iu.h](#).

7.25.1.4 #define [SIU_PFC_INVALID_TYPE_CODE](#) 6

Definition at line 26 of file [scsi_iu.h](#).

7.25.1.5 #define SIU_PFC_NONE 0

Definition at line 22 of file scsi_iu.h.

7.25.1.6 #define SIU_PFC_TMF_FAILED 5

Definition at line 25 of file scsi_iu.h.

7.25.1.7 #define SIU_PFC_TMF_NOT_SUPPORTED 4

Definition at line 24 of file scsi_iu.h.

7.25.1.8 #define SIU_PKTFAIL_CODE(siu) (scsi_4btoul((siu) → pkt_failures) & 0xFF)

Definition at line 21 of file scsi_iu.h.

7.25.1.9 #define SIU_PKTFAIL_OFFSET(siu) 12

Definition at line 20 of file scsi_iu.h.

7.25.1.10 #define SIU_RSPVALID 0x1

Definition at line 13 of file scsi_iu.h.

7.25.1.11 #define SIU_SENSE_OFFSET(siu)**Value:**

```
(12 + (((siu)->flags & SIU_RSPVALID)           \
      ? scsi_4btoul((siu)->pkt_failures_length) \
      : 0))
```

Definition at line 28 of file scsi_iu.h.

7.25.1.12 #define SIU_SNSVALID 0x2

Definition at line 12 of file scsi_iu.h.

7.25.1.13 #define SIU_TASKMGMT_ABORT_TASK 0x01

Definition at line 34 of file scsi_iu.h.

7.25.1.14 #define SIU_TASKMGMT_ABORT_TASK_SET 0x02

Definition at line 35 of file scsi_iu.h.

7.25.1.15 #define SIU_TASKMGMT_CLEAR_ACA 0x40

Definition at line 39 of file scsi_iu.h.

7.25.1.16 #define SIU_TASKMGMT_CLEAR_TASK_SET 0x04

Definition at line 36 of file scsi_iu.h.

7.25.1.17 #define SIU_TASKMGMT_LUN_RESET 0x08

Definition at line 37 of file scsi_iu.h.

7.25.1.18 #define SIU_TASKMGMT_NONE 0x00

Definition at line 33 of file scsi_iu.h.

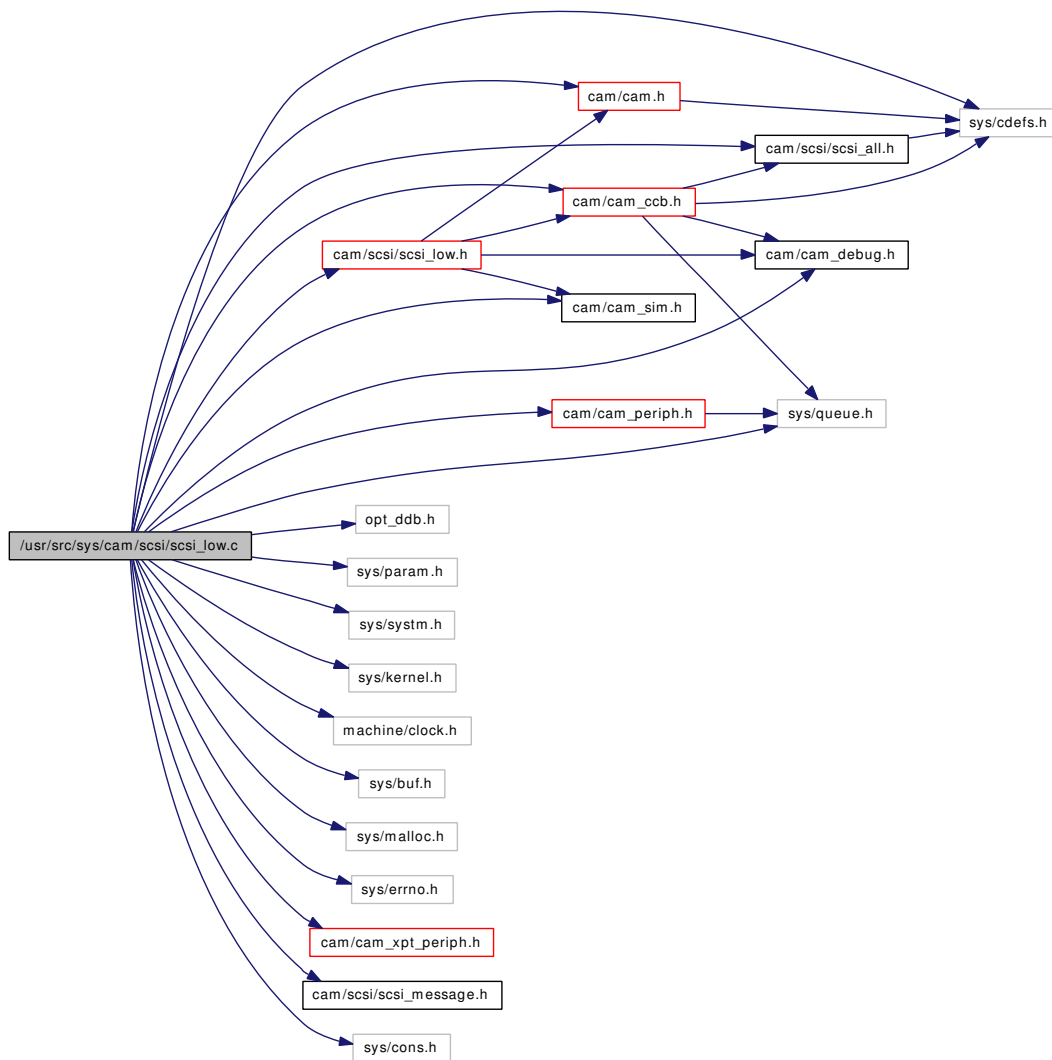
7.25.1.19 #define SIU_TASKMGMT_TARGET_RESET 0x20

Definition at line 38 of file scsi_iu.h.

7.26 /usr/src/sys/cam/scsi/scsi_low.c File Reference

```
#include <sys/cdefs.h>
#include "opt_ddb.h"
#include <sys/param.h>
#include <sys/system.h>
#include <sys/kernel.h>
#include <machine/clock.h>
#include <sys/buf.h>
#include <sys/queue.h>
#include <sys/malloc.h>
#include <sys/errno.h>
#include <cam/cam.h>
#include <cam/cam_ccb.h>
#include <cam/cam_sim.h>
#include <cam/cam_debug.h>
#include <cam/cam_periph.h>
#include <cam/cam_xpt_periph.h>
#include <cam/scsi/scsi_all.h>
#include <cam/scsi/scsi_message.h>
#include <cam/scsi/scsi_low.h>
#include <sys/cons.h>
```

Include dependency graph for scsi_low.c:



Data Structures

- struct [scsi_low_statics](#)
- struct [scsi_low_error_code](#)
- struct [scsi_low_msgout_data](#)
- struct [scsi_low_msgin_data](#)

Defines

- #define [SCSI_LOW_STATICS](#)
- #define [SCSI_LOW_DEBUG](#)
- #define [SCSI_LOW_NEGOTIATE_BEFORE_SENSE](#)
- #define [SCSI_LOW_START_UP_CHECK](#)
- #define [SCSI_LOW_FLAGS_QUIRKS_OK](#)

- #define `SCSI_LOW_POLL_HZ` 1000
- #define `SCSI_LOW_START_NO_QTAG` 0
- #define `SCSI_LOW_START_QTAG` 1
- #define `SCSI_LOW_DONE_COMPLETE` 0
- #define `SCSI_LOW_DONE_RETRY` 1
- #define `SCSI_LOW_DISK_DISC` 0x00000001
- #define `SCSI_LOW_DISK_QTAG` 0x00000002
- #define `SCSI_LOW_DISK_LINK` 0x00000004
- #define `SCSI_LOW_DISK_PARITY` 0x00000008
- #define `SCSI_LOW_DISK_SYNC` 0x00010000
- #define `SCSI_LOW_DISK_WIDE_16` 0x00020000
- #define `SCSI_LOW_DISK_WIDE_32` 0x00040000
- #define `SCSI_LOW_DISK_WIDE` (`SCSI_LOW_DISK_WIDE_16` | `SCSI_LOW_DISK_WIDE_32`)
- #define `SCSI_LOW_DISK_LFLAGS` 0x0000ffff
- #define `SCSI_LOW_DISK_TFLAGS` 0xffff0000
- #define `SCSI_LOW_INFO`(slp, ti, s) `printf("%s: %s\n", (slp) → sl_xname, (s))`
- #define `SCSI_LOW_DEBUG_DONE` 0x00001
- #define `SCSI_LOW_DEBUG_DISC` 0x00002
- #define `SCSI_LOW_DEBUG_SENSE` 0x00004
- #define `SCSI_LOW_DEBUG_CALCF` 0x00008
- #define `SCSI_LOW_DEBUG_ACTION` 0x10000
- #define `SCSI_LOW_MAX_ATTEN_CHECK` 32
- #define `SCSI_LOW_ATTEN_CHECK` 0x0001
- #define `SCSI_LOW_CMDLNK_CHECK` 0x0002
- #define `SCSI_LOW_ABORT_CHECK` 0x0004
- #define `SCSI_LOW_NEXUS_CHECK` 0x0008
- #define `SCSI_LOW_DEBUG_TEST_GO`(fl, id) `((scsi_low_test & (fl)) != 0 && (scsi_low_test_id & (1 << (id))) == 0)`
- #define `SCSI_LOW_DEBUG_GO`(fl, id) `((scsi_low_debug & (fl)) != 0 && (scsi_low_test_id & (1 << (id))) == 0)`
- #define `SCSI_LOW_INLINE` `static __inline`
- #define `SCSI_LOW_CMD_RESIDUAL_CHK` 0x0001
- #define `SCSI_LOW_CMD_ORDERED_QTAG` 0x0002
- #define `SCSI_LOW_CMD_ABORT_WARNING` 0x0004
- #define `SLSC_MODE_SENSE_SHORT` 0x1a
- #define `SCSI_LOW_QTAG_OK` (`SCSI_LOW_QTAG` | `SCSI_LOW_DISC`)
- #define `SCSI_LOW_MSG_ABORT_OK`
- #define `TWIDDLEWAIT` 10000
- #define `MSGCMD_LUN`(msg) `(msg & 0x07)`
- #define `MSGINPTR_CLR`(ti) `{(ti) → ti_msginptr = 0; (ti) → ti_msginlen = 0;}`
- #define `MSGIN_PERIOD`(ti) `((ti) → ti_msgin[3])`
- #define `MSGIN_OFFSET`(ti) `((ti) → ti_msgin[4])`
- #define `MSGIN_WIDTHHP`(ti) `((ti) → ti_msgin[3])`
- #define `MSGIN_DATA_LAST` 0x30
- #define `MSG_RELEASE_ATN` 0x0001
- #define `SCSI_LOW_CMDLNK_NOK` (`CCB_INTERNAL` | `CCB_SENSE` | `CCB_CLEARQ`)

Functions

- `__FBSDDID` ("FreeBSD: src/sys/cam/scsi/scsi_low.c,v 1.26 2006/11/02 00:54:33 mjacob Exp \$")
- `MALLOC_DEFINE` (M_SCSILOW,"SCSI low","SCSI low buffers")
- void `scsi_low_info` (struct `scsi_low_softc` *, struct `targ_info` *, u_char *)
- static void `scsi_low_engage` (void *)
- static struct `slccb` * `scsi_low_establish_ccb` (struct `targ_info` *, struct `lun_info` *, `scsi_low_tag_t`)
- static int `scsi_low_done` (struct `scsi_low_softc` *, struct `slccb` *)
- static int `scsi_low_setup_done` (struct `scsi_low_softc` *, struct `slccb` *)
- static void `scsi_low_bus_release` (struct `scsi_low_softc` *, struct `targ_info` *)
- static void `scsi_low_twiddle_wait` (void)
- static struct `lun_info` * `scsi_low_alloc_li` (struct `targ_info` *, int, int)
- static struct `targ_info` * `scsi_low_alloc_ti` (struct `scsi_low_softc` *, int)
- static void `scsi_low_calcf_lun` (struct `lun_info` *)
- static void `scsi_low_calcf_target` (struct `targ_info` *)
- static void `scsi_low_calcf_show` (struct `lun_info` *)
- static void `scsi_low_reset_nexus` (struct `scsi_low_softc` *, int)
- static void `scsi_low_reset_nexus_target` (struct `scsi_low_softc` *, struct `targ_info` *, int)
- static void `scsi_low_reset_nexus_lun` (struct `scsi_low_softc` *, struct `lun_info` *, int)
- static int `scsi_low_init` (struct `scsi_low_softc` *, u_int)
- static void `scsi_low_start` (struct `scsi_low_softc` *)
- static void `scsi_low_free_ti` (struct `scsi_low_softc` *)
- static int `scsi_low_alloc_qtag` (struct `slccb` *)
- static int `scsi_low_dealloc_qtag` (struct `slccb` *)
- static int `scsi_low_enqueue` (struct `scsi_low_softc` *, struct `targ_info` *, struct `lun_info` *, struct `slccb` *, u_int, u_int)
- static int `scsi_low_message_enqueue` (struct `scsi_low_softc` *, struct `targ_info` *, struct `lun_info` *, u_int)
- static void `scsi_low_unit_ready_cmd` (struct `slccb` *)
- static void `scsi_low_timeout` (void *)
- static int `scsi_low_timeout_check` (struct `scsi_low_softc` *)
- static int `scsi_low_start_up` (struct `scsi_low_softc` *)
- static int `scsi_low_abort_ccb` (struct `scsi_low_softc` *, struct `slccb` *)
- static struct `slccb` * `scsi_low_revoke_ccb` (struct `scsi_low_softc` *, struct `slccb` *, int)
- static void `scsi_low_test_abort` (struct `scsi_low_softc` *, struct `targ_info` *, struct `lun_info` *)
- static void `scsi_low_test_cmdlnk` (struct `scsi_low_softc` *, struct `slccb` *)
- static void `scsi_low_test_attn` (struct `scsi_low_softc` *, struct `targ_info` *, u_int)
- SCSI_LOW_INLINE void `scsi_low_activate_qtag` (struct `slccb` *)
- SCSI_LOW_INLINE void `scsi_low_deactivate_qtag` (struct `slccb` *)
- SCSI_LOW_INLINE void `scsi_low_ccb_message_assert` (struct `slccb` *, u_int)
- SCSI_LOW_INLINE void `scsi_low_ccb_message_exec` (struct `scsi_low_softc` *, struct `slccb` *)
- SCSI_LOW_INLINE void `scsi_low_ccb_message_retry` (struct `slccb` *)
- SCSI_LOW_INLINE void `scsi_low_ccb_message_clear` (struct `slccb` *)
- SCSI_LOW_INLINE void `scsi_low_init_msgsyst` (struct `scsi_low_softc` *, struct `targ_info` *)
- static struct `slccb` * `scsi_low_find_ccb` (struct `scsi_low_softc` *, u_int, u_int, void *)
- static int `scsi_low_translate_error_code` (struct `slccb` *, struct `scsi_low_error_code` *)
- int `scsi_low_is_busy` (struct `scsi_low_softc` *slp)
- int `scsi_low_deactivate` (struct `scsi_low_softc` *slp)
- int `scsi_low_activate` (struct `scsi_low_softc` *slp)
- void `scsi_low_bus_idle` (struct `scsi_low_softc` *slp)

- int `scsi_low_attach` (struct `scsi_low_softc` *slp, int openings, int ntargs, int nluns, int targsize, int lunsize)
- int `scsi_low_dettach` (struct `scsi_low_softc` *slp)
- static int `scsi_low_setup_start` (struct `scsi_low_softc` *, struct `targ_info` *, struct `lun_info` *, struct `slccb` *)
- static int `scsi_low_sense_abort_start` (struct `scsi_low_softc` *, struct `targ_info` *, struct `lun_info` *, struct `slccb` *)
- static int `scsi_low_resume` (struct `scsi_low_softc` *)
- void `scsi_low_arbit_fail` (struct `scsi_low_softc` *slp, struct `slccb` *cb)
- void `scsi_low_bus_reset` (struct `scsi_low_softc` *slp)
- int `scsi_low_restart` (struct `scsi_low_softc` *slp, int flags, u_char *s)
- `targ_info` * `scsi_low_reselected` (struct `scsi_low_softc` *slp, u_int targ)
- int `scsi_low_cmd` (struct `scsi_low_softc` *slp, struct `targ_info` *ti)
- int `scsi_low_data` (struct `scsi_low_softc` *slp, struct `targ_info` *ti, struct buf **bp, int direction)
- static int `scsi_low_errfunc_synch` (struct `scsi_low_softc` *, u_int)
- static int `scsi_low_errfunc_wide` (struct `scsi_low_softc` *, u_int)
- static int `scsi_low_errfunc_identify` (struct `scsi_low_softc` *, u_int)
- static int `scsi_low_errfunc_qtag` (struct `scsi_low_softc` *, u_int)
- static int `scsi_low_msgfunc_synch` (struct `scsi_low_softc` *)
- static int `scsi_low_msgfunc_wide` (struct `scsi_low_softc` *)
- static int `scsi_low_msgfunc_identify` (struct `scsi_low_softc` *)
- static int `scsi_low_msgfunc_abort` (struct `scsi_low_softc` *)
- static int `scsi_low_msgfunc_qabort` (struct `scsi_low_softc` *)
- static int `scsi_low_msgfunc_qtag` (struct `scsi_low_softc` *)
- static int `scsi_low_msgfunc_reset` (struct `scsi_low_softc` *)
- static int `scsi_low_msginfunc_ext` (struct `scsi_low_softc` *)
- static int `scsi_low_synch` (struct `scsi_low_softc` *)
- static int `scsi_low_wide` (struct `scsi_low_softc` *)
- static int `scsi_low_msginfunc_msg_reject` (struct `scsi_low_softc` *)
- static int `scsi_low_msginfunc_rejop` (struct `scsi_low_softc` *)
- static int `scsi_low_msginfunc_rp` (struct `scsi_low_softc` *)
- static int `scsi_low_msginfunc_sdp` (struct `scsi_low_softc` *)
- static int `scsi_low_msginfunc_disc` (struct `scsi_low_softc` *)
- static int `scsi_low_msginfunc_cc` (struct `scsi_low_softc` *)
- static int `scsi_low_msginfunc_lcc` (struct `scsi_low_softc` *)
- static int `scsi_low_msginfunc_parity` (struct `scsi_low_softc` *)
- static int `scsi_low_msginfunc_noop` (struct `scsi_low_softc` *)
- static int `scsi_low_msginfunc_simple_qtag` (struct `scsi_low_softc` *)
- static int `scsi_low_msginfunc_i_wide_residue` (struct `scsi_low_softc` *)
- int `scsi_low_msgout` (struct `scsi_low_softc` *slp, struct `targ_info` *ti, u_int fl)
- int `scsi_low_msgin` (struct `scsi_low_softc` *slp, struct `targ_info` *ti, u_int c)
- int `scsi_low_disconnected` (struct `scsi_low_softc` *slp, struct `targ_info` *ti)
- static int `scsi_low_poll` (struct `scsi_low_softc` *, struct `slccb` *)
- void `scsi_low_print` (struct `scsi_low_softc` *slp, struct `targ_info` *ti)

Variables

- int `scsi_low_version_major` = 2
- int `scsi_low_version_minor` = 17
- static struct `scsi_low_softc_tab` `sl_tab` = LIST_HEAD_INITIALIZER(`sl_tab`)
- int `scsi_low_debug` = 0
- int `scsi_low_test` = 0
- int `scsi_low_test_id` = 0
- static u_int8_t `scsi_low_cmd_flags` [256]
- static u_int8_t `ss_cmd` [6] = {START_STOP, 0, 0, 0, SSS_START, 0}
- static u_int8_t `sms_cmd` [6]
- static u_int8_t `inq_cmd` [6]
- static u_int8_t `unit_ready_cmd` [6]
- static int `tw_pos`
- static char `tw_chars` [] = "|/-\\"
- `scsi_low_msgout_data` `scsi_low_msgout_data` []
- `scsi_low_msgin_data` `scsi_low_msgin_data` []
- static u_char * `phase` []

7.26.1 Define Documentation

7.26.1.1 #define MSG_RELEASE_ATN 0x0001

Definition at line 3191 of file `scsi_low.c`.

Referenced by `scsi_low_msgout()`.

7.26.1.2 #define MSGCMD_LUN(msg) (msg & 0x07)

Definition at line 2954 of file `scsi_low.c`.

Referenced by `scsi_low_msgin()`.

7.26.1.3 #define MSGIN_DATA_LAST 0x30

Definition at line 3171 of file `scsi_low.c`.

Referenced by `scsi_low_msgin()`.

7.26.1.4 #define MSGIN_OFFSET(ti) ((ti) → ti_msgin[4])

Definition at line 3169 of file `scsi_low.c`.

Referenced by `scsi_low_errfunc_synch()`, and `scsi_low_synch()`.

7.26.1.5 #define MSGIN_PERIOD(ti) ((ti) → ti_msgin[3])

Definition at line 3168 of file `scsi_low.c`.

Referenced by `scsi_low_errfunc_synch()`, and `scsi_low_synch()`.

7.26.1.6 #define MSGIN_WIDTHP(ti) ((ti) → ti_msgin[3])

Definition at line 3170 of file scsi_low.c.

Referenced by scsi_low_errfunc_wide(), and scsi_low_wide().

7.26.1.7 #define MSGINPTR_CLR(ti) {(ti) → ti_msginptr = 0; (ti) → ti_msginlen = 0;}

Definition at line 3167 of file scsi_low.c.

Referenced by scsi_low_msgin().

7.26.1.8 #define SCSI_LOW_ABORT_CHECK 0x0004

Definition at line 220 of file scsi_low.c.

7.26.1.9 #define SCSI_LOW_ATTEN_CHECK 0x0001

Definition at line 218 of file scsi_low.c.

Referenced by scsi_low_disconnected(), scsi_low_establish_ccb(), and scsi_low_msginfunc_ext().

7.26.1.10 #define SCSI_LOW_CMD_ABORT_WARNING 0x0004

Definition at line 329 of file scsi_low.c.

7.26.1.11 #define SCSI_LOW_CMD_ORDERED_QTAG 0x0002

Definition at line 328 of file scsi_low.c.

Referenced by scsi_low_start().

7.26.1.12 #define SCSI_LOW_CMD_RESIDUAL_CHK 0x0001

Definition at line 327 of file scsi_low.c.

Referenced by scsi_low_done().

7.26.1.13 #define SCSI_LOW_CMDLNK_CHECK 0x0002

Definition at line 219 of file scsi_low.c.

Referenced by scsi_low_cmd().

7.26.1.14 #define SCSI_LOW_CMDLNK_NOK (CCB_INTERNAL | CCB_SENSE | CCB_CLEARQ)

Referenced by scsi_low_test_cmdlnk().

7.26.1.15 #define SCSI_LOW_DEBUG

Definition at line 8 of file scsi_low.c.

7.26.1.16 #define SCSI_LOW_DEBUG_ACTION 0x10000

Definition at line 214 of file scsi_low.c.

7.26.1.17 #define SCSI_LOW_DEBUG_CALCF 0x00008

Definition at line 213 of file scsi_low.c.

Referenced by scsi_low_calcf_lun(), and scsi_low_calcf_target().

7.26.1.18 #define SCSI_LOW_DEBUG_DISC 0x00002

Definition at line 211 of file scsi_low.c.

Referenced by scsi_low_disconnected().

7.26.1.19 #define SCSI_LOW_DEBUG_DONE 0x00001

Definition at line 210 of file scsi_low.c.

Referenced by scsi_low_done().

7.26.1.20 #define SCSI_LOW_DEBUG_GO(fl, id) ((scsi_low_debug & (fl)) != 0 && (scsi_low_test_id & (1 << (id))) == 0)

Definition at line 230 of file scsi_low.c.

Referenced by scsi_low_calcf_lun(), scsi_low_calcf_target(), scsi_low_disconnected(), and scsi_low_done().

7.26.1.21 #define SCSI_LOW_DEBUG_SENSE 0x00004

Definition at line 212 of file scsi_low.c.

Referenced by scsi_low_setup_done().

7.26.1.22 #define SCSI_LOW_DEBUG_TEST_GO(fl, id) ((scsi_low_test & (fl)) != 0 && (scsi_low_test_id & (1 << (id))) == 0)

Definition at line 228 of file scsi_low.c.

Referenced by scsi_low_cmd(), scsi_low_disconnected(), scsi_low_establish_ccb(), scsi_low_msgin(), scsi_low_msginfunc_ext(), and scsi_low_msginfunc_simple_qtag().

7.26.1.23 #define SCSI_LOW_DISK_DISC 0x00000001

Definition at line 137 of file scsi_low.c.

7.26.1.24 #define SCSI_LOW_DISK_LFLAGS 0x0000ffff

Definition at line 145 of file scsi_low.c.

Referenced by scsi_low_alloc_li(), and scsi_low_reset_nexus_target().

7.26.1.25 #define SCSI_LOW_DISK_LINK 0x00000004

Definition at line 139 of file scsi_low.c.

Referenced by scsi_low_calcf_lun(), and scsi_low_setup_done().

7.26.1.26 #define SCSI_LOW_DISK_PARITY 0x00000008

Definition at line 140 of file scsi_low.c.

Referenced by scsi_low_calcf_lun().

7.26.1.27 #define SCSI_LOW_DISK_QTAG 0x00000002

Definition at line 138 of file scsi_low.c.

Referenced by scsi_low_calcf_lun(), and scsi_low_setup_done().

7.26.1.28 #define SCSI_LOW_DISK_SYNC 0x00010000

Definition at line 141 of file scsi_low.c.

Referenced by scsi_low_calcf_target(), and scsi_low_setup_done().

7.26.1.29 #define SCSI_LOW_DISK_TFLAGS 0xffff0000

Definition at line 146 of file scsi_low.c.

Referenced by scsi_low_alloc_ti(), and scsi_low_reset_nexus_target().

7.26.1.30 #define SCSI_LOW_DISK_WIDE (SCSI_LOW_DISK_WIDE_16 | SCSI_LOW_DISK_WIDE_32)

Definition at line 144 of file scsi_low.c.

Referenced by scsi_low_setup_done().

7.26.1.31 #define SCSI_LOW_DISK_WIDE_16 0x00020000

Definition at line 142 of file scsi_low.c.

Referenced by scsi_low_calcf_target(), and scsi_low_setup_done().

7.26.1.32 #define SCSI_LOW_DISK_WIDE_32 0x00040000

Definition at line 143 of file scsi_low.c.

Referenced by `scsi_low_calcf_target()`, and `scsi_low_setup_done()`.

7.26.1.33 `#define SCSI_LOW_DONE_COMPLETE 0`

Definition at line 133 of file `scsi_low.c`.

Referenced by `scsi_low_done()`, `scsi_low_revoke_ccb()`, and `scsi_low_setup_done()`.

7.26.1.34 `#define SCSI_LOW_DONE_RETRY 1`

Definition at line 134 of file `scsi_low.c`.

Referenced by `scsi_low_disconnected()`, `scsi_low_done()`, `scsi_low_msginfunc_lcc()`, and `scsi_low_setup_done()`.

7.26.1.35 `#define SCSI_LOW_FLAGS_QUIRKS_OK`

Definition at line 22 of file `scsi_low.c`.

7.26.1.36 `#define SCSI_LOW_INFO(slp, ti, s) printf("%s: %s\n", (slp) → sl_xname, (s))`

Definition at line 196 of file `scsi_low.c`.

Referenced by `scsi_low_cmd()`, `scsi_low_data()`, `scsi_low_msgfunc_identify()`, `scsi_low_msgin()`, and `scsi_low_msginfunc_simple_qtag()`.

7.26.1.37 `#define SCSI_LOW_INLINE static __inline`

Definition at line 243 of file `scsi_low.c`.

7.26.1.38 `#define SCSI_LOW_MAX_ATTEN_CHECK 32`

Definition at line 217 of file `scsi_low.c`.

Referenced by `scsi_low_test_atten()`.

7.26.1.39 `#define SCSI_LOW_MSG_ABORT_OK`

Value:

```
(SCSI_LOW_MSG_ABORT | \
                    SCSI_LOW_MSG_ABORT_QTAG | \
                    SCSI_LOW_MSG_CLEAR_QTAG | \
                    SCSI_LOW_MSG_TERMIO)
```

Referenced by `scsi_low_done()`.

7.26.1.40 `#define SCSI_LOW_NEGOTIATE_BEFORE_SENSE`

Definition at line 9 of file `scsi_low.c`.

7.26.1.41 #define SCSI_LOW_NEXUS_CHECK 0x0008

Definition at line 221 of file scsi_low.c.

Referenced by scsi_low_establish_ccb(), scsi_low_msgin(), and scsi_low_msginfunc_simple_qtag().

7.26.1.42 #define SCSI_LOW_POLL_HZ 1000

Definition at line 127 of file scsi_low.c.

Referenced by scsi_low_poll().

7.26.1.43 #define SCSI_LOW_QTAG_OK (SCSI_LOW_QTAG | SCSI_LOW_DISC)

Referenced by scsi_low_start().

7.26.1.44 #define SCSI_LOW_START_NO_QTAG 0

Definition at line 130 of file scsi_low.c.

Referenced by scsi_low_sense_abort_start(), and scsi_low_setup_start().

7.26.1.45 #define SCSI_LOW_START_QTAG 1

Definition at line 131 of file scsi_low.c.

Referenced by scsi_low_setup_start(), and scsi_low_start().

7.26.1.46 #define SCSI_LOW_START_UP_CHECK

Definition at line 10 of file scsi_low.c.

7.26.1.47 #define SCSI_LOW_STATICS

Definition at line 7 of file scsi_low.c.

7.26.1.48 #define SLSC_MODE_SENSE_SHORT 0x1a

Definition at line 2212 of file scsi_low.c.

7.26.1.49 #define TWIDDLEWAIT 10000

Definition at line 2906 of file scsi_low.c.

Referenced by scsi_low_bus_reset(), and scsi_low_twiddle_wait().

7.26.2 Function Documentation

7.26.2.1 `__FBSDID("$FreeBSD: src/sys/cam/scsi/scsi_low.c, v 1.26 2006/11/02 00:54:33 mjacob Exp $")`

7.26.2.2 `MALLOC_DEFINE(M_SCSILOW, "SCSI low", "SCSI low buffers")`

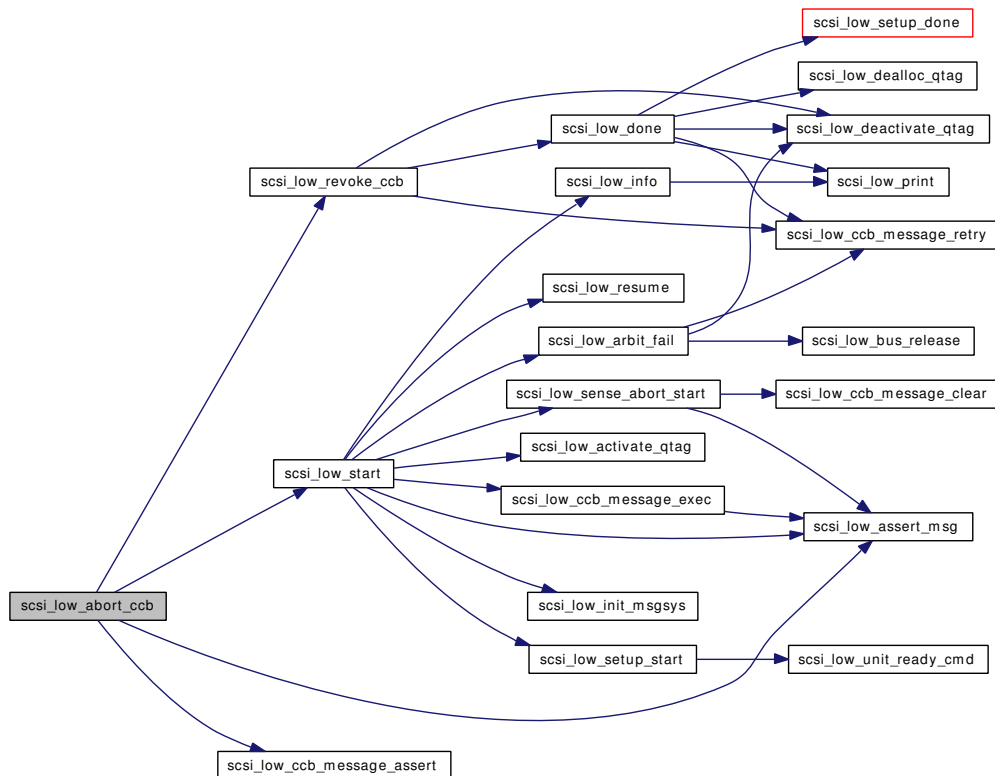
7.26.2.3 `static int scsi_low_abort_ccb (struct scsi_low_softc *, struct slccb *)` [static]

Definition at line 1971 of file `scsi_low.c`.

References `ABORTIO`, `CCB_DISCQ`, `slccb::ccb_error`, `slccb::ccb_flags`, `CCB_NORETRY`, `slccb::ccb_omsgoutflag`, `CCB_STARTQ`, `slccb::ccb_tag`, `slccb::li`, `scsi_low_assert_msg()`, `scsi_low_ccb_message_assert()`, `SCSI_LOW_MSG_ABORT`, `SCSI_LOW_MSG_ABORT_QTAG`, `scsi_low_revoke_ccb()`, `scsi_low_start()`, `SCSI_LOW_UNKTAG`, `slccb::ti`, and `ti`.

Referenced by `scsi_low_attach()`, and `scsi_low_test_abort()`.

Here is the call graph for this function:



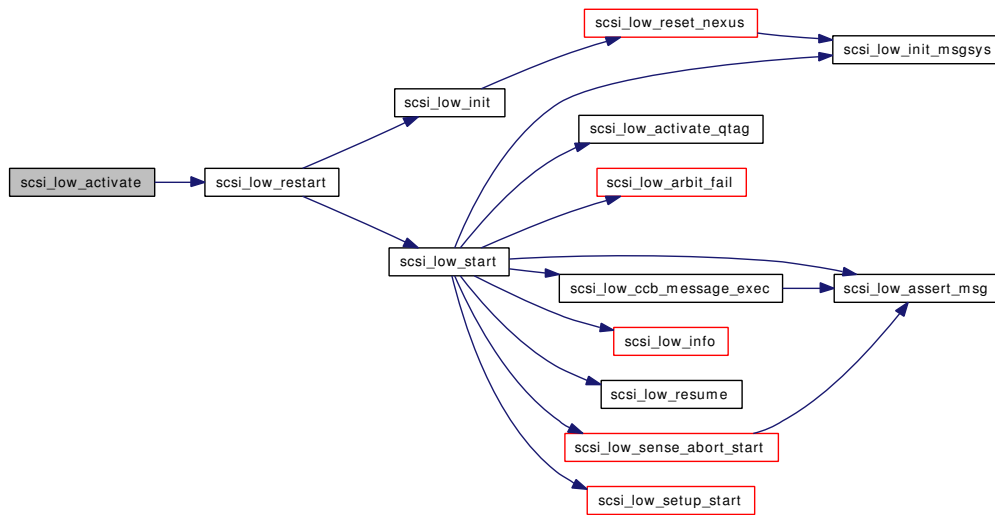
7.26.2.4 `int scsi_low_activate (struct scsi_low_softc * slp)`

Definition at line 1551 of file `scsi_low.c`.

References `HW_INACTIVE`, `scsi_low_restart()`, `SCSI_LOW_RESTART_HARD`, `SCSI_LOW_SPLSCSI`, `SCSI_LOW_TIMEOUT_CH_IO`, and `SCSI_LOW_TIMEOUT_START`.

Referenced by `scsi_low_activate_pisa()`.

Here is the call graph for this function:



7.26.2.5 SCSI_LOW_INLINE void scsi_low_activate_qtag (struct **slccb** *)

Definition at line 253 of file scsi_low.c.

References lun_info::li_nqio, and SCSI_LOW_UNKTAG.

Referenced by scsi_low_start().

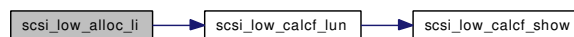
7.26.2.6 static struct **lun_info** * scsi_low_alloc_li (struct **targ_info** *, int, int) [static]

Definition at line 1700 of file scsi_low.c.

References lun_info::li_cfgflags, lun_info::li_discq, lun_info::li_diskflags, lun_info::li_flags_valid, lun_info::li_lun, lun_info::li_qtagbits, lun_info::li_quirks, lun_info::li_ti, SCSI_LOW_BZERO, scsi_low_calcf_lun(), SCSI_LOW_DISC, SCSI_LOW_DISK_LFLAGS, SCSI_LOW_INFO_ALLOC, SCSI_LOW_LINK, SCSI_LOW_LUN_FLAGS_QUIRKS_VALID, SCSI_LOW_LUN_FLAGS_USER_VALID, scsi_low_funcs::scsi_low_lun_init, SCSI_LOW_QTAG, SCSI_LOW_SYNC, scsi_low_softc::sl_funcs, and targ_info::ti_sc.

Referenced by scsi_low_attach(), scsi_low_find_ccb(), and scsi_low_msgin().

Here is the call graph for this function:



7.26.2.7 static int scsi_low_alloc_qtag (struct **slccb** *) [static]

Definition at line 4288 of file scsi_low.c.

References lun_info::li_qd, lun_info::li_qtagarray, lun_info::li_qtagbits, SCSI_LOW_MAXNEXUS, and SCSI_LOW_UNKTAG.

Referenced by `scsi_low_enqueue()`.

7.26.2.8 `static struct targ_info * scsi_low_alloc_ti (struct scsi_low_softc *, int)` [static]

Definition at line 1761 of file `scsi_low.c`.

References `SCSI_LOW_BUS_WIDTH_8`, `SCSI_LOW_BZERO`, `scsi_low_calcf_target()`, `SCSI_LOW_DISK_TFLAGS`, `SCSI_LOW_INFO_ALLOC`, `SCSI_LOW_TARG_FLAGS_QUIRKS_VALID`, `SCSI_LOW_TARG_FLAGS_USER_VALID`, `scsi_low_softc::sl_ti`, `ti`, `targ_info::ti_diskflags`, `targ_info::ti_flags_valid`, `targ_info::ti_id`, `targ_info::ti_litab`, `targ_info::ti_owidth`, `targ_info::ti_quirks`, and `targ_info::ti_sc`.

Referenced by `scsi_low_attach()`.

Here is the call graph for this function:



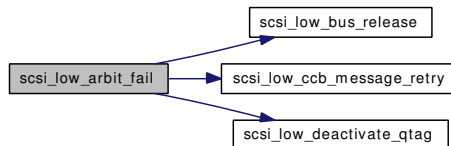
7.26.2.9 `void scsi_low_arbit_fail (struct scsi_low_softc * slp, struct slccb * cb)`

Definition at line 2485 of file `scsi_low.c`.

References `slccb::ccb_flags`, `slccb::ccb_selrcnt`, `CCB_STARTQ`, `scsi_low_bus_release()`, `scsi_low_ccb_message_retry()`, `scsi_low_deactivate_qtag()`, `ti`, and `slccb::ti`.

Referenced by `scsi_low_reselected()`, and `scsi_low_start()`.

Here is the call graph for this function:

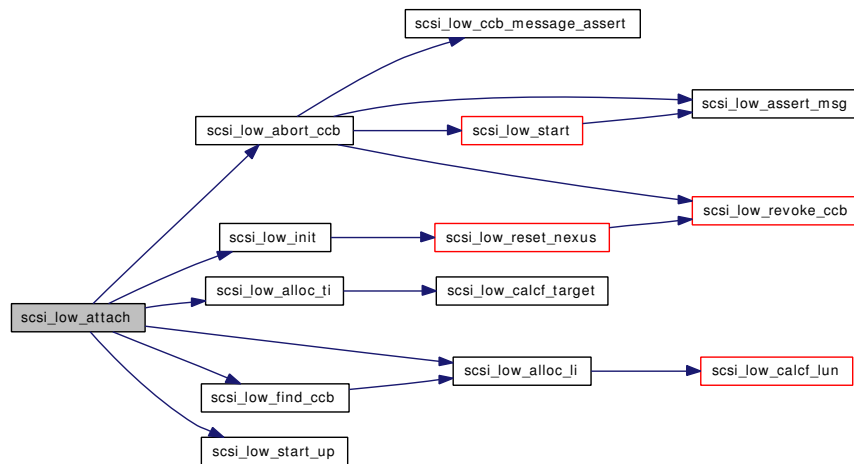


7.26.2.10 `int scsi_low_attach (struct scsi_low_softc * slp, int openings, int ntargs, int nluns, int targsize, int lunsiz)`

Definition at line 2023 of file `scsi_low.c`.

References `scsi_low_abort_ccb()`, `scsi_low_alloc_li()`, `scsi_low_alloc_ti()`, `SCSI_LOW_DELAY`, `scsi_low_find_ccb()`, `scsi_low_init()`, `SCSI_LOW_MAX_RETRY`, `SCSI_LOW_NCCB`, `SCSI_LOW_NTARGETS`, `SCSI_LOW_RESTART_HARD`, `SCSI_LOW_SPLSCSI`, `scsi_low_start_up()`, `SCSI_LOW_TIMEOUT_CH_IO`, `SCSI_LOW_TIMEOUT_START`, `sl_tab`, `ti`, and `targ_info::ti_lunsiz`.

Here is the call graph for this function:

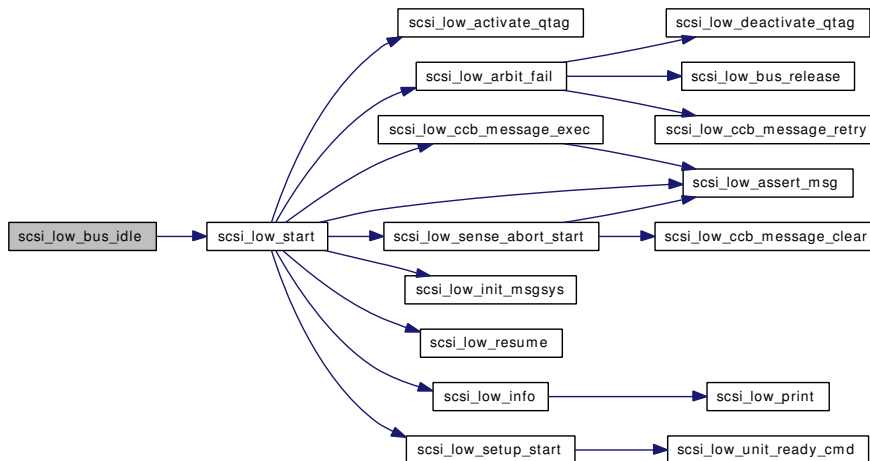


7.26.2.11 void scsi_low_bus_idle (struct [scsi_low_softc](#) * *slp*)

Definition at line 1832 of file `scsi_low.c`.

References `scsi_low_start()`.

Here is the call graph for this function:



7.26.2.12 static void scsi_low_bus_release (struct [scsi_low_softc](#) *, struct [targ_info](#) *) [static]

Definition at line 2509 of file `scsi_low.c`.

References `PH_DISC`, `PH_NULL`, `SCSI_LOW_RWUNK`, `SCSI_LOW_SETUP_PHASE`, `ti`, and `targ_info::ti_disc`.

Referenced by `scsi_low_arbit_fail()`, `scsi_low_disconnected()`, and `scsi_low_reset_nexus()`.

7.26.2.13 void scsi_low_bus_reset (struct [scsi_low_softc](#) * *slp*)

Definition at line 2919 of file `scsi_low.c`.

References `SCSI2_RESET_DELAY`, `scsi_low_twiddle_wait()`, and `TWIDDLEWAIT`.

Here is the call graph for this function:



7.26.2.14 static void scsi_low_calcf_lun (struct [lun_info](#) *) [static]

Definition at line 4449 of file `scsi_low.c`.

References `CFG_NODISC`, `CFG_NOPARITY`, `CFG_NOQTAG`, `targ_info::synch::offset`, `scsi_low_calcf_show()`, `SCSI_LOW_DEBUG_CALCF`, `SCSI_LOW_DEBUG_GO`, `SCSI_LOW_DISC`, `SCSI_LOW_DISK_LINK`, `SCSI_LOW_DISK_PARITY`, `SCSI_LOW_DISK_QTAG`, `SCSI_LOW_LINK`, `SCSI_LOW_LUN_FLAGS_ALL_VALID`, `SCSI_LOW_MAXNEXUS`, `SCSI_LOW_NOPARITY`, `SCSI_LOW_QTAG`, `SCSI_LOW_SYNC`, `scsi_low_softc::sl_cfgflags`, `ti`, `targ_info::ti_id`, `targ_info::ti_maxsynch`, and `targ_info::ti_sc`.

Referenced by `scsi_low_alloc_li()`, `scsi_low_errfunc_identify()`, `scsi_low_errfunc_qtag()`, `scsi_low_reset_nexus_target()`, and `scsi_low_setup_done()`.

Here is the call graph for this function:



7.26.2.15 static void scsi_low_calcf_show (struct [lun_info](#) *) [static]

Definition at line 4572 of file `scsi_low.c`.

References `targ_info::synch::offset`, `targ_info::synch::period`, `SCSI_LOW_BITS`, `scsi_low_softc::sl_xname`, `ti`, `targ_info::ti_id`, `targ_info::ti_maxsynch`, `targ_info::ti_sc`, and `targ_info::ti_width`.

Referenced by `scsi_low_calcf_lun()`, and `scsi_low_setup_done()`.

7.26.2.16 static void scsi_low_calcf_target (struct [targ_info](#) *) [static]

Definition at line 4513 of file `scsi_low.c`.

References `CFG_ASYNC`, `period`, `SCSI_LOW_BUS_WIDTH_16`, `SCSI_LOW_BUS_WIDTH_8`, `SCSI_LOW_DEBUG_CALCF`, `SCSI_LOW_DEBUG_GO`, `SCSI_LOW_DISK_SYNC`, `SCSI_LOW_DISK_WIDE_16`, `SCSI_LOW_DISK_WIDE_32`, `SCSI_LOW_MSG_SYNCH`, `SCSI_LOW_MSG_WIDE`, `SCSI_LOW_TARG_FLAGS_ALL_VALID`, `scsi_low_softc::sl_cfgflags`, `scsi_low_softc::sl_xname`, and `targ_info::ti_sc`.

Referenced by `scsi_low_alloc_ti()`, `scsi_low_reset_nexus_target()`, and `scsi_low_setup_done()`.

7.26.2.17 SCSI_LOW_INLINE void scsi_low_ccb_message_assert (struct [slccb](#) *, u_int)

Definition at line 289 of file `scsi_low.c`.

Referenced by `scsi_low_abort_ccb()`, `scsi_low_enqueue()`, and `scsi_low_establish_ccb()`.

7.26.2.18 SCSI_LOW_INLINE void scsi_low_ccb_message_clear (struct [slccb](#) *)

Definition at line 305 of file `scsi_low.c`.

Referenced by `scsi_low_sense_abort_start()`.

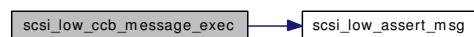
7.26.2.19 SCSI_LOW_INLINE void scsi_low_ccb_message_exec (struct [scsi_low_softc](#) *, struct [slccb](#) *)

Definition at line 279 of file `scsi_low.c`.

References `slccb::ccb_msgoutflag`, `scsi_low_assert_msg()`, and `slccb::ti`.

Referenced by `scsi_low_establish_ccb()`, and `scsi_low_start()`.

Here is the call graph for this function:

**7.26.2.20 SCSI_LOW_INLINE void scsi_low_ccb_message_retry (struct [slccb](#) *)**

Definition at line 298 of file `scsi_low.c`.

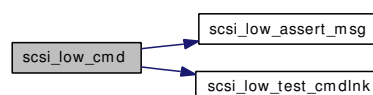
Referenced by `scsi_low_arbit_fail()`, `scsi_low_done()`, and `scsi_low_revoke_ccb()`.

7.26.2.21 int scsi_low_cmd (struct [scsi_low_softc](#) * *slp*, struct [targ_info](#) * *ti*)

Definition at line 3094 of file `scsi_low.c`.

References `FATALIO`, `scsi_low_assert_msg()`, `SCSI_LOW_CMDLNK_CHECK`, `SCSI_LOW_DEBUG_TEST_GO`, `SCSI_LOW_INFO`, `SCSI_LOW_MSG_ABORT`, `SCSI_LOW_READ`, `scsi_low_test_cmdlnk()`, `ti`, and `targ_info::ti_id`.

Here is the call graph for this function:

**7.26.2.22 int scsi_low_data (struct [scsi_low_softc](#) * *slp*, struct [targ_info](#) * *ti*, struct buf ** *bp*, int *direction*)**

Definition at line 3131 of file `scsi_low.c`.

References `slccb::bp`, `slccb::ccb_sscp`, `FATALIO`, `PDMAERR`, `sc_p::scp_direction`, `scsi_low_assert_msg()`, `SCSI_LOW_INFO`, `SCSI_LOW_MSG_ABORT`, `ti`, `targ_info::ti_ophase`, and `targ_info::ti_phase`.

Here is the call graph for this function:



7.26.2.23 `int scsi_low_deactivate (struct scsi_low_softc * slp)`

Definition at line 1535 of file `scsi_low.c`.

References `HW_INACTIVE`, `SCSI_LOW_SPLSCSI`, `SCSI_LOW_TIMEOUT_CH_ENGAGE`, `SCSI_LOW_TIMEOUT_CH_IO`, and `SCSI_LOW_TIMEOUT_STOP`.

Referenced by `scsi_low_deactivate_pisa()`, and `scsi_low_dettach()`.

7.26.2.24 `SCSI_LOW_INLINE void scsi_low_deactivate_qtag (struct slccb *)`

Definition at line 266 of file `scsi_low.c`.

References `lun_info::li_nqio`, and `SCSI_LOW_UNKTAG`.

Referenced by `scsi_low_arbit_fail()`, `scsi_low_done()`, `scsi_low_errfunc_qtag()`, and `scsi_low_revoke_ccb()`.

7.26.2.25 `static int scsi_low_dealloc_qtag (struct slccb *)` `[static]`

Definition at line 4325 of file `scsi_low.c`.

References `lun_info::li_qtagarray`, `lun_info::li_qtagbits`, `SCSI_LOW_MAXNEXUS`, and `SCSI_LOW_UNKTAG`.

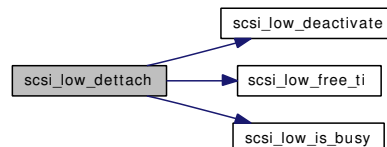
Referenced by `scsi_low_done()`, and `scsi_low_msginfunc_lcc()`.

7.26.2.26 `int scsi_low_dettach (struct scsi_low_softc * slp)`

Definition at line 2119 of file `scsi_low.c`.

References `scsi_low_deactivate()`, `scsi_low_free_ti()`, `scsi_low_is_busy()`, and `SCSI_LOW_SPLSCSI`.

Here is the call graph for this function:

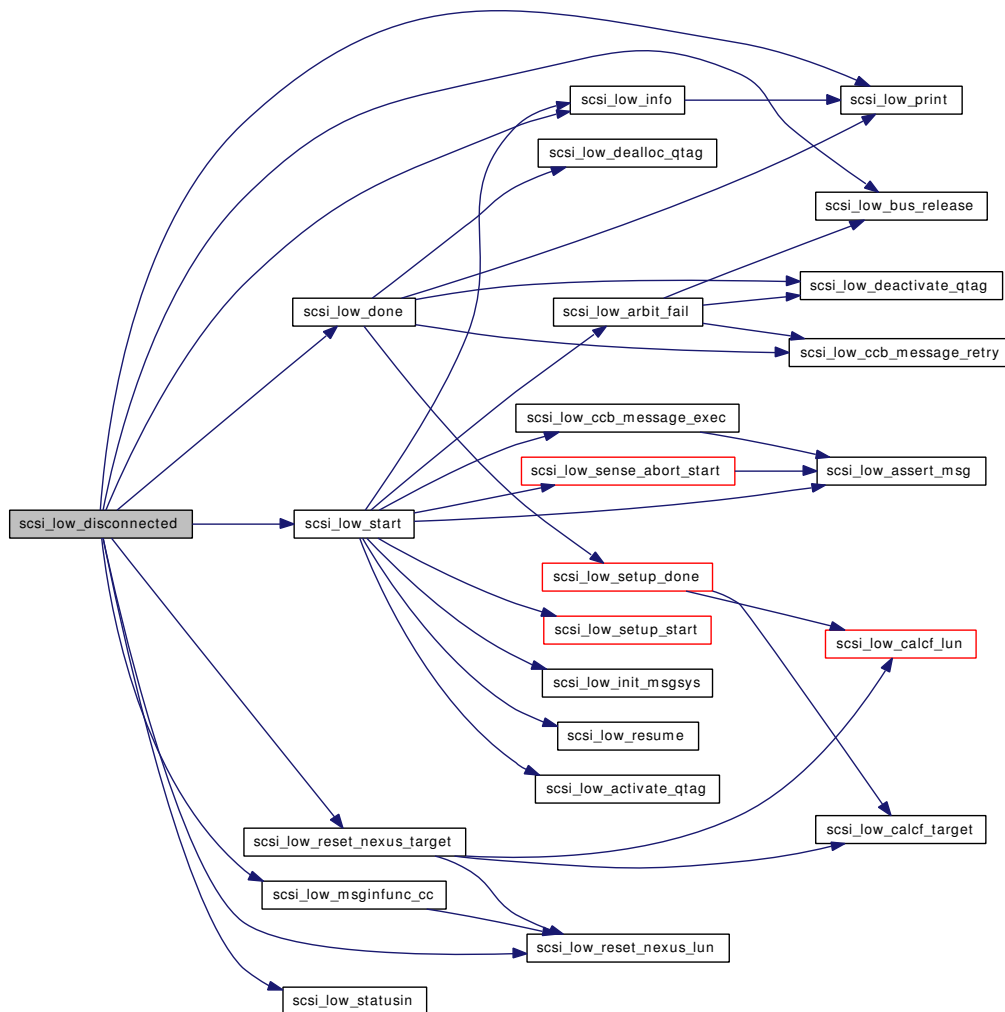


7.26.2.27 `int scsi_low_disconnected (struct scsi_low_softc * slp, struct targ_info * ti)`

Definition at line 4191 of file `scsi_low.c`.

References CCB_DISCQ, slccb::ccb_error, slccb::ccb_flags, slccb::ccb_msgoutflag, slccb::ccb_omsgoutflag, CCB_STARTQ, FATALIO, slccb::li, lun_info::li_disc, lun_info::li_discq, MSGPH_ABORT, MSGPH_CMDC, MSGPH_DISC, MSGPH_LCTERM, MSGPH_NULL, MSGPH_RESET, MSGPH_TERM, PH_SELSTART, SCSI_LOW_ATTEN_CHECK, scsi_low_bus_release(), SCSI_LOW_DEBUG_DISC, SCSI_LOW_DEBUG_GO, SCSI_LOW_DEBUG_TEST_GO, scsi_low_done(), SCSI_LOW_DONE_RETRY, scsi_low_info(), SCSI_LOW_MSG_NOOP, scsi_low_msginfunc_cc(), scsi_low_print(), scsi_low_reset_nexus_lun(), scsi_low_reset_nexus_target(), scsi_low_start(), scsi_low_statusin(), SELTIMEOUTIO, ST_GOOD, ti, targ_info::ti_disc, targ_info::ti_id, targ_info::ti_msgflags, targ_info::ti_phase, and UBFERR.

Here is the call graph for this function:



7.26.2.28 static int scsi_low_done (struct scsi_low_softc *, struct slccb *) [static]

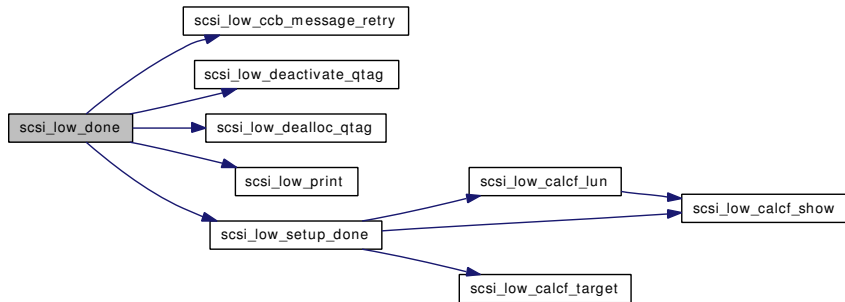
Definition at line 2661 of file scsi_low.c.

References ABORTIO, BUSYERR, CCB_AUTOSENSE, CCB_CLEARQ, slccb::ccb_dataalen, slccb::ccb_error, slccb::ccb_flags, CCB_INTERNAL, CCB_NORETRY, slccb::ccb_omsgoutflag,

slccb::ccb_rcnt, slccb::ccb_scp, CCB_SENSE, slccb::ccb_sscp, FATALIO, slccb::osdep, PDMAERR, sc_p::scp_cmd, sc_p::scp_cmdlen, sc_p::scp_datalen, sc_p::scp_status, scsi_low_ccb_message_retry(), SCSI_LOW_CMD_RESIDUAL_CHK, scsi_low_deactivate_qtag(), scsi_low_dealloc_qtag(), SCSI_LOW_DEBUG_DONE, SCSI_LOW_DEBUG_GO, SCSI_LOW_DONE_COMPLETE, SCSI_LOW_DONE_RETRY, SCSI_LOW_MSG_ABORT_OK, scsi_low_print(), scsi_low_setup_done(), SENSEERR, SENSEIO, ST_BUSY, ST_CHKCOND, ST_CMDTERM, ST_CONFLICT, ST_GOOD, ST_INTERGOOD, ST_INTERMET, ST_MET, ST_QUEFULL, ST_UNKNOWN, STATERR, slccb::ti, targ_info::ti_id, and UACAERR.

Referenced by `scsi_low_disconnected()`, `scsi_low_msginfunc_lcc()`, and `scsi_low_revoke_ccb()`.

Here is the call graph for this function:

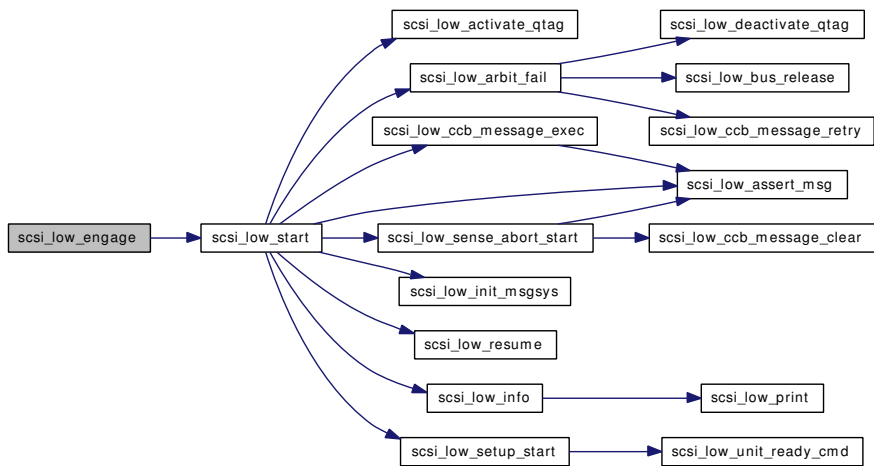


7.26.2.29 static void scsi_low_engage (void *) [static]

Definition at line 1632 of file `scsi_low.c`.

References `HW_RESUME`, `SCSI_LOW_ENGAGE`, `scsi_low_osdep_funcs::scsi_low_osdep_timeout`, `scsi_low_funcs::scsi_low_power`, `SCSI_LOW_SPLSCSI`, `scsi_low_start()`, `SCSI_LOW_TIMEOUT_CH_ENGAGE`, `SCSI_LOW_TIMEOUT_START`, `scsi_low_softc::sl_flags`, `scsi_low_softc::sl_funcs`, `scsi_low_softc::sl_osdep_fp`, and `scsi_low_softc::sl_rstep`.

Here is the call graph for this function:



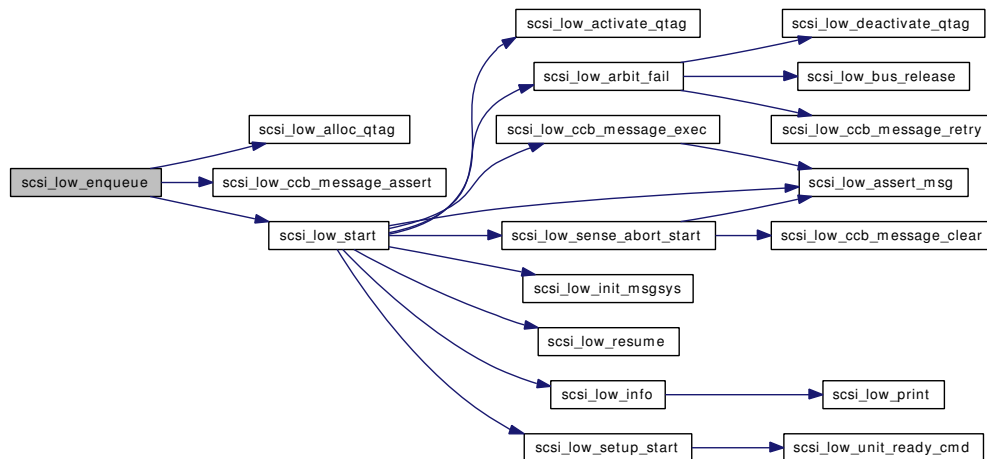
7.26.2.30 static int scsi_low_enqueue (struct [scsi_low_softc](#) *, struct [targ_info](#) *, struct [lun_info](#) *, struct [slccb](#) *, u_int, u_int) [static]

Definition at line 2150 of file [scsi_low.c](#).

References [slccb::ccb_error](#), [slccb::ccb_flags](#), [slccb::ccb_otag](#), [CCB_STARTQ](#), [slccb::ccb_tag](#), [slccb::ccb_tc](#), [slccb::ccb_tmax](#), [CCB_URGENT](#), [slccb::li](#), [PENDINGIO](#), [scsi_low_alloc_qtag\(\)](#), [scsi_low_ccb_message_assert\(\)](#), [SCSI_LOW_MIN_TOUT](#), [scsi_low_start\(\)](#), [SCSI_LOW_UNKTAG](#), [ti](#), and [slccb::ti](#).

Referenced by [scsi_low_message_enqueue\(\)](#).

Here is the call graph for this function:

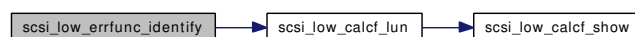


7.26.2.31 static int scsi_low_errfunc_identify (struct [scsi_low_softc](#) *, u_int) [static]

Definition at line 3409 of file [scsi_low.c](#).

References [scsi_low_calcf_lun\(\)](#), and [SCSI_LOW_DISC](#).

Here is the call graph for this function:

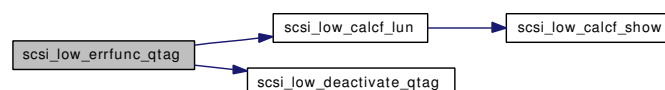


7.26.2.32 static int scsi_low_errfunc_qtag (struct [scsi_low_softc](#) *, u_int) [static]

Definition at line 3448 of file [scsi_low.c](#).

References [scsi_low_calcf_lun\(\)](#), [scsi_low_deactivate_qtag\(\)](#), [SCSI_LOW_MSG_REJECT](#), and [SCSI_LOW_QTAG](#).

Here is the call graph for this function:

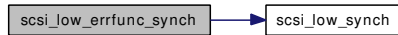


7.26.2.33 `static int scsi_low_errfunc_synch (struct scsi_low_softc *, u_int)` [static]

Definition at line 3423 of file `scsi_low.c`.

References `MSGIN_OFFSET`, `MSGIN_PERIOD`, `scsi_low_synch()`, and `ti`.

Here is the call graph for this function:

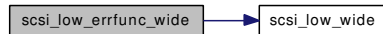


7.26.2.34 `static int scsi_low_errfunc_wide (struct scsi_low_softc *, u_int)` [static]

Definition at line 3436 of file `scsi_low.c`.

References `MSGIN_WIDTHHP`, `scsi_low_wide()`, and `ti`.

Here is the call graph for this function:



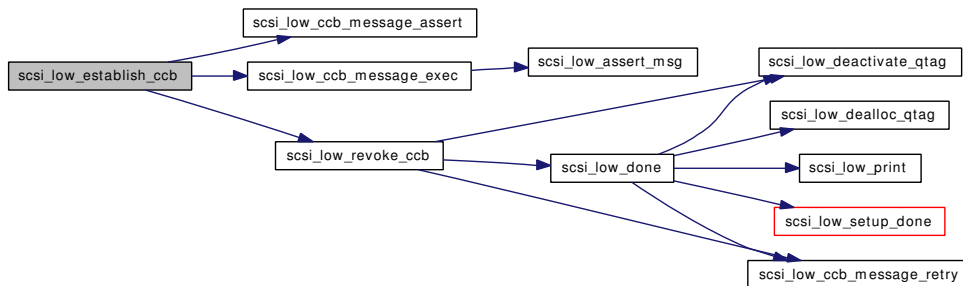
7.26.2.35 `static struct slccb * scsi_low_establish_ccb (struct targ_info *, struct lun_info *, scsi_low_tag_t)` [static]

Definition at line 2957 of file `scsi_low.c`.

References `CCB_DISCQ`, `slccb::ccb_error`, `slccb::ccb_flags`, `slccb::ccb_msgoutflag`, `CCB_NORETRY`, `slccb::ccb_omsgoutflag`, `CCB_SILENT`, `slccb::ccb_sscp`, `slccb::ccb_tag`, `slccb::li`, `lun_info::li_disc`, `lun_info::li_discq`, `SCSI_LOW_ATTEN_CHECK`, `scsi_low_ccb_message_assert()`, `scsi_low_ccb_message_exec()`, `SCSI_LOW_DEBUG_TEST_GO`, `scsi_low_funes::scsi_low_establish_ccb_nexus`, `SCSI_LOW_MSG_NOOP`, `SCSI_LOW_NEXUS_CHECK`, `scsi_low_revoke_ccb()`, `scsi_low_softc::sl_disc`, `scsi_low_softc::sl_error`, `scsi_low_softc::sl_funes`, `scsi_low_softc::sl_Qnexus`, `scsi_low_softc::sl_scp`, and `scsi_low_softc::sl_xname`.

Referenced by `scsi_low_msgin()`, and `scsi_low_msginfunc_simple_qtag()`.

Here is the call graph for this function:



7.26.2.36 `static struct slccb * scsi_low_find_ccb (struct scsi_low_softc *, u_int, u_int, void *)`
`[static]`

Definition at line 348 of file `scsi_low.c`.

References `slccb::li`, `lun_info::li_discq`, `slccb::osdep`, `scsi_low_alloc_li()`, and `ti`.

Referenced by `scsi_low_attach()`.

Here is the call graph for this function:



7.26.2.37 `static void scsi_low_free_ti (struct scsi_low_softc *)` `[static]`

Definition at line 1799 of file `scsi_low.c`.

References `SCSI_LOW_INFO_DEALLOC`, `ti`, and `targ_info::ti_litab`.

Referenced by `scsi_low_dettach()`.

7.26.2.38 `void scsi_low_info (struct scsi_low_softc *, struct targ_info *, u_char *)`

Definition at line 4732 of file `scsi_low.c`.

References `scsi_low_print()`, `sl_tab`, and `ti`.

Referenced by `scsi_low_disconnected()`, `scsi_low_start()`, and `scsi_low_timeout_check()`.

Here is the call graph for this function:



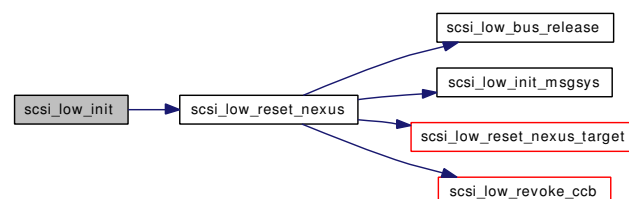
7.26.2.39 `static int scsi_low_init (struct scsi_low_softc *, u_int)` `[static]`

Definition at line 1660 of file `scsi_low.c`.

References `HW_INACTIVE`, `HW_INITIALIZING`, `HW_POWDOWN`, `HW_POWERCTRL`, `HW_RESUME`, `SCSI_LOW_POWDOWN_TC`, `scsi_low_reset_nexus()`, `SCSI_LOW_RESTART_SOFT`, `SCSI_LOW_TIMEOUT_CH_ENGAGE`, and `SCSI_LOW_TIMEOUT_STOP`.

Referenced by `scsi_low_attach()`, `scsi_low_restart()`, and `scsi_low_timeout_check()`.

Here is the call graph for this function:



7.26.2.40 SCSI_LOW_INLINE void scsi_low_init_msgsys (struct [scsi_low_softc](#) *, struct [targ_info](#) *)

Definition at line 312 of file [scsi_low.c](#).

References [MSGPH_NULL](#), [SCSI_LOW_DEASSERT_ATN](#), [SCSI_LOW_SETUP_MSGPHASE](#), [ti](#), [targ_info::ti_emsgflags](#), [targ_info::ti_msgflags](#), [targ_info::ti_msginptr](#), and [targ_info::ti_omsgflags](#).

Referenced by [scsi_low_msginfunc_lcc\(\)](#), [scsi_low_reselected\(\)](#), [scsi_low_reset_nexus\(\)](#), and [scsi_low_start\(\)](#).

7.26.2.41 int scsi_low_is_busy (struct [scsi_low_softc](#) * *slp*)

Definition at line 1525 of file [scsi_low.c](#).

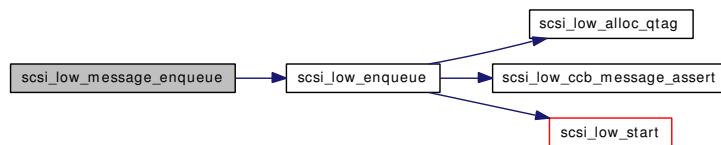
Referenced by [scsi_low_dettach\(\)](#).

7.26.2.42 static int scsi_low_message_enqueue (struct [scsi_low_softc](#) *, struct [targ_info](#) *, struct [lun_info](#) *, u_int) [static]

Definition at line 2187 of file [scsi_low.c](#).

References [slccb::bp](#), [CCB_NORETRY](#), [slccb::li](#), [slccb::osdep](#), [scsi_low_enqueue\(\)](#), [ti](#), and [targ_info::ti_setup_msg](#).

Here is the call graph for this function:



7.26.2.43 static int scsi_low_msgfunc_abort (struct [scsi_low_softc](#) *) [static]

Definition at line 3354 of file [scsi_low.c](#).

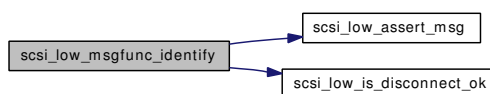
References [MSGPH_ABORT](#), and [SCSI_LOW_SETUP_MSGPHASE](#).

7.26.2.44 static int scsi_low_msgfunc_identify (struct [scsi_low_softc](#) *) [static]

Definition at line 3317 of file [scsi_low.c](#).

References [slccb::ccb_tag](#), [FATALIO](#), [slccb::li](#), [lun_info::li_lun](#), [MSG_IDENTIFY](#), [MSG_IDENTIFY_DISCRIV](#), [PH_MSGOUT](#), [scsi_low_assert_msg\(\)](#), [SCSI_LOW_INFO](#), [scsi_low_is_disconnect_ok\(\)](#), [SCSI_LOW_MSG_ABORT](#), [SCSI_LOW_UNKTAG](#), [ti](#), [targ_info::ti_msgoutlen](#), [targ_info::ti_msgoutstr](#), and [targ_info::ti_phase](#).

Here is the call graph for this function:



7.26.2.45 `static int scsi_low_msgfunc_qabort (struct scsi_low_softc *)` [static]

Definition at line 3363 of file `scsi_low.c`.

References `MSGPH_TERM`, and `SCSI_LOW_SETUP_MSGPHASE`.

7.26.2.46 `static int scsi_low_msgfunc_qtag (struct scsi_low_softc *)` [static]

Definition at line 3381 of file `scsi_low.c`.

References `slccb::ccb_tag`, `MSG_NOOP`, `PH_MSGOUT`, `SCSI_LOW_UNKTAG`, `ti`, `targ_info::ti_msgoutlen`, `targ_info::ti_msgoutstr`, and `targ_info::ti_phase`.

7.26.2.47 `static int scsi_low_msgfunc_reset (struct scsi_low_softc *)` [static]

Definition at line 3372 of file `scsi_low.c`.

References `MSGPH_RESET`, and `SCSI_LOW_SETUP_MSGPHASE`.

7.26.2.48 `static int scsi_low_msgfunc_synch (struct scsi_low_softc *)` [static]

Definition at line 3290 of file `scsi_low.c`.

References `MSG_EXTEND_SYNCHCODE`, `MSG_EXTEND_SYNCHLEN`, `targ_info::synch::offset`, `targ_info::synch::period`, `ti`, `targ_info::ti_maxsynch`, `targ_info::ti_msgoutlen`, and `targ_info::ti_msgoutstr`.

7.26.2.49 `static int scsi_low_msgfunc_wide (struct scsi_low_softc *)` [static]

Definition at line 3304 of file `scsi_low.c`.

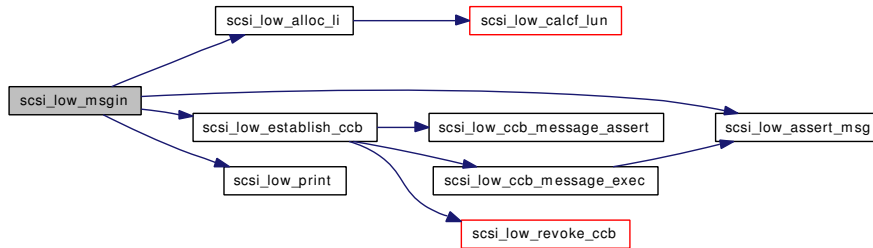
References `MSG_EXTEND_WIDECODE`, `MSG_EXTEND_WIDELLEN`, `ti`, `targ_info::ti_msgoutlen`, `targ_info::ti_msgoutstr`, and `targ_info::ti_width`.

7.26.2.50 `int scsi_low_msgin (struct scsi_low_softc * slp, struct targ_info * ti, u_int c)`

Definition at line 4047 of file `scsi_low.c`.

References `FATALIO`, `lun_info::li_lun`, `lun_info::li_nqio`, `scsi_low_msgin_data::md_len`, `scsi_low_msgin_data::md_msgfunc`, `MSG_IDENTIFY`, `MSGCMD_LUN`, `MSGIN_DATA_LAST`, `MSGINPTR_CLR`, `scsi_low_alloc_li()`, `scsi_low_assert_msg()`, `SCSI_LOW_DATA_PE`, `SCSI_LOW_DEBUG_TEST_GO`, `scsi_low_establish_ccb()`, `SCSI_LOW_INFO`, `SCSI_LOW_MAX_MSGLEN`, `SCSI_LOW_MAX_PHCHANGES`, `SCSI_LOW_MSG_ABORT`, `SCSI_LOW_MSG_PARITY`, `SCSI_LOW_MSG_REJECT`, `SCSI_LOW_NEXUS_CHECK`, `scsi_low_print()`, `SCSI_LOW_UNKTAG`, `ti`, `targ_info::ti_id`, `targ_info::ti_log_msgin`, `targ_info::ti_msgin`, `targ_info::ti_msgin_parity_error`, `targ_info::ti_msginlen`, `targ_info::ti_msginptr`, `targ_info::ti_ophase`, and `targ_info::ti_phase`.

Here is the call graph for this function:



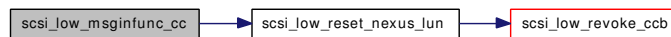
7.26.2.51 static int scsi_low_msginfunc_cc (struct [scsi_low_softc](#) *) [static]

Definition at line 3596 of file `scsi_low.c`.

References `lun_info::li_maxnexus`, `lun_info::li_maxnqio`, `lun_info::li_nqio`, `lun_info::li_qflags`, `MSGERR`, `MSGPH_CMDDC`, `SCSI_LOW_QFLAG_CA_QCLEAR`, `scsi_low_reset_nexus_lun()`, `SCSI_LOW_SETUP_MSGPHASE`, `ST_BUSY`, `ST_CHKCOND`, `ST_GOOD`, `ST_INTERGOOD`, `ST_INTERMET`, and `ST_QUEFULL`.

Referenced by `scsi_low_disconnected()`.

Here is the call graph for this function:



7.26.2.52 static int scsi_low_msginfunc_disc (struct [scsi_low_softc](#) *) [static]

Definition at line 3729 of file `scsi_low.c`.

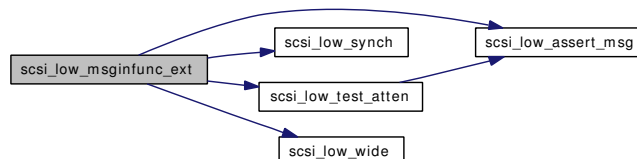
References `MSGPH_DISC`, and `SCSI_LOW_SETUP_MSGPHASE`.

7.26.2.53 static int scsi_low_msginfunc_ext (struct [scsi_low_softc](#) *) [static]

Definition at line 3936 of file `scsi_low.c`.

References `slccb::ccb_scp`, `MKMSG_EXTEND`, `MSG_EXTEND_MDPCODE`, `MSG_EXTEND_MDPLEN`, `MSG_EXTEND_SYNCHCODE`, `MSG_EXTEND_SYNCHLEN`, `MSG_EXTEND_WIDECODE`, `MSG_EXTEND_WIDELLEN`, `sc_p::scp_dataalen`, `scsi_low_assert_msg()`, `SCSI_LOW_ATTEN_CHECK`, `SCSI_LOW_DEBUG_TEST_GO`, `SCSI_LOW_MSG_REJECT`, `SCSI_LOW_MSG_SYNCH`, `SCSI_LOW_MSG_WIDE`, `scsi_low_synch()`, `scsi_low_test_atten()`, `scsi_low_wide()`, `ti`, `targ_info::ti_emsgflags`, `targ_info::ti_id`, `targ_info::ti_msgin`, `targ_info::ti_msginlen`, and `targ_info::ti_msginptr`.

Here is the call graph for this function:

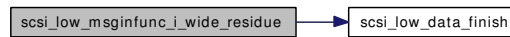


7.26.2.54 `static int scsi_low_msginfunc_i_wide_residue (struct scsi_low_softc *)` [static]

Definition at line 3914 of file `scsi_low.c`.

References `slccb::ccb_scp`, `sc_p::scp_dataalen`, `SCSI_LOW_BUS_WIDTH_16`, `SCSI_LOW_BUS_WIDTH_32`, `scsi_low_data_finish()`, `ti`, `targ_info::ti_msgin`, and `targ_info::ti_width`.

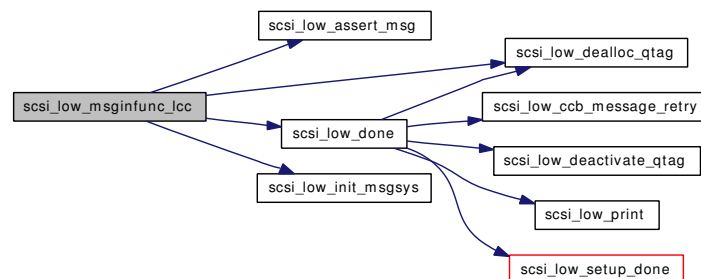
Here is the call graph for this function:

**7.26.2.55** `static int scsi_low_msginfunc_lcc (struct scsi_low_softc *)` [static]

Definition at line 3643 of file `scsi_low.c`.

References `slccb::ccb_dataalen`, `slccb::ccb_error`, `slccb::ccb_flags`, `CCB_INTERNAL`, `slccb::ccb_otag`, `slccb::ccb_scp`, `slccb::ccb_sscp`, `CCB_STARTQ`, `slccb::ccb_tag`, `slccb::ccb_tc`, `slccb::ccb_temax`, `slccb::li`, `lun_info::li_flags`, `lun_info::li_maxnexus`, `lun_info::li_maxnqio`, `MSGERR`, `MSGPH_LCTERM`, `sc_p::scp_status`, `scsi_low_assert_msg()`, `scsi_low_dealloc_qtag()`, `scsi_low_done()`, `SCSI_LOW_DONE_RETRY`, `scsi_low_init_msgsys()`, `SCSI_LOW_LINK`, `SCSI_LOW_MIN_TOUT`, `SCSI_LOW_MSG_REJECT`, `SCSI_LOW_SETUP_MSGPHASE`, `SCSI_LOW_UNKTAG`, `ST_INTERGOOD`, `ST_INTERMET`, `ST_UNKNOWN`, `ti`, `targ_info::ti_log_msgin`, and `targ_info::ti_log_msgout`.

Here is the call graph for this function:

**7.26.2.56** `static int scsi_low_msginfunc_msg_reject (struct scsi_low_softc *)` [static]

Definition at line 4013 of file `scsi_low.c`.

References `scsi_low_msgout_data::md_errfunc`, `scsi_low_msgout_data::md_flags`, `SCSI_LOW_MSG_ALL`, `SCSI_LOW_MSG_REJECT`, `ti`, and `targ_info::ti_emsgflags`.

7.26.2.57 `static int scsi_low_msginfunc_noop (struct scsi_low_softc *)` [static]

Definition at line 3576 of file `scsi_low.c`.

7.26.2.58 `static int scsi_low_msginfunc_parity (struct scsi_low_softc *)` [static]

Definition at line 4002 of file `scsi_low.c`.

References `scsi_low_assert_msg()`, `SCSI_LOW_MSG_REJECT`, and `ti`.

Here is the call graph for this function:



7.26.2.59 `static int scsi_low_msginfunc_rejop (struct scsi_low_softc *) [static]`

Definition at line 3584 of file `scsi_low.c`.

References `scsi_low_assert_msg()`, `SCSI_LOW_MSG_REJECT`, `ti`, and `targ_info::ti_msgin`.

Here is the call graph for this function:



7.26.2.60 `static int scsi_low_msginfunc_rp (struct scsi_low_softc *) [static]`

Definition at line 3754 of file `scsi_low.c`.

References `scsi_low_assert_msg()`, and `SCSI_LOW_MSG_REJECT`.

Here is the call graph for this function:



7.26.2.61 `static int scsi_low_msginfunc_sdp (struct scsi_low_softc *) [static]`

Definition at line 3738 of file `scsi_low.c`.

References `slccb::ccb_sscp`, `sc_p::scp_data`, `sc_p::scp_dataalen`, `scsi_low_assert_msg()`, and `SCSI_LOW_MSG_REJECT`.

Here is the call graph for this function:

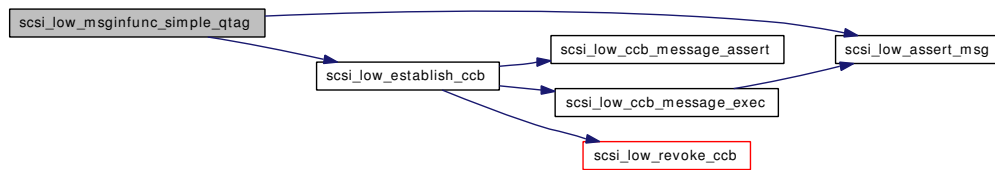


7.26.2.62 `static int scsi_low_msginfunc_simple_qtag (struct scsi_low_softc *) [static]`

Definition at line 3884 of file `scsi_low.c`.

References `FATALIO`, `scsi_low_assert_msg()`, `SCSI_LOW_DEBUG_TEST_GO`, `scsi_low_establish_ccb()`, `SCSI_LOW_INFO`, `SCSI_LOW_MSG_ABORT`, `SCSI_LOW_MSG_ABORT_QTAG`, `SCSI_LOW_NEXUS_CHECK`, `ti`, `targ_info::ti_id`, and `targ_info::ti_msgin`.

Here is the call graph for this function:

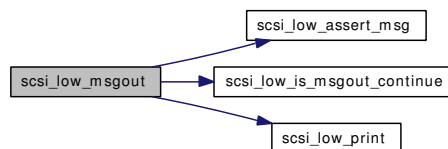


7.26.2.63 `int scsi_low_msgout(struct scsi_low_softc *slp, struct targ_info *ti, u_int fl)`

Definition at line 3471 of file `scsi_low.c`.

References `FATALIO`, `scsi_low_msgout_data::md_condition`, `scsi_low_msgout_data::md_flags`, `scsi_low_msgout_data::md_msg`, `scsi_low_msgout_data::md_msgfunc`, `MSG_RELEASE_ATN`, `scsi_low_assert_msg()`, `SCSI_LOW_DIAGNOSTIC`, `scsi_low_is_msgout_continue()`, `SCSI_LOW_MAX_MSGLEN`, `SCSI_LOW_MAX_PHCHANGES`, `SCSI_LOW_MSG_ABORT`, `SCSI_LOW_MSG_ALL`, `SCSI_LOW_MSG_NOOP`, `scsi_low_msgout_data`, `SCSI_LOW_MSGOUT_INIT`, `SCSI_LOW_MSGOUT_UNIFY`, `scsi_low_print()`, `ti`, `targ_info::ti_emsgflags`, `targ_info::ti_log_msgout`, `targ_info::ti_msgflags`, `targ_info::ti_msgoutlen`, `targ_info::ti_msgoutstr`, and `targ_info::ti_omsgflags`.

Here is the call graph for this function:

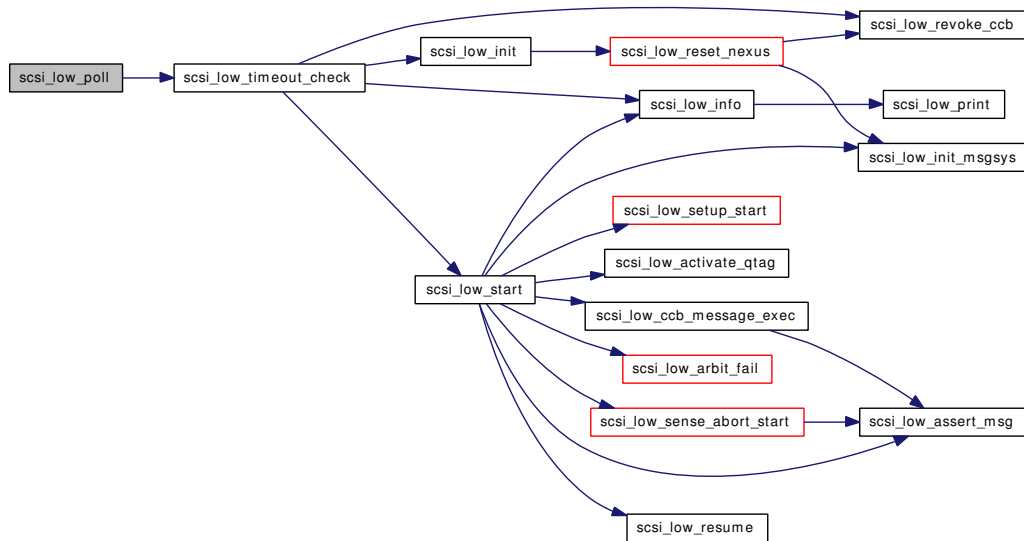


7.26.2.64 `static int scsi_low_poll(struct scsi_low_softc *, struct slecb *)` `[static]`

Definition at line 4655 of file `scsi_low.c`.

References `SCSI_LOW_DELAY`, `SCSI_LOW_POLL_HZ`, `scsi_low_timeout_check()`, and `SCSI_LOW_TIMEOUT_HZ`.

Here is the call graph for this function:



7.26.2.65 void scsi_low_print (struct scsi_low_softc * slp, struct targ_info * ti)

Definition at line 4765 of file scsi_low.c.

References lun_info::li_discq, sc_p::scp_data, sc_p::scp_dataLen, sc_p::scp_status, SCSI_LOW_BITS, SCSI_LOW_ERRORBITS, ti, and targ_info::ti_litab.

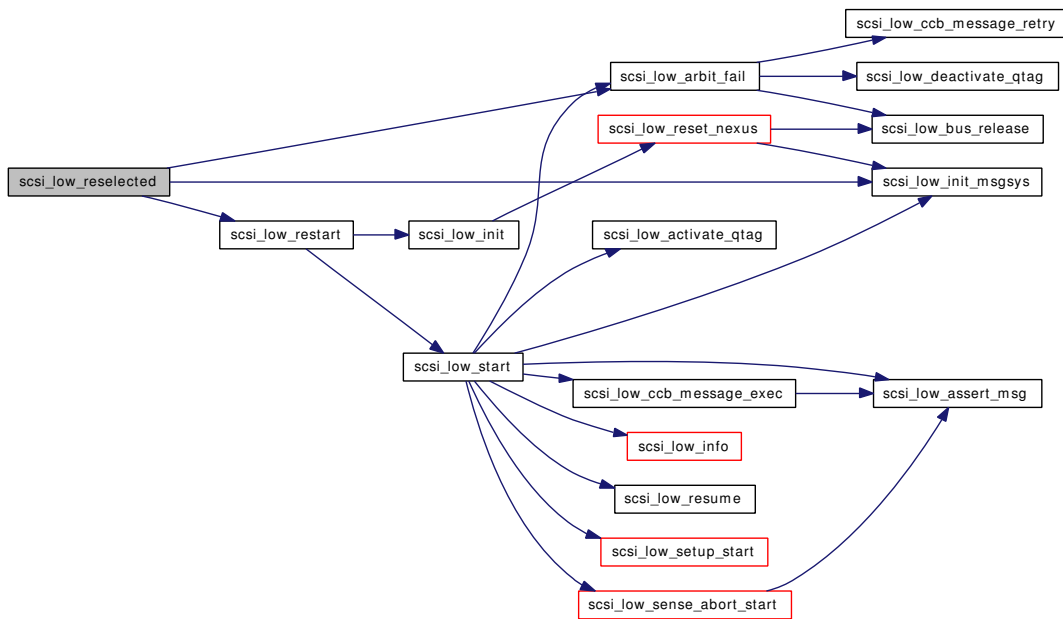
Referenced by scsi_low_disconnected(), scsi_low_done(), scsi_low_info(), scsi_low_msgin(), and scsi_low_msgout().

7.26.2.66 struct targ_info* scsi_low_reselected (struct scsi_low_softc * slp, u_int targ)

Definition at line 3019 of file scsi_low.c.

References PH_DISC, PH_NULL, PH_RESEL, scsi_low_arbit_fail(), scsi_low_init_msgsys(), scsi_low_restart(), SCSI_LOW_RESTART_HARD, SCSI_LOW_SETUP_PHASE, ti, and targ_info::ti_phase.

Here is the call graph for this function:



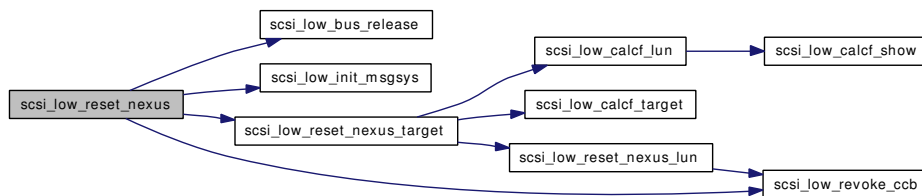
7.26.2.67 static void scsi_low_reset_nexus (struct [scsi_low_softc](#) *, int) [static]

Definition at line 2868 of file `scsi_low.c`.

References `slccb::ccb_flags`, `CCB_STARTQ`, `HW_PDMASRT`, `scsi_low_bus_release()`, `scsi_low_init_msgsys()`, `scsi_low_reset_nexus_target()`, `scsi_low_revoke_ccb()`, and `ti`.

Referenced by `scsi_low_init()`.

Here is the call graph for this function:



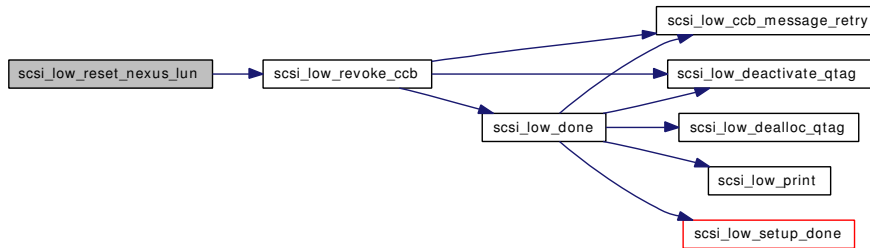
7.26.2.68 static void scsi_low_reset_nexus_lun (struct [scsi_low_softc](#) *, struct [lun_info](#) *, int) [static]

Definition at line 4409 of file `scsi_low.c`.

References `slccb::ccb_flags`, `CCB_STARTQ`, `slccb::li`, `lun_info::li_discq`, and `scsi_low_revoke_ccb()`.

Referenced by `scsi_low_disconnected()`, `scsi_low_msginfunc_cc()`, and `scsi_low_reset_nexus_target()`.

Here is the call graph for this function:



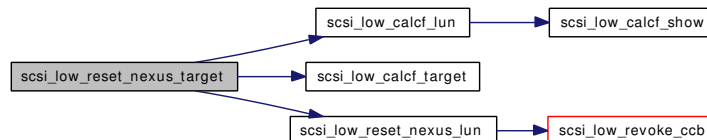
7.26.2.69 static void scsi_low_reset_nexus_target (struct [scsi_low_softc](#) *, struct [targ_info](#) *, int) [static]

Definition at line 2818 of file [scsi_low.c](#).

References [lun_info::li_diskflags](#), [lun_info::li_flags](#), [lun_info::li_flags_valid](#), [lun_info::li_maxnqio](#), [lun_info::li_state](#), [targ_info::synch::offset](#), [targ_info::synch::period](#), [SCSI_LOW_BUS_WIDTH_8](#), [scsi_low_calcf_lun\(\)](#), [scsi_low_calcf_target\(\)](#), [SCSI_LOW_DISK_LFLAGS](#), [SCSI_LOW_DISK_TFLAGS](#), [SCSI_LOW_INFO_REVOKE](#), [SCSI_LOW_LUN_FLAGS_DISK_VALID](#), [SCSI_LOW_LUN_SLEEP](#), [scsi_low_reset_nexus_lun\(\)](#), [SCSI_LOW_TARG_FLAGS_DISK_VALID](#), [ti](#), [targ_info::ti_disc](#), [targ_info::ti_diskflags](#), [targ_info::ti_flags_valid](#), [targ_info::ti_litab](#), [targ_info::ti_osynch](#), [targ_info::ti_owidth](#), [targ_info::ti_setup_msg](#), and [targ_info::ti_setup_msg_done](#).

Referenced by [scsi_low_disconnected\(\)](#), and [scsi_low_reset_nexus\(\)](#).

Here is the call graph for this function:



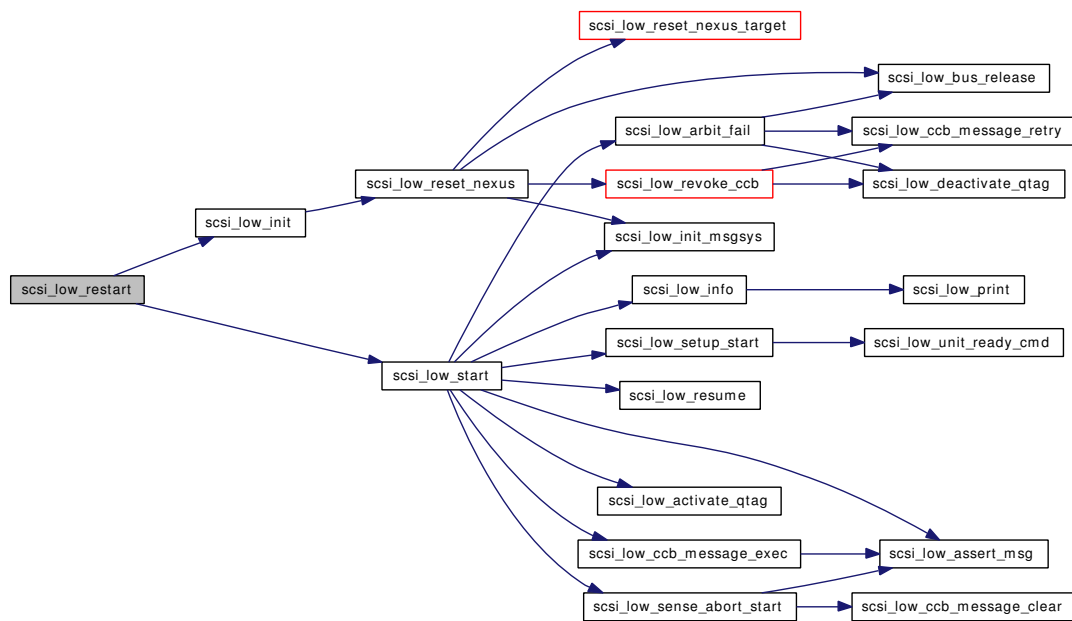
7.26.2.70 int scsi_low_restart (struct [scsi_low_softc](#) * *slp*, int *flags*, u_char * *s*)

Definition at line 2934 of file [scsi_low.c](#).

References [scsi_low_init\(\)](#), and [scsi_low_start\(\)](#).

Referenced by [scsi_low_activate\(\)](#), and [scsi_low_reselected\(\)](#).

Here is the call graph for this function:



7.26.2.71 static int scsi_low_resume (struct scsi_low_softc *) [static]

Definition at line 2316 of file scsi_low.c.

References HW_POWDOWN, HW_RESUME, SCSI_LOW_ENGAGE, SCSI_LOW_TIMEOUT_CHANGE, and SCSI_LOW_TIMEOUT_START.

Referenced by scsi_low_start().

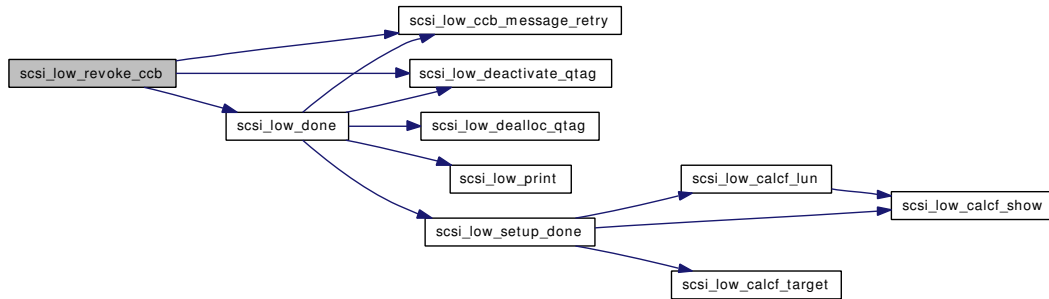
7.26.2.72 static struct slccb * scsi_low_revoke_ccb (struct scsi_low_softc *, struct slccb *, int) [static]

Definition at line 4356 of file scsi_low.c.

References CCB_AUTOSENSE, CCB_CLEARQ, CCB_DISCQ, slccb::ccb_error, slccb::ccb_flags, CCB_INTERNAL, CCB_NORETRY, slccb::ccb_rcnt, CCB_SENSE, CCB_STARTQ, slccb::ccb_tc, slccb::ccb_tcmx, FATALIO, slccb::li, lun_info::li_disc, lun_info::li_discq, PENDINGIO, scsi_low_ccb_message_retry(), scsi_low_deactivate_qtag(), scsi_low_done(), SCSI_LOW_DONE_COMPLETE, SCSI_LOW_MIN_TOUT, slccb::ti, ti, and targ_info::ti_disc.

Referenced by scsi_low_abort_ccb(), scsi_low_establish_ccb(), scsi_low_reset_nexus(), scsi_low_reset_nexus_lun(), and scsi_low_timeout_check().

Here is the call graph for this function:



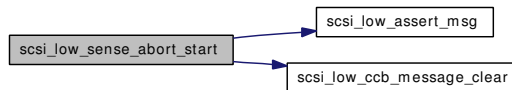
7.26.2.73 `static int scsi_low_sense_abort_start (struct scsi_low_softc *, struct targ_info *, struct lun_info *, struct slccb *)` [static]

Definition at line 2236 of file `scsi_low.c`.

References `CCB_CLEARQ`, `slccb::ccb_flags`, `slccb::ccb_scp`, `slccb::ccb_scsi_cmd`, `slccb::ccb_sense`, `slccb::ccb_tcmx`, `REQUEST_SENSE`, `sc_p::scp_cmd`, `sc_p::scp_cmdlen`, `sc_p::scp_data`, `sc_p::scp_datalen`, `sc_p::scp_direction`, `scsi_low_assert_msg()`, `SCSI_LOW_BZERO`, `scsi_low_ccb_message_clear()`, `SCSI_LOW_MSG_ABORT`, `SCSI_LOW_READ`, `SCSI_LOW_START_NO_QTAG`, `ti`, and `targ_info::ti_setup_msg_done`.

Referenced by `scsi_low_start()`.

Here is the call graph for this function:



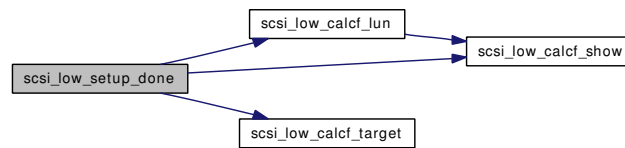
7.26.2.74 `static int scsi_low_setup_done (struct scsi_low_softc *, struct slccb *)` [static]

Definition at line 2539 of file `scsi_low.c`.

References `ABORTIO`, `scsi_sense_data::add_sense_code`, `scsi_sense_data::add_sense_code_qual`, `slccb::ccb_error`, `slccb::ccb_rcnt`, `slccb::ccb_sense`, `lun_info::scsi_low_mode_sense_data::cmp_page`, `lun_info::scsi_low_mode_sense_data::cmp_qc`, `scsi_sense_data::error_code`, `scsi_sense_data::flags`, `slccb::li`, `lun_info::li_diskflags`, `lun_info::li_flags_valid`, `lun_info::li_inq`, `lun_info::li_lun`, `lun_info::li_qflags`, `lun_info::li_sms`, `lun_info::li_state`, `scsi_low_calcf_lun()`, `scsi_low_calcf_show()`, `scsi_low_calcf_target()`, `scsi_low_debug`, `SCSI_LOW_DEBUG_SENSE`, `SCSI_LOW_DISK_LINK`, `SCSI_LOW_DISK_QTAG`, `SCSI_LOW_DISK_SYNC`, `SCSI_LOW_DISK_WIDE`, `SCSI_LOW_DISK_WIDE_16`, `SCSI_LOW_DISK_WIDE_32`, `SCSI_LOW_DONE_COMPLETE`, `SCSI_LOW_DONE_RETRY`, `SCSI_LOW_LUN_FLAGS_ALL_VALID`, `SCSI_LOW_LUN_FLAGS_DISK_VALID`, `SCSI_LOW_LUN_INQ`, `SCSI_LOW_LUN_MODEQ`, `SCSI_LOW_LUN_OK`, `SCSI_LOW_LUN_SLEEP`, `SCSI_LOW_QFLAG_CA_QCLEAR`, `SCSI_LOW_TARG_FLAGS_DISK_VALID`, `lun_info::scsi_low_inq_data::sd_len`, `lun_info::scsi_low_inq_data::sd_support`, `lun_info::scsi_low_inq_data::sd_version`, `scsi_sense_data::segment`, `SELTIMEOUTIO`, `SENSEIO`, `SHOW_CALCF_RES`, `lun_info::scsi_low_mode_sense_data::sms_cmp`, `slccb::ti`, `ti`, `targ_info::ti_diskflags`, and `targ_info::ti_flags_valid`.

Referenced by `scsi_low_done()`.

Here is the call graph for this function:



7.26.2.75 static int scsi_low_setup_start (struct scsi_low_softc *, struct targ_info *, struct lun_info *, struct slccb *) [static]

Definition at line 2269 of file scsi_low.c.

References `slccb::ccb_scp`, `slccb::ccb_tcmx`, `lun_info::li_inq`, `lun_info::li_sms`, `lun_info::li_state`, `sc_p::scp_cmd`, `sc_p::scp_cmdlen`, `sc_p::scp_data`, `sc_p::scp_dataalen`, `sc_p::scp_direction`, `SCSI_LOW_LUN_INQ`, `SCSI_LOW_LUN_MODEQ`, `SCSI_LOW_LUN_SLEEP`, `SCSI_LOW_LUN_START`, `SCSI_LOW_READ`, `SCSI_LOW_START_NO_QTAG`, `SCSI_LOW_START_QTAG`, and `scsi_low_unit_ready_cmd()`.

Referenced by `scsi_low_start()`.

Here is the call graph for this function:



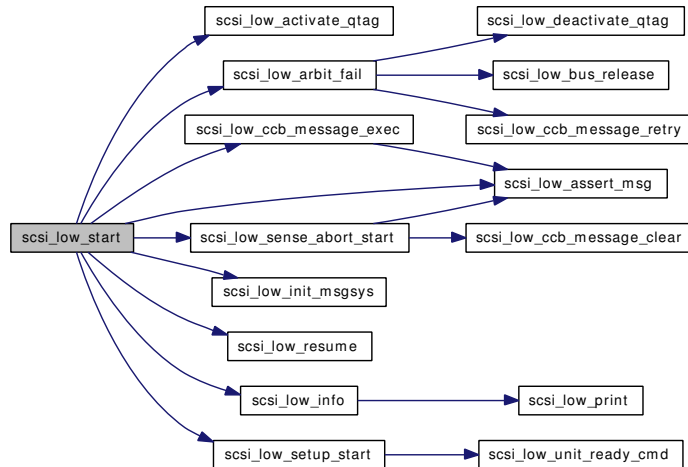
7.26.2.76 static void scsi_low_start (struct scsi_low_softc *) [static]

Definition at line 2337 of file scsi_low.c.

References `CCB_CLEARQ`, `slccb::ccb_dataalen`, `slccb::ccb_error`, `slccb::ccb_flags`, `CCB_INTERNAL`, `slccb::ccb_msgoutflag`, `slccb::ccb_scp`, `CCB_SENSE`, `slccb::ccb_sscp`, `CCB_STARTQ`, `slccb::ccb_tc`, `slccb::ccb_tcmx`, `CCB_URGENT`, `HW_INACTIVE`, `HW_INITIALIZING`, `HW_POWDOWN`, `HW_POWERCTRL`, `HW_RESUME`, `slccb::li`, `lun_info::li_disc`, `lun_info::li_flags`, `lun_info::li_maxnqio`, `lun_info::li_nqio`, `lun_info::li_state`, `sc_p::scp_cmd`, `sc_p::scp_status`, `scsi_low_activate_qtag()`, `scsi_low_arbit_fail()`, `scsi_low_assert_msg()`, `scsi_low_ccb_message_exec()`, `SCSI_LOW_CMD_ORDERED_QTAG`, `scsi_low_info()`, `scsi_low_init_msgsys()`, `SCSI_LOW_LUN_OK`, `SCSI_LOW_MIN_TOUT`, `SCSI_LOW_MSG_HEAD_QTAG`, `SCSI_LOW_MSG_IDENTIFY`, `SCSI_LOW_MSG_ORDERED_QTAG`, `SCSI_LOW_MSG_SIMPLE_QTAG`, `SCSI_LOW_QTAG_OK`, `scsi_low_resume()`, `scsi_low_sense_abort_start()`, `scsi_low_setup_start()`, `SCSI_LOW_START_OK`, `SCSI_LOW_START_QTAG`, `ST_UNKNOWN`, `slccb::ti`, `ti`, `targ_info::ti_log_msgin`, and `targ_info::ti_log_msgout`.

Referenced by `scsi_low_abort_ccb()`, `scsi_low_bus_idle()`, `scsi_low_disconnected()`, `scsi_low_engage()`, `scsi_low_enqueue()`, `scsi_low_restart()`, and `scsi_low_timeout_check()`.

Here is the call graph for this function:



7.26.2.77 `static int scsi_low_start_up (struct scsi_low_softc *) [static]`

Definition at line 4593 of file `scsi_low.c`.

References `SHOW_PROBE_RES`, and `ti`.

Referenced by `scsi_low_attach()`.

7.26.2.78 `static int scsi_low_synch (struct scsi_low_softc *) [static]`

Definition at line 3766 of file `scsi_low.c`.

References `slccb::ccb_flags`, `CCB_SENSE`, `MSGIN_OFFSET`, `MSGIN_PERIOD`, `targ_info::synch::offset`, `targ_info::synch::period`, `period`, `SCSI_LOW_MSG_SYNCH`, `SHOW_SYNCH_NEG`, `ti`, `targ_info::ti_id`, `targ_info::ti_maxsynch`, `targ_info::ti_osynch`, and `targ_info::ti_setup_msg_done`.

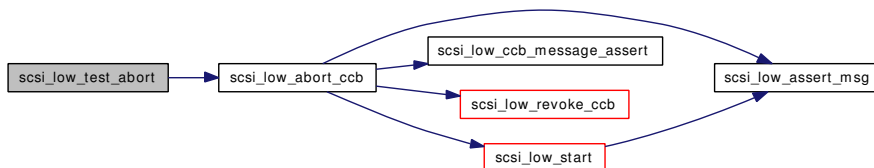
Referenced by `scsi_low_errfunc_synch()`, and `scsi_low_msginfunc_ext()`.

7.26.2.79 `static void scsi_low_test_abort (struct scsi_low_softc *, struct targ_info *, struct lun_info *) [static]`

Definition at line 4683 of file `scsi_low.c`.

References `slccb::li`, `lun_info::li_disc`, `lun_info::li_discq`, and `scsi_low_abort_ccb()`.

Here is the call graph for this function:



7.26.2.80 static void scsi_low_test_atten (struct [scsi_low_softc](#) *, struct [targ_info](#) *, u_int) [static]

Definition at line 4702 of file [scsi_low.c](#).

References [scsi_low_assert_msg\(\)](#), [SCSI_LOW_MAX_ATTEN_CHECK](#), and [ti](#).

Referenced by [scsi_low_msginfunc_ext\(\)](#).

Here is the call graph for this function:



7.26.2.81 static void scsi_low_test_cmdlnk (struct [scsi_low_softc](#) *, struct [slccb](#) *) [static]

Definition at line 4715 of file [scsi_low.c](#).

References [slccb::ccb_flags](#), [slccb::ccb_scsi_cmd](#), and [SCSI_LOW_CMDLNK_NOK](#).

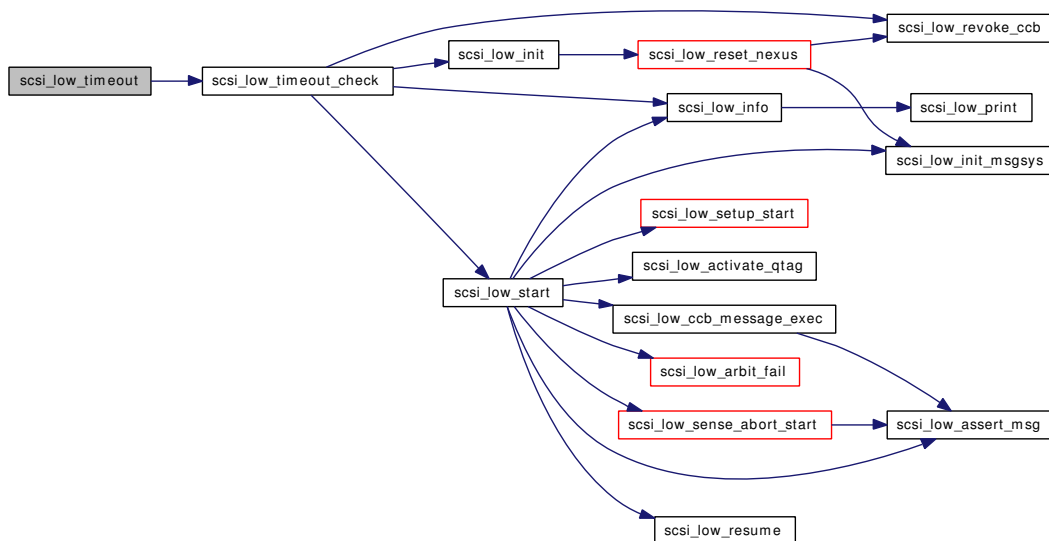
Referenced by [scsi_low_cmd\(\)](#).

7.26.2.82 static void scsi_low_timeout (void *) [static]

Definition at line 1842 of file [scsi_low.c](#).

References [scsi_low_osdep_funcs::scsi_low_osdep_timeout](#), [SCSI_LOW_SPLSCSI](#), [SCSI_LOW_TIMEOUT_CH_IO](#), [scsi_low_timeout_check\(\)](#), [SCSI_LOW_TIMEOUT_START](#), and [scsi_low_softc::sl_osdep_fp](#).

Here is the call graph for this function:



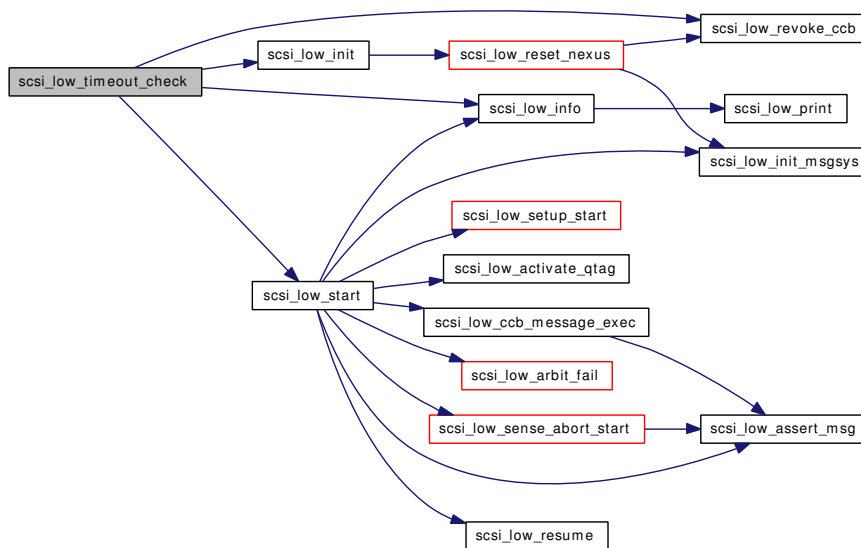
7.26.2.83 `static int scsi_low_timeout_check (struct scsi_low_softc *) [static]`

Definition at line 1856 of file `scsi_low.c`.

References `slccb::ccb_error`, `slccb::ccb_flags`, `CCB_NORETRY`, `slccb::ccb_selrent`, `slccb::ccb_tc`, `HW_POWDOWN`, `HW_POWERCTRL`, `HW_RESUME`, `slccb::li`, `lun_info::li_discq`, `scsi_low_info()`, `scsi_low_init()`, `SCSI_LOW_MAX_SELECTION_RETRY`, `SCSI_LOW_POWDOWN`, `SCSI_LOW_POWDOWN_TC`, `SCSI_LOW_RESTART_HARD`, `scsi_low_revoke_ccb()`, `scsi_low_start()`, `SCSI_LOW_TIMEOUT_CHECK_INTERVAL`, `SCSI_LOW_TIMEOUT_HZ`, `SELTIMEOUTIO`, `ti`, `targ_info::ti_disc`, `targ_info::ti_litab`, and `TIMEOUTIO`.

Referenced by `scsi_low_poll()`, and `scsi_low_timeout()`.

Here is the call graph for this function:



7.26.2.84 `static int scsi_low_translate_error_code (struct slccb *, struct scsi_low_error_code *) [static]`

Definition at line 382 of file `scsi_low.c`.

References `scsi_low_error_code::error_bits`, and `scsi_low_error_code::error_code`.

7.26.2.85 `static void scsi_low_twiddle_wait (void) [static]`

Definition at line 2909 of file `scsi_low.c`.

References `SCSI_LOW_DELAY`, and `TWIDDLEWAIT`.

Referenced by `scsi_low_bus_reset()`.

7.26.2.86 `static void scsi_low_unit_ready_cmd (struct slccb *) [static]`

Definition at line 2224 of file `scsi_low.c`.

References `slccb::ccb_scp`, `sc_p::scp_cmd`, and `SCSI_LOW_READ`.

Referenced by `scsi_low_setup_start()`.

7.26.2.87 `static int scsi_low_wide (struct scsi_low_softc *)` `[static]`

Definition at line 3841 of file `scsi_low.c`.

References `slccb::ccb_flags`, `CCB_SENSE`, `MSGIN_WIDTHHP`, `SCSI_LOW_BUS_WIDTH_8`, `SCSI_LOW_MSG_SYNC`, `SCSI_LOW_MSG_WIDE`, `SHOW_WIDE_NEG`, `ti`, `targ_info::ti_id`, `targ_info::ti_owidth`, `targ_info::ti_setup_msg_done`, and `targ_info::ti_width`.

Referenced by `scsi_low_errfunc_wide()`, and `scsi_low_msginfunc_ext()`.

7.26.3 Variable Documentation

7.26.3.1 `u_int8_t inq_cmd[6]` `[static]`

Initial value:

```
{INQUIRY, 0, 0, 0,
                                sizeof(struct scsi_low_inq_data), 0}
```

Definition at line 2216 of file `scsi_low.c`.

7.26.3.2 `u_char* phase[]` `[static]`

Initial value:

```
{
    "FREE", "ARBSTART", "SELSTART", "SELECTED",
    "CMDOUT", "DATA", "MSGIN", "MSGOUT", "STATIN", "DISC", "RESEL"
}
```

Definition at line 4758 of file `scsi_low.c`.

7.26.3.3 `u_int8_t scsi_low_cmd_flags[256]` `[static]`

Initial value:

```
{
    0, 0, 0, 0, 0, 0, 0, 0, 0, 5, 0, 5, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 2, 2, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 5, 0, 5, 0, 0, 0, 5, 5,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 5, 5,
}
```

Definition at line 331 of file `scsi_low.c`.

7.26.3.4 `int scsi_low_debug = 0`

Definition at line 215 of file `scsi_low.c`.

Referenced by `scsi_low_setup_done()`.

7.26.3.5 struct `scsi_low_msgin_data` `scsi_low_msgin_data` []

Definition at line 3234 of file `scsi_low.c`.

7.26.3.6 struct `scsi_low_msgout_data` `scsi_low_msgout_data` []

Initial value:

```
{
    {SCSI_LOW_MSG_RESET, MSG_RESET, scsi_low_msgfunc_reset, NULL, MSG_RELEASE_ATN},
    {SCSI_LOW_MSG_REJECT, MSG_REJECT, NULL, NULL, MSG_RELEASE_ATN},
    {SCSI_LOW_MSG_PARITY, MSG_PARITY, NULL, NULL, MSG_RELEASE_ATN},
    {SCSI_LOW_MSG_ERROR, MSG_I_ERROR, NULL, NULL, MSG_RELEASE_ATN},
    {SCSI_LOW_MSG_IDENTIFY, MSG_IDENTIFY, scsi_low_msgfunc_identify, scsi_low_errfunc_identify, 0},
    {SCSI_LOW_MSG_ABORT, MSG_ABORT, scsi_low_msgfunc_abort, NULL, MSG_RELEASE_ATN},
    {SCSI_LOW_MSG_TERMIO, MSG_TERM_IO, NULL, NULL, MSG_RELEASE_ATN},
    {SCSI_LOW_MSG_SIMPLE_QTAG, MSG_SIMPLE_QTAG, scsi_low_msgfunc_qtag, scsi_low_errfunc_qtag, 0},
    {SCSI_LOW_MSG_ORDERED_QTAG, MSG_ORDERED_QTAG, scsi_low_msgfunc_qtag, scsi_low_errfunc_qtag, 0},
    {SCSI_LOW_MSG_HEAD_QTAG, MSG_HEAD_QTAG, scsi_low_msgfunc_qtag, scsi_low_errfunc_qtag, 0},
    {SCSI_LOW_MSG_ABORT_QTAG, MSG_ABORT_QTAG, scsi_low_msgfunc_qabort, NULL, MSG_RELEASE_ATN},
    {SCSI_LOW_MSG_CLEAR_QTAG, MSG_CLEAR_QTAG, scsi_low_msgfunc_abort, NULL, MSG_RELEASE_ATN},
    {SCSI_LOW_MSG_WIDE, MSG_EXTEND, scsi_low_msgfunc_wide, scsi_low_errfunc_wide, MSG_RELEASE_ATN},
    {SCSI_LOW_MSG_SYNCH, MSG_EXTEND, scsi_low_msgfunc_synch, scsi_low_errfunc_synch, MSG_RELEASE_ATN},
    {SCSI_LOW_MSG_NOOP, MSG_NOOP, NULL, NULL, MSG_RELEASE_ATN},
    {SCSI_LOW_MSG_ALL, 0},
}
```

Definition at line 3195 of file `scsi_low.c`.

Referenced by `scsi_low_msgout()`.

7.26.3.7 int `scsi_low_test` = 0

Definition at line 222 of file `scsi_low.c`.

7.26.3.8 int `scsi_low_test_id` = 0

Definition at line 223 of file `scsi_low.c`.

7.26.3.9 int `scsi_low_version_major` = 2

Definition at line 185 of file `scsi_low.c`.

7.26.3.10 int `scsi_low_version_minor` = 17

Definition at line 186 of file `scsi_low.c`.

7.26.3.11 struct `scsi_low_softc_tab` `sl_tab` = LIST_HEAD_INITIALIZER(`sl_tab`) [static]

Definition at line 188 of file `scsi_low.c`.

Referenced by `scsi_low_attach()`, and `scsi_low_info()`.

7.26.3.12 `u_int8_t sms_cmd[6]` [static]**Initial value:**

```
{SLSC_MODE_SENSE_SHORT, 0x08, 0x0a, 0,  
    sizeof(struct scsi_low_mode_sense_data), 0}
```

Definition at line 2214 of file scsi_low.c.

7.26.3.13 `u_int8_t ss_cmd[6] = {START_STOP, 0, 0, 0, SSS_START, 0}` [static]

Definition at line 2213 of file scsi_low.c.

7.26.3.14 `char tw_chars[] = "|/-\\" data-bbox="144 357 424 373" data-label="Text">

Definition at line 2905 of file scsi_low.c.`**7.26.3.15** `int tw_pos` [static]

Definition at line 2904 of file scsi_low.c.

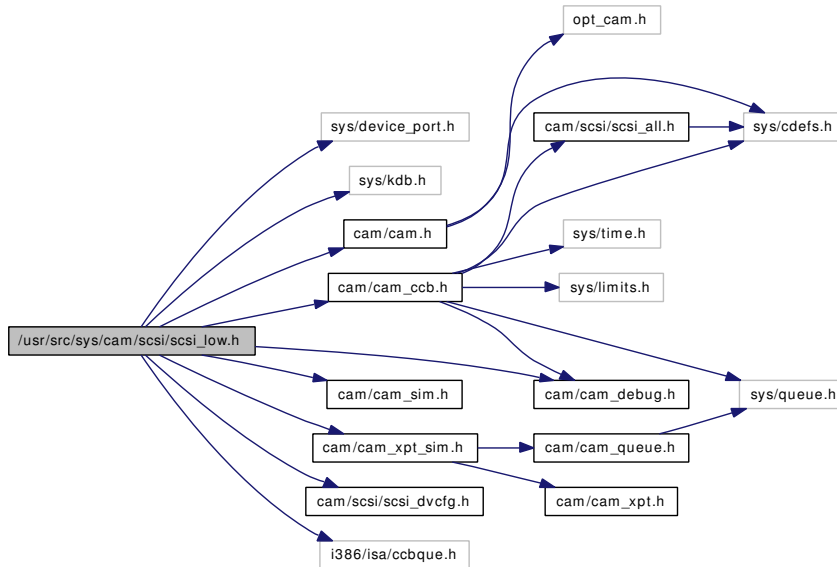
7.26.3.16 `u_int8_t unit_ready_cmd[6]` [static]

Definition at line 2218 of file scsi_low.c.

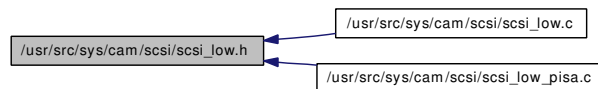
7.27 /usr/src/sys/cam/scsi/scsi_low.h File Reference

```
#include <sys/device_port.h>
#include <sys/kdb.h>
#include <cam/cam.h>
#include <cam/cam_ccb.h>
#include <cam/cam_sim.h>
#include <cam/cam_xpt_sim.h>
#include <cam/cam_debug.h>
#include <cam/scsi/scsi_dvcfg.h>
#include <i386/isa/ccbqueue.h>
```

Include dependency graph for scsi_low.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- struct [scsi_low_osdep_interface](#)
- struct [scsi_low_osdep_targ_interface](#)
- struct [scsi_low_osdep_lun_interface](#)
- struct [scsi_low_osdep_funcs](#)
- struct [sc_p](#)

- struct [slccb](#)
- struct [lun_info](#)
- struct [lun_info::scsi_low_inq_data](#)
- struct [lun_info::scsi_low_mode_sense_data](#)
- struct [scsi_low_msg_log](#)
- struct [targ_info](#)
- struct [targ_info::synch](#)
- struct [scsi_low_funcs](#)
- struct [scsi_low_softc](#)

Defines

- #define [SCSI_LOW_DIAGNOSTIC](#)
- #define [SCSI_LOW_ALT_QTAG_ALLOCATE](#)
- #define [SCSI_LOW_INTERFACE_CAM](#)
- #define [CAM](#)
- #define [SCSI_LOW_DEBUGGER](#)(dev) [kdb_enter](#)(dev)
- #define [SCSI_LOW_DELAY](#)(mu) [DELAY](#)((mu))
- #define [SCSI_LOW_SPLSCSI](#) [splcam](#)
- #define [SCSI_LOW_BZERO](#)(pt, size) [bzero](#)((pt), (size))
- #define [SCSI_LOW_TIMEOUT_STOP](#) 0
- #define [SCSI_LOW_TIMEOUT_START](#) 1
- #define [SCSI_LOW_TIMEOUT_CH_IO](#) 0
- #define [SCSI_LOW_TIMEOUT_CH_ENGAGE](#) 1
- #define [SCSI_LOW_TIMEOUT_CH_RECOVER](#) 2
- #define [SCSI_LOW_SYNC](#) [DVF_SCSI_SYNC](#)
- #define [SCSI_LOW_DISC](#) [DVF_SCSI_DISC](#)
- #define [SCSI_LOW_WAIT](#) [DVF_SCSI_WAIT](#)
- #define [SCSI_LOW_LINK](#) [DVF_SCSI_LINK](#)
- #define [SCSI_LOW_QTAG](#) [DVF_SCSI_QTAG](#)
- #define [SCSI_LOW_NOPARITY](#) [DVF_SCSI_NOPARITY](#)
- #define [SCSI_LOW_SAVESP](#) [DVF_SCSI_SAVESP](#)
- #define [SCSI_LOW_DEFCFG](#) [DVF_SCSI_DEFCFG](#)
- #define [SCSI_LOW_BITS](#) [DVF_SCSI_BITS](#)
- #define [SCSI_LOW_PERIOD](#)(n) [DVF_SCSI_PERIOD](#)(n)
- #define [SCSI_LOW_OFFSET](#)(n) [DVF_SCSI_OFFSET](#)(n)
- #define [SCSI_LOW_NTARGETS](#) 8
- #define [SCSI_LOW_NCCB](#) 128
- #define [SCSI_LOW_MAX_RETRY](#) 3
- #define [SCSI_LOW_MAX_SELECTION_RETRY](#) 10
- #define [SCSI_LOW_TIMEOUT_HZ](#) 10
- #define [SCSI_LOW_MIN_TOUT](#) 12
- #define [SCSI_LOW_TIMEOUT_CHECK_INTERVAL](#) 1
- #define [SCSI_LOW_POWDOWN_TC](#) 15
- #define [SCSI_LOW_MAX_PHCHANGES](#) 256
- #define [SCSI2_RESET_DELAY](#) 5000000
- #define [SCSI_LOW_MAX_MSGLEN](#) 32
- #define [SCSI_LOW_MSG_LOG_DATALEN](#) 8
- #define [SCSI_LOW_RWUNK](#) (-1)
- #define [SCSI_LOW_WRITE](#) 0

- #define `SCSI_LOW_READ` 1
- #define `SCSI_LOW_UNKLUN` ((u_int) -1)
- #define `SCSI_LOW_UNKTAG` ((scsi_low_tag_t) -1)
- #define `CCB_INTERNAL` 0x0001
- #define `CCB_SENSE` 0x0002
- #define `CCB_CLEARQ` 0x0004
- #define `CCB_DISCQ` 0x0008
- #define `CCB_STARTQ` 0x0010
- #define `CCB_POLLED` 0x0100
- #define `CCB_NORETRY` 0x0200
- #define `CCB_AUTOSENSE` 0x0400
- #define `CCB_URGENT` 0x0800
- #define `CCB_NOSDONE` 0x1000
- #define `CCB_SCSIIO` 0x2000
- #define `CCB_SILENT` 0x4000
- #define `SCSI_LOW_MAXNEXUS` (sizeof(u_int) * NBBY)
- #define `SCSI_LOW_QFLAG_CA_QCLEAR` 0x01
- #define `SCSI_LOW_LUN_SLEEP` 0x00
- #define `SCSI_LOW_LUN_START` 0x01
- #define `SCSI_LOW_LUN_INQ` 0x02
- #define `SCSI_LOW_LUN_MODEQ` 0x03
- #define `SCSI_LOW_LUN_OK` 0x04
- #define `SCSI_LOW_LUN_FLAGS_USER_VALID` 0x0001
- #define `SCSI_LOW_LUN_FLAGS_DISK_VALID` 0x0002
- #define `SCSI_LOW_LUN_FLAGS_QUIRKS_VALID` 0x0004
- #define `SCSI_LOW_LUN_FLAGS_ALL_VALID`
- #define `PH_NULL` 0x00
- #define `PH_ARBSTART` 0x01
- #define `PH_SELSTART` 0x02
- #define `PH_SELECTED` 0x03
- #define `PH_CMD` 0x04
- #define `PH_DATA` 0x05
- #define `PH_MSGIN` 0x06
- #define `PH_MSGOUT` 0x07
- #define `PH_STAT` 0x08
- #define `PH_DISC` 0x09
- #define `PH_RESEL` 0x0a
- #define `SCSI_LOW_MSG_RESET` 0x00000001
- #define `SCSI_LOW_MSG_REJECT` 0x00000002
- #define `SCSI_LOW_MSG_PARITY` 0x00000004
- #define `SCSI_LOW_MSG_ERROR` 0x00000008
- #define `SCSI_LOW_MSG_IDENTIFY` 0x00000010
- #define `SCSI_LOW_MSG_ABORT` 0x00000020
- #define `SCSI_LOW_MSG_TERMIO` 0x00000040
- #define `SCSI_LOW_MSG_SIMPLE_QTAG` 0x00000080
- #define `SCSI_LOW_MSG_ORDERED_QTAG` 0x00000100
- #define `SCSI_LOW_MSG_HEAD_QTAG` 0x00000200
- #define `SCSI_LOW_MSG_ABORT_QTAG` 0x00000400
- #define `SCSI_LOW_MSG_CLEAR_QTAG` 0x00000800
- #define `SCSI_LOW_MSG_WIDE` 0x00001000

- #define `SCSI_LOW_MSG_SYNCH` 0x00002000
- #define `SCSI_LOW_MSG_NOOP` 0x00004000
- #define `SCSI_LOW_MSG_LAST` 0x00008000
- #define `SCSI_LOW_MSG_ALL` 0xffffffff
- #define `SCSI_LOW_TARG_FLAGS_USER_VALID` 0x0001
- #define `SCSI_LOW_TARG_FLAGS_DISK_VALID` 0x0002
- #define `SCSI_LOW_TARG_FLAGS_QUIRKS_VALID` 0x0004
- #define `SCSI_LOW_TARG_FLAGS_ALL_VALID`
- #define `SCSI_LOW_BUS_WIDTH_8` 0
- #define `SCSI_LOW_BUS_WIDTH_16` 1
- #define `SCSI_LOW_BUS_WIDTH_32` 2
- #define `SCSI_LOW_START_OK` 0
- #define `SCSI_LOW_START_FAIL` 1
- #define `SCSI_LOW_INFO_ALLOC` 0
- #define `SCSI_LOW_INFO_REVOKE` 1
- #define `SCSI_LOW_INFO_DEALLOC` 2
- #define `SCSI_LOW_POWDOWN` 1
- #define `SCSI_LOW_ENGAGE` 2
- #define `SC_LOW_INIT_T` (int (*)(sc_low_t, int))
- #define `SC_LOW_BUSRST_T` (void (*)(sc_low_t))
- #define `SC_LOW_TARG_INIT_T` (int (*)(sc_low_t, struct targ_info *, int))
- #define `SC_LOW_LUN_INIT_T` (int (*)(sc_low_t, struct targ_info *, struct lun_info *, int))
- #define `SC_LOW_SELECT_T` (int (*)(sc_low_t, struct slccb *))
- #define `SC_LOW_ATTEN_T` (void (*)(sc_low_t))
- #define `SC_LOW_NEXUS_T` (int (*)(sc_low_t))
- #define `SC_LOW_MSG_T` (int (*)(sc_low_t, struct targ_info *, u_int))
- #define `SC_LOW_POLL_T` (int (*)(void *))
- #define `SC_LOW_POWER_T` (int (*)(sc_low_t, u_int))
- #define `SC_LOW_TIMEOUT_T` (int (*)(sc_low_t))
- #define `sl_dev` sl_si.si_dev
- #define `MSGPH_NULL` 0x00
- #define `MSGPH_DISC` 0x01
- #define `MSGPH_CMDC` 0x02
- #define `MSGPH_ABORT` 0x03
- #define `MSGPH_TERM` 0x04
- #define `MSGPH_LCTERM` 0x05
- #define `MSGPH_RESET` 0x06
- #define `FATALIO` 0x0001
- #define `ABORTIO` 0x0002
- #define `TIMEOUTIO` 0x0004
- #define `SELTIMEOUTIO` 0x0008
- #define `PDMAERR` 0x0010
- #define `MSGERR` 0x0020
- #define `PARITYERR` 0x0040
- #define `BUSYERR` 0x0080
- #define `STATERR` 0x0100
- #define `UACAERR` 0x0200
- #define `SENSEIO` 0x1000
- #define `SENSEERR` 0x2000
- #define `UBFERR` 0x4000

- #define PENDINGIO 0x8000
- #define SCSI_LOW_ERRORBITS "\020\017ubferr\016senseerr\015senseio\012uacaerr\011staterr\010busy\007pari
- #define HW_POWDOWN 0x0001
- #define HW_RESUME 0x0002
- #define HW_PDMASTART 0x0004
- #define HW_INACTIVE 0x0008
- #define HW_POWERCTRL 0x0010
- #define HW_INITIALIZING 0x0020
- #define HW_READ_PADDING 0x1000
- #define HW_WRITE_PADDING 0x2000
- #define CFG_NODISC 0x0001
- #define CFG_NOPARITY 0x0002
- #define CFG_NOATTEN 0x0004
- #define CFG_ASYNC 0x0008
- #define CFG_NOQTAG 0x0010
- #define SHOW_SYNCH_NEG 0x0001
- #define SHOW_WIDE_NEG 0x0002
- #define SHOW_CALCF_RES 0x0010
- #define SHOW_PROBE_RES 0x0020
- #define SHOW_ALL_NEG -1
- #define SCSI_LOW_MSGOUT_INIT 0x00000001
- #define SCSI_LOW_MSGOUT_UNIFY 0x00000002
- #define SCSI_LOW_DATA_PE 0x80000000
- #define SCSI_LOW_RESTART_HARD 1
- #define SCSI_LOW_RESTART_SOFT 0
- #define SCSI_LOW_SETUP_PHASE(ti, phase)
- #define SCSI_LOW_SETUP_MSGPHASE(slp, PHASE)
- #define SCSI_LOW_ASSERT_ATN(slp)
- #define SCSI_LOW_DEASSERT_ATN(slp)
- #define ST_GOOD 0x00
- #define ST_CHKCOND 0x02
- #define ST_MET 0x04
- #define ST_BUSY 0x08
- #define ST_INTERGOOD 0x10
- #define ST_INTERMET 0x14
- #define ST_CONFLICT 0x18
- #define ST_CMDTERM 0x22
- #define ST_QUEFULL 0x28
- #define ST_UNKNOWN 0xff
- #define MSG_COMP 0x00
- #define MSG_EXTEND 0x01
- #define MKMSG_EXTEND(XLEN, XCODE) (((u_int)(XLEN)) << NBBY) | ((u_int)(XCODE))
- #define MSG_EXTEND_MDPCODE 0x00
- #define MSG_EXTEND_MDPLEN 0x05
- #define MSG_EXTEND_SYNCHCODE 0x01
- #define MSG_EXTEND_SYNCHLEN 0x03
- #define MSG_EXTEND_WIDECODE 0x03
- #define MSG_EXTEND_WIDELLEN 0x02
- #define MSG_SAVESP 0x02
- #define MSG_RESTORESP 0x03

- #define `MSG_DISCON` 0x04
- #define `MSG_I_ERROR` 0x05
- #define `MSG_ABORT` 0x06
- #define `MSG_REJECT` 0x07
- #define `MSG_NOOP` 0x08
- #define `MSG_PARITY` 0x09
- #define `MSG_LCOMP` 0x0a
- #define `MSG_LCOMP_F` 0x0b
- #define `MSG_RESET` 0x0c
- #define `MSG_ABORT_QTAG` 0x0d
- #define `MSG_CLEAR_QTAG` 0x0e
- #define `MSG_TERM_IO` 0x11
- #define `MSG_SIMPLE_QTAG` 0x20
- #define `MSG_HEAD_QTAG` 0x21
- #define `MSG_ORDERED_QTAG` 0x22
- #define `MSG_IDENTIFY` 0x80
- #define `MSG_IDENTIFY_DISCPRIV` 0x40

Typedefs

- typedef `scsi_sense_data` `scsi_low_osdep_sense_data_t`
- typedef int `scsi_low_tag_t`
- typedef `scsi_low_softc *` `sc_low_t`

Functions

- `LIST_HEAD` (`scsi_low_softc_tab`, `scsi_low_softc`)
- `TAILQ_HEAD` (`targ_info_tab`, `targ_info`)
- `LIST_HEAD` (`lun_info_tab`, `lun_info`)
- int `scsi_low_attach` (`struct scsi_low_softc *`, int, int, int, int, int)
- int `scsi_low_dettach` (`struct scsi_low_softc *`)
- int `scsi_low_is_busy` (`struct scsi_low_softc *`)
- int `scsi_low_activate` (`struct scsi_low_softc *`)
- int `scsi_low_deactivate` (`struct scsi_low_softc *`)
- void `scsi_low_bus_idle` (`struct scsi_low_softc *`)
- void `scsi_low_arbit_fail` (`struct scsi_low_softc *`, `struct slccb *`)
- static __inline void `scsi_low_arbit_win` (`struct scsi_low_softc *`)
- int `scsi_low_msgout` (`struct scsi_low_softc *`, `struct targ_info *`, u_int)
- int `scsi_low_msgin` (`struct scsi_low_softc *`, `struct targ_info *`, u_int)
- static __inline int `scsi_low_statusin` (`struct scsi_low_softc *`, `struct targ_info *`, u_int)
- int `scsi_low_data` (`struct scsi_low_softc *`, `struct targ_info *`, `struct buf **`, int)
- static __inline void `scsi_low_data_finish` (`struct scsi_low_softc *`)
- int `scsi_low_cmd` (`struct scsi_low_softc *`, `struct targ_info *`)
- `targ_info *` `scsi_low_reselected` (`struct scsi_low_softc *`, u_int)
- int `scsi_low_disconnected` (`struct scsi_low_softc *`, `struct targ_info *`)
- int `scsi_low_restart` (`struct scsi_low_softc *`, int, u_char *)
- void `scsi_low_print` (`struct scsi_low_softc *`, `struct targ_info *`)
- void `scsi_low_bus_reset` (`struct scsi_low_softc *`)
- static __inline void `scsi_low_attention` (`struct scsi_low_softc *`)

- static `__inline int scsi_low_is_msgout_continue` (struct `targ_info *`, `u_int`)
- static `__inline int scsi_low_assert_msg` (struct `scsi_low_softc *`, struct `targ_info *`, `u_int`, `int`)
- static `__inline int scsi_low_is_disconnect_ok` (struct `slccb *`)
- static `__inline int scsi_low_is_msgout_continue` (`ti`, `mask`) struct `targ_info *ti`

Variables

- `u_int mask`
- `targ_info * ti`
- `u_int msg`
- `int now`
- `targ_info * ti`
- `u_int c`

7.27.1 Define Documentation

7.27.1.1 `#define ABORTIO 0x0002`

Definition at line 596 of file `scsi_low.h`.

Referenced by `scsi_low_abort_ccb()`, `scsi_low_done()`, and `scsi_low_setup_done()`.

7.27.1.2 `#define BUSYERR 0x0080`

Definition at line 602 of file `scsi_low.h`.

Referenced by `scsi_low_done()`.

7.27.1.3 `#define CAM`

Definition at line 58 of file `scsi_low.h`.

7.27.1.4 `#define CCB_AUTOSENSE 0x0400`

Definition at line 268 of file `scsi_low.h`.

Referenced by `scsi_low_done()`, and `scsi_low_revoke_ccb()`.

7.27.1.5 `#define CCB_CLEARQ 0x0004`

Definition at line 263 of file `scsi_low.h`.

Referenced by `scsi_low_done()`, `scsi_low_revoke_ccb()`, `scsi_low_sense_abort_start()`, and `scsi_low_start()`.

7.27.1.6 `#define CCB_DISCQ 0x0008`

Definition at line 264 of file `scsi_low.h`.

Referenced by `scsi_low_abort_ccb()`, `scsi_low_disconnected()`, `scsi_low_establish_ccb()`, and `scsi_low_revoke_ccb()`.

7.27.1.7 #define CCB_INTERNAL 0x0001

Definition at line 261 of file scsi_low.h.

Referenced by scsi_low_done(), scsi_low_msginfunc_lcc(), scsi_low_revoke_ccb(), and scsi_low_start().

7.27.1.8 #define CCB_NORETRY 0x0200

Definition at line 267 of file scsi_low.h.

Referenced by scsi_low_abort_ccb(), scsi_low_done(), scsi_low_establish_ccb(), scsi_low_message_enqueue(), scsi_low_revoke_ccb(), and scsi_low_timeout_check().

7.27.1.9 #define CCB_NOSDONE 0x1000

Definition at line 270 of file scsi_low.h.

7.27.1.10 #define CCB_POLLED 0x0100

Definition at line 266 of file scsi_low.h.

7.27.1.11 #define CCB_SCSIIO 0x2000

Definition at line 271 of file scsi_low.h.

7.27.1.12 #define CCB_SENSE 0x0002

Definition at line 262 of file scsi_low.h.

Referenced by scsi_low_done(), scsi_low_revoke_ccb(), scsi_low_start(), scsi_low_synch(), and scsi_low_wide().

7.27.1.13 #define CCB_SILENT 0x4000

Definition at line 272 of file scsi_low.h.

Referenced by scsi_low_establish_ccb().

7.27.1.14 #define CCB_STARTQ 0x0010

Definition at line 265 of file scsi_low.h.

Referenced by scsi_low_abort_ccb(), scsi_low_arbit_fail(), scsi_low_disconnected(), scsi_low_enqueue(), scsi_low_msginfunc_lcc(), scsi_low_reset_nexus(), scsi_low_reset_nexus_lun(), scsi_low_revoke_ccb(), and scsi_low_start().

7.27.1.15 #define CCB_URGENT 0x0800

Definition at line 269 of file scsi_low.h.

Referenced by scsi_low_enqueue(), and scsi_low_start().

7.27.1.16 #define CFG_ASYNC 0x0008

Definition at line 634 of file scsi_low.h.

Referenced by scsi_low_calcf_target().

7.27.1.17 #define CFG_NOATTEN 0x0004

Definition at line 633 of file scsi_low.h.

7.27.1.18 #define CFG_NODISC 0x0001

Definition at line 631 of file scsi_low.h.

Referenced by scsi_low_calcf_lun().

7.27.1.19 #define CFG_NOPARITY 0x0002

Definition at line 632 of file scsi_low.h.

Referenced by scsi_low_calcf_lun().

7.27.1.20 #define CFG_NOQTAG 0x0010

Definition at line 635 of file scsi_low.h.

Referenced by scsi_low_calcf_lun().

7.27.1.21 #define FATALIO 0x0001

Definition at line 595 of file scsi_low.h.

Referenced by scsi_low_cmd(), scsi_low_data(), scsi_low_disconnected(), scsi_low_done(), scsi_low_msgfunc_identify(), scsi_low_msgin(), scsi_low_msginfunc_simple_qtag(), scsi_low_msgout(), and scsi_low_revoke_ccb().

7.27.1.22 #define HW_INACTIVE 0x0008

Definition at line 624 of file scsi_low.h.

Referenced by scsi_low_activate(), scsi_low_deactivate(), scsi_low_init(), and scsi_low_start().

7.27.1.23 #define HW_INITIALIZING 0x0020

Definition at line 626 of file scsi_low.h.

Referenced by scsi_low_init(), and scsi_low_start().

7.27.1.24 #define HW_PDMASSTART 0x0004

Definition at line 623 of file scsi_low.h.

Referenced by scsi_low_reset_nexus().

7.27.1.25 **#define HW_POWDOWN 0x0001**

Definition at line 621 of file scsi_low.h.

Referenced by scsi_low_init(), scsi_low_resume(), scsi_low_start(), and scsi_low_timeout_check().

7.27.1.26 **#define HW_POWERCTRL 0x0010**

Definition at line 625 of file scsi_low.h.

Referenced by scsi_low_init(), scsi_low_start(), and scsi_low_timeout_check().

7.27.1.27 **#define HW_READ_PADDING 0x1000**

Definition at line 627 of file scsi_low.h.

7.27.1.28 **#define HW_RESUME 0x0002**

Definition at line 622 of file scsi_low.h.

Referenced by scsi_low_engage(), scsi_low_init(), scsi_low_resume(), scsi_low_start(), and scsi_low_timeout_check().

7.27.1.29 **#define HW_WRITE_PADDING 0x2000**

Definition at line 628 of file scsi_low.h.

7.27.1.30 **#define MKMSG_EXTEND(XLEN, XCODE) (((u_int)(XLEN)) << NBBY) | ((u_int)(XCODE))**

Referenced by scsi_low_msginfunc_ext().

7.27.1.31 **#define MSG_ABORT 0x06**

7.27.1.32 **#define MSG_ABORT_QTAG 0x0d**

7.27.1.33 **#define MSG_CLEAR_QTAG 0x0e**

7.27.1.34 **#define MSG_COMP 0x00**

7.27.1.35 **#define MSG_DISCON 0x04**

7.27.1.36 **#define MSG_EXTEND 0x01**

7.27.1.37 **#define MSG_EXTEND_MDPCODE 0x00**

Referenced by scsi_low_msginfunc_ext().

7.27.1.38 #define MSG_EXTEND_MDPLEN 0x05

Referenced by `scsi_low_msgfunc_ext()`.

7.27.1.39 #define MSG_EXTEND_SYNCHCODE 0x01

Referenced by `scsi_low_msgfunc_synch()`, and `scsi_low_msgfunc_ext()`.

7.27.1.40 #define MSG_EXTEND_SYNCHLEN 0x03

Referenced by `scsi_low_msgfunc_synch()`, and `scsi_low_msgfunc_ext()`.

7.27.1.41 #define MSG_EXTEND_WIDECODE 0x03

Referenced by `scsi_low_msgfunc_wide()`, and `scsi_low_msgfunc_ext()`.

7.27.1.42 #define MSG_EXTEND_WIDELEN 0x02

Referenced by `scsi_low_msgfunc_wide()`, and `scsi_low_msgfunc_ext()`.

7.27.1.43 #define MSG_HEAD_QTAG 0x21**7.27.1.44 #define MSG_I_ERROR 0x05****7.27.1.45 #define MSG_IDENTIFY 0x80**

Referenced by `scsi_low_msgfunc_identify()`, and `scsi_low_msgin()`.

7.27.1.46 #define MSG_IDENTIFY_DISCPRIV 0x40

Referenced by `scsi_low_msgfunc_identify()`.

7.27.1.47 #define MSG_LCOMP 0x0a**7.27.1.48 #define MSG_LCOMP_F 0x0b****7.27.1.49 #define MSG_NOOP 0x08**

Referenced by `scsi_low_msgfunc_qtag()`.

7.27.1.50 #define MSG_ORDERED_QTAG 0x22

7.27.1.51 #define MSG_PARITY 0x09

7.27.1.52 #define MSG_REJECT 0x07

7.27.1.53 #define MSG_RESET 0x0c

7.27.1.54 #define MSG_RESTORESP 0x03

7.27.1.55 #define MSG_SAVESP 0x02

7.27.1.56 #define MSG_SIMPLE_QTAG 0x20

7.27.1.57 #define MSG_TERM_IO 0x11

7.27.1.58 #define MSGERR 0x0020

Definition at line 600 of file scsi_low.h.

Referenced by scsi_low_msginfunc_cc(), and scsi_low_msginfunc_lcc().

7.27.1.59 #define MSGPH_ABORT 0x03

Definition at line 588 of file scsi_low.h.

Referenced by scsi_low_disconnected(), and scsi_low_msgfunc_abort().

7.27.1.60 #define MSGPH_CMDDC 0x02

Definition at line 587 of file scsi_low.h.

Referenced by scsi_low_disconnected(), and scsi_low_msginfunc_cc().

7.27.1.61 #define MSGPH_DISC 0x01

Definition at line 586 of file scsi_low.h.

Referenced by scsi_low_disconnected(), and scsi_low_msginfunc_disc().

7.27.1.62 #define MSGPH_LCTERM 0x05

Definition at line 590 of file scsi_low.h.

Referenced by scsi_low_disconnected(), and scsi_low_msginfunc_lcc().

7.27.1.63 #define MSGPH_NULL 0x00

Definition at line 585 of file scsi_low.h.

Referenced by scsi_low_disconnected(), and scsi_low_init_msgsys().

7.27.1.64 #define MSGPH_RESET 0x06

Definition at line 591 of file scsi_low.h.

Referenced by scsi_low_disconnected(), and scsi_low_msgfunc_reset().

7.27.1.65 #define MSGPH_TERM 0x04

Definition at line 589 of file scsi_low.h.

Referenced by scsi_low_disconnected(), and scsi_low_msgfunc_qabort().

7.27.1.66 #define PARITYERR 0x0040

Definition at line 601 of file scsi_low.h.

7.27.1.67 #define PDMAERR 0x0010

Definition at line 599 of file scsi_low.h.

Referenced by scsi_low_data(), and scsi_low_done().

7.27.1.68 #define PENDINGIO 0x8000

Definition at line 608 of file scsi_low.h.

Referenced by scsi_low_enqueue(), and scsi_low_revoke_ccb().

7.27.1.69 #define PH_ARBSTART 0x01

Definition at line 412 of file scsi_low.h.

7.27.1.70 #define PH_CMD 0x04

Definition at line 415 of file scsi_low.h.

7.27.1.71 #define PH_DATA 0x05

Definition at line 416 of file scsi_low.h.

7.27.1.72 #define PH_DISC 0x09

Definition at line 420 of file scsi_low.h.

Referenced by scsi_low_bus_release(), and scsi_low_reselected().

7.27.1.73 #define PH_MSGIN 0x06

Definition at line 417 of file scsi_low.h.

7.27.1.74 #define PH_MSGOUT 0x07

Definition at line 418 of file scsi_low.h.

Referenced by scsi_low_msgfunc_identify(), and scsi_low_msgfunc_qtag().

7.27.1.75 #define PH_NULL 0x00

Definition at line 411 of file scsi_low.h.

Referenced by scsi_low_bus_release(), and scsi_low_reselected().

7.27.1.76 #define PH_RESEL 0x0a

Definition at line 421 of file scsi_low.h.

Referenced by scsi_low_reselected().

7.27.1.77 #define PH_SELECTED 0x03

Definition at line 414 of file scsi_low.h.

7.27.1.78 #define PH_SELSTART 0x02

Definition at line 413 of file scsi_low.h.

Referenced by scsi_low_disconnected().

7.27.1.79 #define PH_STAT 0x08

Definition at line 419 of file scsi_low.h.

7.27.1.80 #define SC_LOW_ATTEN_T (void (*)(sc_low_t))

Definition at line 522 of file scsi_low.h.

7.27.1.81 #define SC_LOW_BUSRST_T (void (*)(sc_low_t))

Definition at line 518 of file scsi_low.h.

7.27.1.82 #define SC_LOW_INIT_T (int (*)(sc_low_t, int))

Definition at line 517 of file scsi_low.h.

7.27.1.83 #define SC_LOW_LUN_INIT_T (int (*)(sc_low_t, struct targ_info *, struct lun_info *, int))

Definition at line 520 of file scsi_low.h.

7.27.1.84 `#define SC_LOW_MSG_T (int (*)(sc_low_t, struct targ_info *, u_int))`

Definition at line 524 of file scsi_low.h.

7.27.1.85 `#define SC_LOW_NEXUS_T (int (*)(sc_low_t))`

Definition at line 523 of file scsi_low.h.

7.27.1.86 `#define SC_LOW_POLL_T (int (*)(void *))`

Definition at line 525 of file scsi_low.h.

7.27.1.87 `#define SC_LOW_POWER_T (int (*)(sc_low_t, u_int))`

Definition at line 526 of file scsi_low.h.

7.27.1.88 `#define SC_LOW_SELECT_T (int (*)(sc_low_t, struct slccb *))`

Definition at line 521 of file scsi_low.h.

7.27.1.89 `#define SC_LOW_TARG_INIT_T (int (*)(sc_low_t, struct targ_info *, int))`

Definition at line 519 of file scsi_low.h.

7.27.1.90 `#define SC_LOW_TIMEOUT_T (int (*)(sc_low_t))`

Definition at line 527 of file scsi_low.h.

7.27.1.91 `#define SCSI2_RESET_DELAY 500000`

Definition at line 198 of file scsi_low.h.

Referenced by scsi_low_bus_reset().

7.27.1.92 `#define SCSI_LOW_ALT_QTAG_ALLOCATE`

Definition at line 6 of file scsi_low.h.

7.27.1.93 `#define SCSI_LOW_ASSERT_ATN(slp)`

Value:

```
{
    (slp)->sl_atten = 1;
}
```

Definition at line 744 of file scsi_low.h.

7.27.1.94 #define SCSI_LOW_BITS DVF SCSI_BITS

Definition at line 178 of file scsi_low.h.

Referenced by scsi_low_calcf_show(), and scsi_low_print().

7.27.1.95 #define SCSI_LOW_BUS_WIDTH_16 1

Definition at line 487 of file scsi_low.h.

Referenced by scsi_low_calcf_target(), and scsi_low_msginfunc_i_wide_residue().

7.27.1.96 #define SCSI_LOW_BUS_WIDTH_32 2

Definition at line 488 of file scsi_low.h.

Referenced by scsi_low_msginfunc_i_wide_residue().

7.27.1.97 #define SCSI_LOW_BUS_WIDTH_8 0

Definition at line 486 of file scsi_low.h.

Referenced by scsi_low_alloc_ti(), scsi_low_calcf_target(), scsi_low_reset_nexus_target(), and scsi_low_wide().

7.27.1.98 #define SCSI_LOW_BZERO(pt, size) bzero((pt), (size))

Definition at line 93 of file scsi_low.h.

Referenced by scsi_low_alloc_li(), scsi_low_alloc_ti(), and scsi_low_sense_abort_start().

7.27.1.99 #define SCSI_LOW_DATA_PE 0x80000000

Definition at line 694 of file scsi_low.h.

Referenced by scsi_low_msgin().

7.27.1.100 #define SCSI_LOW_DEASSERT_ATN(slp)**Value:**

```
{
    (slp)->sl_atten = 0;
}
```

Definition at line 749 of file scsi_low.h.

Referenced by scsi_low_init_msgsys().

7.27.1.101 #define SCSI_LOW_DEBUGGER(dev) kdb_enter(dev)

Definition at line 90 of file scsi_low.h.

7.27.1.102 #define SCSI_LOW_DEFCFG DVF SCSI_DEFCFG

Definition at line 177 of file scsi_low.h.

7.27.1.103 #define SCSI_LOW_DELAY(mu) DELAY((mu))

Definition at line 91 of file scsi_low.h.

Referenced by scsi_low_attach(), scsi_low_poll(), and scsi_low_twiddle_wait().

7.27.1.104 #define SCSI_LOW_DIAGNOSTIC

Definition at line 5 of file scsi_low.h.

Referenced by scsi_low_msgout().

7.27.1.105 #define SCSI_LOW_DISC DVF SCSI_DISC

Definition at line 171 of file scsi_low.h.

Referenced by scsi_low_alloc_li(), scsi_low_calcf_lun(), and scsi_low_errfunc_identify().

7.27.1.106 #define SCSI_LOW_ENGAGE 2

Definition at line 515 of file scsi_low.h.

Referenced by scsi_low_engage(), and scsi_low_resume().

**7.27.1.107 #define SCSI_LOW_-
ERRORBITS "\020\017ubferr\016senseerr\015senseio\012uacaerr\011staterr\010busy\007parity\006m**

Definition at line 609 of file scsi_low.h.

Referenced by scsi_low_print().

7.27.1.108 #define SCSI_LOW_INFO_ALLOC 0

Definition at line 511 of file scsi_low.h.

Referenced by scsi_low_alloc_li(), and scsi_low_alloc_ti().

7.27.1.109 #define SCSI_LOW_INFO_DEALLOC 2

Definition at line 513 of file scsi_low.h.

Referenced by scsi_low_free_ti().

7.27.1.110 #define SCSI_LOW_INFO_REVOKE 1

Definition at line 512 of file scsi_low.h.

Referenced by scsi_low_reset_nexus_target().

7.27.1.111 #define SCSI_LOW_INTERFACE_CAM

Definition at line 57 of file scsi_low.h.

7.27.1.112 #define SCSI_LOW_LINK DVF SCSI_LINK

Definition at line 173 of file scsi_low.h.

Referenced by scsi_low_alloc_li(), scsi_low_calcf_lun(), and scsi_low_msginfunc_lcc().

7.27.1.113 #define SCSI_LOW_LUN_FLAGS_ALL_VALID**Value:**

```
(SCSI_LOW_LUN_FLAGS_USER_VALID | \  
    SCSI_LOW_LUN_FLAGS_DISK_VALID | SCSI_LOW_LUN_FLAGS_QUIRKS_VALID)
```

Definition at line 347 of file scsi_low.h.

Referenced by scsi_low_calcf_lun(), and scsi_low_setup_done().

7.27.1.114 #define SCSI_LOW_LUN_FLAGS_DISK_VALID 0x0002

Definition at line 345 of file scsi_low.h.

Referenced by scsi_low_reset_nexus_target(), and scsi_low_setup_done().

7.27.1.115 #define SCSI_LOW_LUN_FLAGS_QUIRKS_VALID 0x0004

Definition at line 346 of file scsi_low.h.

Referenced by scsi_low_alloc_li().

7.27.1.116 #define SCSI_LOW_LUN_FLAGS_USER_VALID 0x0001

Definition at line 344 of file scsi_low.h.

Referenced by scsi_low_alloc_li().

7.27.1.117 #define SCSI_LOW_LUN_INQ 0x02

Definition at line 335 of file scsi_low.h.

Referenced by scsi_low_setup_done(), and scsi_low_setup_start().

7.27.1.118 #define SCSI_LOW_LUN_MODEQ 0x03

Definition at line 336 of file scsi_low.h.

Referenced by scsi_low_setup_done(), and scsi_low_setup_start().

7.27.1.119 #define SCSI_LOW_LUN_OK 0x04

Definition at line 337 of file scsi_low.h.

Referenced by scsi_low_setup_done(), and scsi_low_start().

7.27.1.120 #define SCSI_LOW_LUN_SLEEP 0x00

Definition at line 333 of file scsi_low.h.

Referenced by scsi_low_reset_nexus_target(), scsi_low_setup_done(), and scsi_low_setup_start().

7.27.1.121 #define SCSI_LOW_LUN_START 0x01

Definition at line 334 of file scsi_low.h.

Referenced by scsi_low_setup_start().

7.27.1.122 #define SCSI_LOW_MAX_MSGLEN 32

Definition at line 201 of file scsi_low.h.

Referenced by scsi_low_msgin(), and scsi_low_msgout().

7.27.1.123 #define SCSI_LOW_MAX_PHCHANGES 256

Definition at line 197 of file scsi_low.h.

Referenced by scsi_low_msgin(), and scsi_low_msgout().

7.27.1.124 #define SCSI_LOW_MAX_RETRY 3

Definition at line 189 of file scsi_low.h.

Referenced by scsi_low_attach().

7.27.1.125 #define SCSI_LOW_MAX_SELECTION_RETRY 10

Definition at line 190 of file scsi_low.h.

Referenced by scsi_low_timeout_check().

7.27.1.126 #define SCSI_LOW_MAXNEXUS (sizeof(u_int) * NBBY)

Definition at line 319 of file scsi_low.h.

Referenced by scsi_low_alloc_qtag(), scsi_low_calcf_lun(), and scsi_low_dealloc_qtag().

7.27.1.127 #define SCSI_LOW_MIN_TOUT 12

Definition at line 194 of file scsi_low.h.

Referenced by `scsi_low_enqueue()`, `scsi_low_msginfunc_lcc()`, `scsi_low_revoke_ccb()`, and `scsi_low_start()`.

7.27.1.128 **#define SCSI_LOW_MSG_ABORT 0x00000020**

Definition at line 444 of file `scsi_low.h`.

Referenced by `scsi_low_abort_ccb()`, `scsi_low_cmd()`, `scsi_low_data()`, `scsi_low_msgfunc_identify()`, `scsi_low_msgin()`, `scsi_low_msginfunc_simple_qtag()`, `scsi_low_msgout()`, and `scsi_low_sense_abort_start()`.

7.27.1.129 **#define SCSI_LOW_MSG_ABORT_QTAG 0x00000400**

Definition at line 449 of file `scsi_low.h`.

Referenced by `scsi_low_abort_ccb()`, and `scsi_low_msginfunc_simple_qtag()`.

7.27.1.130 **#define SCSI_LOW_MSG_ALL 0xffffffff**

Definition at line 455 of file `scsi_low.h`.

Referenced by `scsi_low_msginfunc_msg_reject()`, and `scsi_low_msgout()`.

7.27.1.131 **#define SCSI_LOW_MSG_CLEAR_QTAG 0x00000800**

Definition at line 450 of file `scsi_low.h`.

7.27.1.132 **#define SCSI_LOW_MSG_ERROR 0x00000008**

Definition at line 442 of file `scsi_low.h`.

7.27.1.133 **#define SCSI_LOW_MSG_HEAD_QTAG 0x00000200**

Definition at line 448 of file `scsi_low.h`.

Referenced by `scsi_low_start()`.

7.27.1.134 **#define SCSI_LOW_MSG_IDENTIFY 0x00000010**

Definition at line 443 of file `scsi_low.h`.

Referenced by `scsi_low_start()`.

7.27.1.135 **#define SCSI_LOW_MSG_LAST 0x00008000**

Definition at line 454 of file `scsi_low.h`.

7.27.1.136 **#define SCSI_LOW_MSG_LOG_DATALEN 8**

Definition at line 202 of file `scsi_low.h`.

7.27.1.137 #define SCSI_LOW_MSG_NOOP 0x00004000

Definition at line 453 of file scsi_low.h.

Referenced by scsi_low_disconnected(), scsi_low_establish_ccb(), and scsi_low_msgout().

7.27.1.138 #define SCSI_LOW_MSG_ORDERED_QTAG 0x00000100

Definition at line 447 of file scsi_low.h.

Referenced by scsi_low_start().

7.27.1.139 #define SCSI_LOW_MSG_PARITY 0x00000004

Definition at line 441 of file scsi_low.h.

Referenced by scsi_low_msgin().

7.27.1.140 #define SCSI_LOW_MSG_REJECT 0x00000002

Definition at line 440 of file scsi_low.h.

Referenced by scsi_low_errfunc_qtag(), scsi_low_msgin(), scsi_low_msginfunc_ext(), scsi_low_msginfunc_lcc(), scsi_low_msginfunc_msg_reject(), scsi_low_msginfunc_parity(), scsi_low_msginfunc_rejop(), scsi_low_msginfunc_rp(), and scsi_low_msginfunc_sdp().

7.27.1.141 #define SCSI_LOW_MSG_RESET 0x00000001

Definition at line 439 of file scsi_low.h.

7.27.1.142 #define SCSI_LOW_MSG_SIMPLE_QTAG 0x00000080

Definition at line 446 of file scsi_low.h.

Referenced by scsi_low_start().

7.27.1.143 #define SCSI_LOW_MSG_SYNCH 0x00002000

Definition at line 452 of file scsi_low.h.

Referenced by scsi_low_calcf_target(), scsi_low_msginfunc_ext(), scsi_low_synch(), and scsi_low_wide().

7.27.1.144 #define SCSI_LOW_MSG_TERMIO 0x00000040

Definition at line 445 of file scsi_low.h.

7.27.1.145 #define SCSI_LOW_MSG_WIDE 0x00001000

Definition at line 451 of file scsi_low.h.

Referenced by scsi_low_calcf_target(), scsi_low_msginfunc_ext(), and scsi_low_wide().

7.27.1.146 #define SCSI_LOW_MSGOUT_INIT 0x00000001

Definition at line 689 of file scsi_low.h.

Referenced by scsi_low_msgout().

7.27.1.147 #define SCSI_LOW_MSGOUT_UNIFY 0x00000002

Definition at line 690 of file scsi_low.h.

Referenced by scsi_low_msgout().

7.27.1.148 #define SCSI_LOW_NCCB 128

Definition at line 187 of file scsi_low.h.

Referenced by scsi_low_attach().

7.27.1.149 #define SCSI_LOW_NOPARITY DVF SCSI_NOPARITY

Definition at line 175 of file scsi_low.h.

Referenced by scsi_low_calcf_lun().

7.27.1.150 #define SCSI_LOW_NTARGETS 8

Definition at line 185 of file scsi_low.h.

Referenced by scsi_low_attach().

7.27.1.151 #define SCSI_LOW_OFFSET(n) DVF SCSI_OFFSET(n)

Definition at line 181 of file scsi_low.h.

7.27.1.152 #define SCSI_LOW_PERIOD(n) DVF SCSI_PERIOD(n)

Definition at line 180 of file scsi_low.h.

7.27.1.153 #define SCSI_LOW_POWDOWN 1

Definition at line 514 of file scsi_low.h.

Referenced by scsi_low_timeout_check().

7.27.1.154 #define SCSI_LOW_POWDOWN_TC 15

Definition at line 196 of file scsi_low.h.

Referenced by scsi_low_init(), and scsi_low_timeout_check().

7.27.1.155 #define SCSI_LOW_QFLAG_CA_QCLEAR 0x01

Definition at line 327 of file scsi_low.h.

Referenced by scsi_low_msginfunc_cc(), and scsi_low_setup_done().

7.27.1.156 #define SCSI_LOW_QTAG DVF SCSI_QTAG

Definition at line 174 of file scsi_low.h.

Referenced by scsi_low_alloc_li(), scsi_low_calcf_lun(), and scsi_low_errfunc_qtag().

7.27.1.157 #define SCSI_LOW_READ 1

Definition at line 218 of file scsi_low.h.

Referenced by scsi_low_cmd(), scsi_low_sense_abort_start(), scsi_low_setup_start(), and scsi_low_unit_ready_cmd().

7.27.1.158 #define SCSI_LOW_RESTART_HARD 1

Definition at line 717 of file scsi_low.h.

Referenced by scsi_low_activate(), scsi_low_attach(), scsi_low_reselected(), and scsi_low_timeout_check().

7.27.1.159 #define SCSI_LOW_RESTART_SOFT 0

Definition at line 718 of file scsi_low.h.

Referenced by scsi_low_init().

7.27.1.160 #define SCSI_LOW_RWUNK (-1)

Definition at line 216 of file scsi_low.h.

Referenced by scsi_low_bus_release().

7.27.1.161 #define SCSI_LOW_SAVESP DVF SCSI_SAVESP

Definition at line 176 of file scsi_low.h.

7.27.1.162 #define SCSI_LOW_SETUP_MSGPHASE(slp, PHASE)**Value:**

```
{
    (slp)->sl_msgphase = (PHASE);
}
```

Definition at line 739 of file scsi_low.h.

Referenced by `scsi_low_init_msgsys()`, `scsi_low_msgfunc_abort()`, `scsi_low_msgfunc_qabort()`, `scsi_low_msgfunc_reset()`, `scsi_low_msginfunc_cc()`, `scsi_low_msginfunc_disc()`, and `scsi_low_msginfunc_lcc()`.

7.27.1.163 #define SCSI_LOW_SETUP_PHASE(ti, phase)

Value:

```
{
    (ti)->ti_ophase = ti->ti_phase;
    (ti)->ti_phase = (phase);
}
```

Definition at line 733 of file `scsi_low.h`.

Referenced by `scsi_low_bus_release()`, and `scsi_low_reselected()`.

7.27.1.164 #define SCSI_LOW_SPLSCSI splcam

Definition at line 92 of file `scsi_low.h`.

Referenced by `scsi_low_activate()`, `scsi_low_attach()`, `scsi_low_deactivate()`, `scsi_low_dettach()`, `scsi_low_engage()`, and `scsi_low_timeout()`.

7.27.1.165 #define SCSI_LOW_START_FAIL 1

Definition at line 510 of file `scsi_low.h`.

7.27.1.166 #define SCSI_LOW_START_OK 0

Definition at line 509 of file `scsi_low.h`.

Referenced by `scsi_low_start()`.

7.27.1.167 #define SCSI_LOW_SYNC DVF SCSI_SYNC

Definition at line 170 of file `scsi_low.h`.

Referenced by `scsi_low_alloc_li()`, and `scsi_low_calcf_lun()`.

7.27.1.168 #define SCSI_LOW_TARG_FLAGS_ALL_VALID

Value:

```
(SCSI_LOW_TARG_FLAGS_USER_VALID | \
    SCSI_LOW_TARG_FLAGS_DISK_VALID | SCSI_LOW_TARG_FLAGS_QUIRKS_VALID)
```

Definition at line 474 of file `scsi_low.h`.

Referenced by `scsi_low_calcf_target()`.

7.27.1.169 #define SCSI_LOW_TARG_FLAGS_DISK_VALID 0x0002

Definition at line 472 of file scsi_low.h.

Referenced by scsi_low_reset_nexus_target(), and scsi_low_setup_done().

7.27.1.170 #define SCSI_LOW_TARG_FLAGS_QUIRKS_VALID 0x0004

Definition at line 473 of file scsi_low.h.

Referenced by scsi_low_alloc_ti().

7.27.1.171 #define SCSI_LOW_TARG_FLAGS_USER_VALID 0x0001

Definition at line 471 of file scsi_low.h.

Referenced by scsi_low_alloc_ti().

7.27.1.172 #define SCSI_LOW_TIMEOUT_CH_ENGAGE 1

Definition at line 145 of file scsi_low.h.

Referenced by scsi_low_deactivate(), scsi_low_engage(), scsi_low_init(), and scsi_low_resume().

7.27.1.173 #define SCSI_LOW_TIMEOUT_CH_IO 0

Definition at line 144 of file scsi_low.h.

Referenced by scsi_low_activate(), scsi_low_attach(), scsi_low_deactivate(), and scsi_low_timeout().

7.27.1.174 #define SCSI_LOW_TIMEOUT_CH_RECOVER 2

Definition at line 146 of file scsi_low.h.

7.27.1.175 #define SCSI_LOW_TIMEOUT_CHECK_INTERVAL 1

Definition at line 195 of file scsi_low.h.

Referenced by scsi_low_timeout_check().

7.27.1.176 #define SCSI_LOW_TIMEOUT_HZ 10

Definition at line 193 of file scsi_low.h.

Referenced by scsi_low_poll(), and scsi_low_timeout_check().

7.27.1.177 #define SCSI_LOW_TIMEOUT_START 1

Definition at line 143 of file scsi_low.h.

Referenced by scsi_low_activate(), scsi_low_attach(), scsi_low_engage(), scsi_low_resume(), and scsi_low_timeout().

7.27.1.178 #define SCSI_LOW_TIMEOUT_STOP 0

Definition at line 142 of file scsi_low.h.

Referenced by scsi_low_deactivate(), and scsi_low_init().

7.27.1.179 #define SCSI_LOW_UNKLUN ((u_int) -1)

Definition at line 229 of file scsi_low.h.

7.27.1.180 #define SCSI_LOW_UNKTAG ((scsi_low_tag_t) -1)

Definition at line 230 of file scsi_low.h.

Referenced by scsi_low_abort_ccb(), scsi_low_activate_qtag(), scsi_low_alloc_qtag(), scsi_low_deactivate_qtag(), scsi_low_dealloc_qtag(), scsi_low_enqueue(), scsi_low_msgfunc_identify(), scsi_low_msgfunc_qtag(), scsi_low_msgin(), and scsi_low_msginfunc_lcc().

7.27.1.181 #define SCSI_LOW_WAIT DVF SCSI_WAIT

Definition at line 172 of file scsi_low.h.

7.27.1.182 #define SCSI_LOW_WRITE 0

Definition at line 217 of file scsi_low.h.

7.27.1.183 #define SELTIMEOUTIO 0x0008

Definition at line 598 of file scsi_low.h.

Referenced by scsi_low_disconnected(), scsi_low_setup_done(), and scsi_low_timeout_check().

7.27.1.184 #define SENSEERR 0x2000

Definition at line 606 of file scsi_low.h.

Referenced by scsi_low_done().

7.27.1.185 #define SENSEIO 0x1000

Definition at line 605 of file scsi_low.h.

Referenced by scsi_low_done(), and scsi_low_setup_done().

7.27.1.186 #define SHOW_ALL_NEG -1

Definition at line 642 of file scsi_low.h.

7.27.1.187 #define SHOW_CALCF_RES 0x0010

Definition at line 640 of file scsi_low.h.

Referenced by scsi_low_setup_done().

7.27.1.188 #define SHOW_PROBE_RES 0x0020

Definition at line 641 of file scsi_low.h.

Referenced by scsi_low_start_up().

7.27.1.189 #define SHOW_SYNCH_NEG 0x0001

Definition at line 638 of file scsi_low.h.

Referenced by scsi_low_synch().

7.27.1.190 #define SHOW_WIDE_NEG 0x0002

Definition at line 639 of file scsi_low.h.

Referenced by scsi_low_wide().

7.27.1.191 #define sl_dev sl_si.si_dev

Definition at line 548 of file scsi_low.h.

7.27.1.192 #define ST_BUSY 0x08

Referenced by scsi_low_done(), and scsi_low_msginfunc_cc().

7.27.1.193 #define ST_CHKCOND 0x02

Referenced by scsi_low_done(), and scsi_low_msginfunc_cc().

7.27.1.194 #define ST_CMDTERM 0x22

Referenced by scsi_low_done().

7.27.1.195 #define ST_CONFLICT 0x18

Referenced by scsi_low_done().

7.27.1.196 #define ST_GOOD 0x00

Referenced by scsi_low_disconnected(), scsi_low_done(), and scsi_low_msginfunc_cc().

7.27.1.197 #define ST_INTERGOOD 0x10

Referenced by `scsi_low_done()`, `scsi_low_msginfunc_cc()`, and `scsi_low_msginfunc_lcc()`.

7.27.1.198 #define ST_INTERMET 0x14

Referenced by `scsi_low_done()`, `scsi_low_msginfunc_cc()`, and `scsi_low_msginfunc_lcc()`.

7.27.1.199 #define ST_MET 0x04

Referenced by `scsi_low_done()`.

7.27.1.200 #define ST_QUEFULL 0x28

Referenced by `scsi_low_done()`, and `scsi_low_msginfunc_cc()`.

7.27.1.201 #define ST_UNKNOWN 0xff

Referenced by `scsi_low_done()`, `scsi_low_msginfunc_lcc()`, and `scsi_low_start()`.

7.27.1.202 #define STATERR 0x0100

Definition at line 603 of file `scsi_low.h`.

Referenced by `scsi_low_done()`.

7.27.1.203 #define TIMEOUTIO 0x0004

Definition at line 597 of file `scsi_low.h`.

Referenced by `scsi_low_timeout_check()`.

7.27.1.204 #define UACAERR 0x0200

Definition at line 604 of file `scsi_low.h`.

Referenced by `scsi_low_done()`.

7.27.1.205 #define UBFERR 0x4000

Definition at line 607 of file `scsi_low.h`.

Referenced by `scsi_low_disconnected()`.

7.27.2 Typedef Documentation**7.27.2.1 typedef struct [scsi_low_softc](#)* [sc_low_t](#)**

Definition at line 507 of file `scsi_low.h`.

7.27.2.2 typedef struct [scsi_sense_data](#) [scsi_low_osdep_sense_data_t](#)

Definition at line 115 of file [scsi_low.h](#).

7.27.2.3 typedef int [scsi_low_tag_t](#)

Definition at line 226 of file [scsi_low.h](#).

7.27.3 Function Documentation

7.27.3.1 LIST_HEAD ([lun_info_tab](#), [lun_info](#))

7.27.3.2 LIST_HEAD ([scsi_low_softc_tab](#), [scsi_low_softc](#))

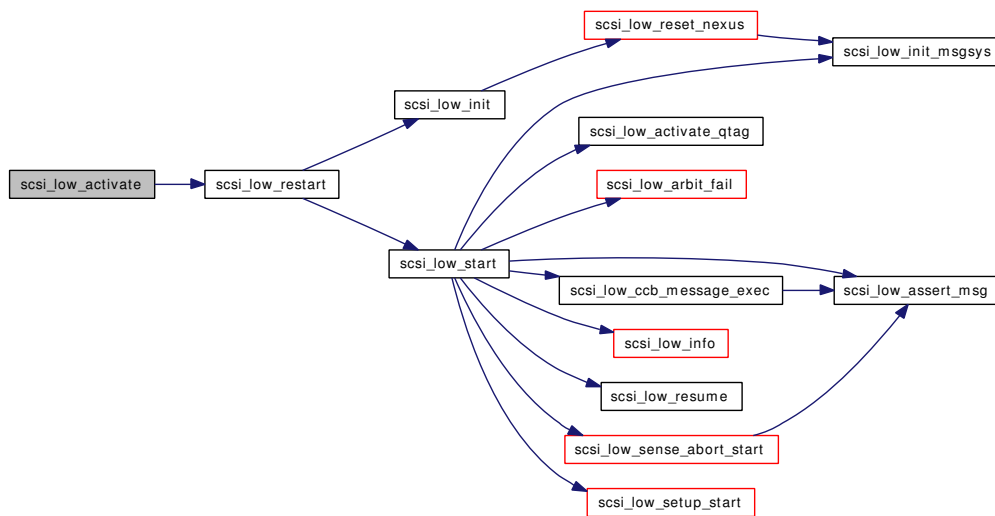
7.27.3.3 int [scsi_low_activate](#) (struct [scsi_low_softc](#) *)

Definition at line 1551 of file [scsi_low.c](#).

References [HW_INACTIVE](#), [scsi_low_restart\(\)](#), [SCSI_LOW_RESTART_HARD](#), [SCSI_LOW_SPLSCSI](#), [SCSI_LOW_TIMEOUT_CH_IO](#), and [SCSI_LOW_TIMEOUT_START](#).

Referenced by [scsi_low_activate_pisa\(\)](#).

Here is the call graph for this function:



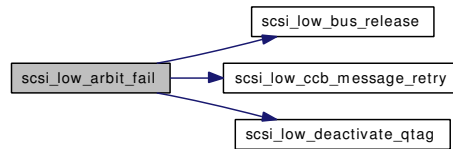
7.27.3.4 void [scsi_low_arbit_fail](#) (struct [scsi_low_softc](#) *, struct [slccb](#) *)

Definition at line 2485 of file [scsi_low.c](#).

References `slccb::ccb_flags`, `slccb::ccb_selrent`, `CCB_STARTQ`, `scsi_low_bus_release()`, `scsi_low_ccb_message_retry()`, `scsi_low_deactivate_qtag()`, `slccb::ti`, and `ti`.

Referenced by `scsi_low_reselected()`, and `scsi_low_start()`.

Here is the call graph for this function:



7.27.3.5 `static __inline void scsi_low_arbit_win (struct scsi_low_softc *)` [static]

7.27.3.6 `static __inline int scsi_low_assert_msg (struct scsi_low_softc *, struct targ_info *, u_int, int)` [static]

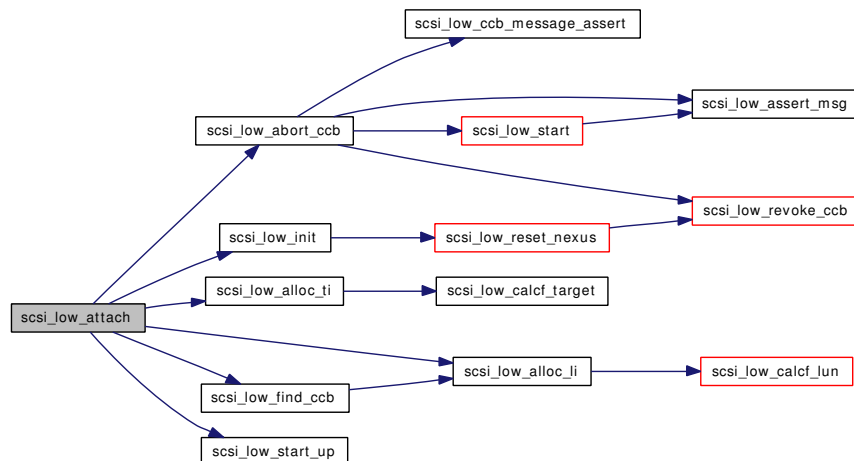
Referenced by `scsi_low_abort_ccb()`, `scsi_low_ccb_message_exec()`, `scsi_low_cmd()`, `scsi_low_data()`, `scsi_low_msgfunc_identify()`, `scsi_low_msgin()`, `scsi_low_msginfunc_ext()`, `scsi_low_msginfunc_lcc()`, `scsi_low_msginfunc_parity()`, `scsi_low_msginfunc_rejop()`, `scsi_low_msginfunc_rp()`, `scsi_low_msginfunc_sdp()`, `scsi_low_msginfunc_simple_qtag()`, `scsi_low_msgout()`, `scsi_low_sense_abort_start()`, `scsi_low_start()`, and `scsi_low_test_atten()`.

7.27.3.7 `int scsi_low_attach (struct scsi_low_softc *, int, int, int, int, int)`

Definition at line 2023 of file `scsi_low.c`.

References `scsi_low_abort_ccb()`, `scsi_low_alloc_li()`, `scsi_low_alloc_ti()`, `SCSI_LOW_DELAY`, `scsi_low_find_ccb()`, `scsi_low_init()`, `SCSI_LOW_MAX_RETRY`, `SCSI_LOW_NCCB`, `SCSI_LOW_NTARGETS`, `SCSI_LOW_RESTART_HARD`, `SCSI_LOW_SPLSCSI`, `scsi_low_start_up()`, `SCSI_LOW_TIMEOUT_CH_IO`, `SCSI_LOW_TIMEOUT_START`, `sl_tab`, `ti`, and `targ_info::ti_lunsize`.

Here is the call graph for this function:



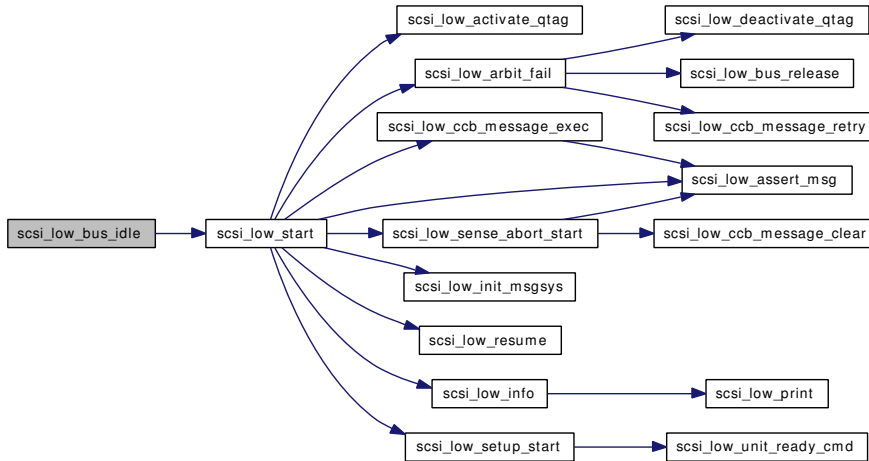
7.27.3.8 `static __inline void scsi_low_attention (struct scsi_low_softc *)` [static]

7.27.3.9 `void scsi_low_bus_idle (struct scsi_low_softc *)`

Definition at line 1832 of file `scsi_low.c`.

References `scsi_low_start()`.

Here is the call graph for this function:



7.27.3.10 void `scsi_low_bus_reset` (struct `scsi_low_softc *`)

Definition at line 2919 of file `scsi_low.c`.

References `SCSI2_RESET_DELAY`, `scsi_low_twiddle_wait()`, and `TWIDDLEWAIT`.

Here is the call graph for this function:

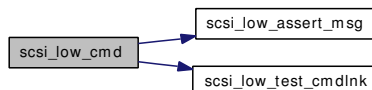


7.27.3.11 int `scsi_low_cmd` (struct `scsi_low_softc *`, struct `targ_info *`)

Definition at line 3094 of file `scsi_low.c`.

References `FATALIO`, `scsi_low_assert_msg()`, `SCSI_LOW_CMDLNK_CHECK`, `SCSI_LOW_DEBUG_TEST_GO`, `SCSI_LOW_INFO`, `SCSI_LOW_MSG_ABORT`, `SCSI_LOW_READ`, `scsi_low_test_cmdlnk()`, `ti`, and `targ_info::ti_id`.

Here is the call graph for this function:



7.27.3.12 int `scsi_low_data` (struct `scsi_low_softc *`, struct `targ_info *`, struct `buf **`, int)

Definition at line 3131 of file `scsi_low.c`.

References `slccb::bp`, `slccb::ccb_ssep`, `FATALIO`, `PDMAERR`, `sc_p::scp_direction`, `scsi_low_assert_msg()`, `SCSI_LOW_INFO`, `SCSI_LOW_MSG_ABORT`, `ti`, `targ_info::ti_ophase`, and `targ_info::ti_phase`.

Here is the call graph for this function:



7.27.3.13 `static __inline void scsi_low_data_finish (struct scsi_low_softc *)` `[static]`

Referenced by `scsi_low_msginfunc_i_wide_residue()`.

7.27.3.14 `int scsi_low_deactivate (struct scsi_low_softc *)`

Definition at line 1535 of file `scsi_low.c`.

References `HW_INACTIVE`, `SCSI_LOW_SPLSCSI`, `SCSI_LOW_TIMEOUT_CH_ENGAGE`, `SCSI_LOW_TIMEOUT_CH_IO`, and `SCSI_LOW_TIMEOUT_STOP`.

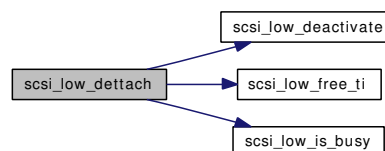
Referenced by `scsi_low_deactivate_pisa()`, and `scsi_low_dettach()`.

7.27.3.15 `int scsi_low_dettach (struct scsi_low_softc *)`

Definition at line 2119 of file `scsi_low.c`.

References `scsi_low_deactivate()`, `scsi_low_free_ti()`, `scsi_low_is_busy()`, and `SCSI_LOW_SPLSCSI`.

Here is the call graph for this function:

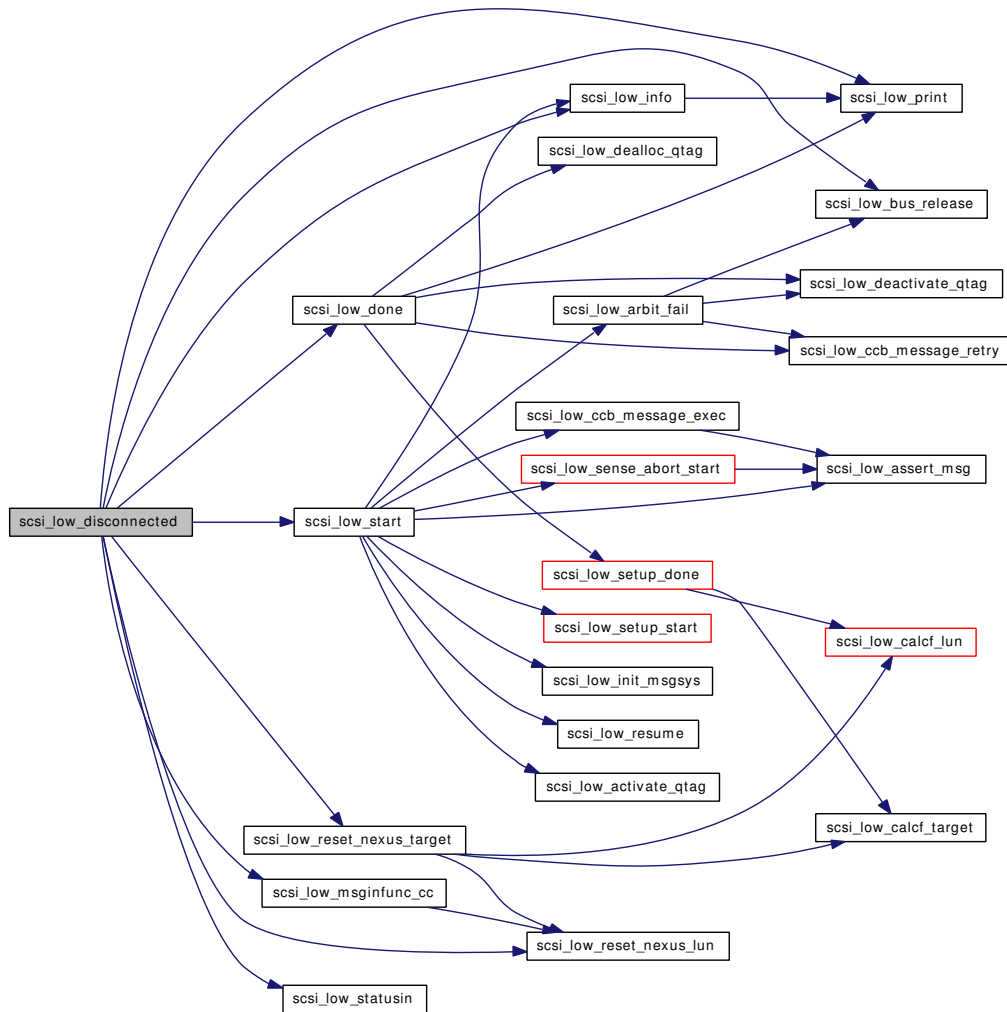


7.27.3.16 `int scsi_low_disconnected (struct scsi_low_softc *, struct targ_info *)`

Definition at line 4191 of file `scsi_low.c`.

References `CCB_DISCQ`, `slccb::ccb_error`, `slccb::ccb_flags`, `slccb::ccb_msgoutflag`, `slccb::ccb_omsgoutflag`, `CCB_STARTQ`, `FATALIO`, `slccb::li`, `lun_info::li_disc`, `lun_info::li_discq`, `MSGPH_ABORT`, `MSGPH_CMDC`, `MSGPH_DISC`, `MSGPH_LCTERM`, `MSGPH_NULL`, `MSGPH_RESET`, `MSGPH_TERM`, `PH_SELSTART`, `SCSI_LOW_ATTEN_CHECK`, `scsi_low_bus_release()`, `SCSI_LOW_DEBUG_DISC`, `SCSI_LOW_DEBUG_GO`, `SCSI_LOW_DEBUG_TEST_GO`, `scsi_low_done()`, `SCSI_LOW_DONE_RETRY`, `scsi_low_info()`, `SCSI_LOW_MSG_NOOP`, `scsi_low_msginfunc_cc()`, `scsi_low_print()`, `scsi_low_reset_nexus_lun()`, `scsi_low_reset_nexus_target()`, `scsi_low_start()`, `scsi_low_statusin()`, `SELTIMEOUTIO`, `ST_GOOD`, `ti`, `targ_info::ti_disc`, `targ_info::ti_id`, `targ_info::ti_msgflags`, `targ_info::ti_phase`, and `UBFERR`.

Here is the call graph for this function:



7.27.3.17 int scsi_low_is_busy (struct [scsi_low_softc](#) *)

Definition at line 1525 of file `scsi_low.c`.

Referenced by `scsi_low_dettach()`.

7.27.3.18 static __inline int scsi_low_is_disconnect_ok (struct [slccb](#) *) [static]

Referenced by `scsi_low_msgfunc_identify()`.

7.27.3.19 static __inline int scsi_low_is_msgout_continue ([ti](#), [mask](#)) [static]

7.27.3.20 static __inline int scsi_low_is_msgout_continue (struct [targ_info](#) *, [u_int](#)) [static]

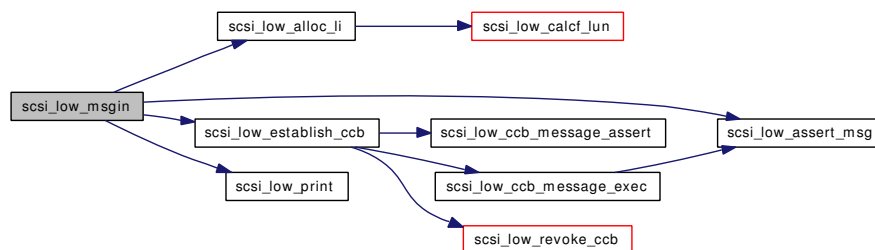
Referenced by `scsi_low_msgout()`.

7.27.3.21 int scsi_low_msgin (struct scsi_low_softc *, struct targ_info *, u_int)

Definition at line 4047 of file scsi_low.c.

References FATALIO, lun_info::li_lun, lun_info::li_nqio, scsi_low_msgin_data::md_len, scsi_low_msgin_data::md_msgfunc, MSG_IDENTIFY, MSGCMD_LUN, MSGIN_DATA_LAST, MSGINPTR_CLR, scsi_low_alloc_li(), scsi_low_assert_msg(), SCSI_LOW_DATA_PE, SCSI_LOW_DEBUG_TEST_GO, scsi_low_establish_ccb(), SCSI_LOW_INFO, SCSI_LOW_MAX_MSGLEN, SCSI_LOW_MAX_PHCHANGES, SCSI_LOW_MSG_ABORT, SCSI_LOW_MSG_PARITY, SCSI_LOW_MSG_REJECT, SCSI_LOW_NEXUS_CHECK, scsi_low_print(), SCSI_LOW_UNKTAG, ti, targ_info::ti_id, targ_info::ti_log_msgin, targ_info::ti_msgin, targ_info::ti_msgin_parity_error, targ_info::ti_msginlen, targ_info::ti_msginptr, targ_info::ti_ophase, and targ_info::ti_phase.

Here is the call graph for this function:

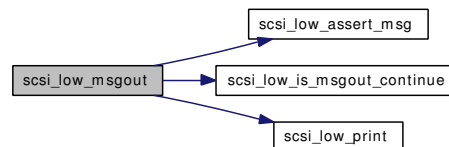


7.27.3.22 int scsi_low_msgout (struct scsi_low_softc *, struct targ_info *, u_int)

Definition at line 3471 of file scsi_low.c.

References FATALIO, scsi_low_msgout_data::md_condition, scsi_low_msgout_data::md_flags, scsi_low_msgout_data::md_msg, scsi_low_msgout_data::md_msgfunc, MSG_RELEASE_ATN, scsi_low_assert_msg(), SCSI_LOW_DIAGNOSTIC, scsi_low_is_msgout_continue(), SCSI_LOW_MAX_MSGLEN, SCSI_LOW_MAX_PHCHANGES, SCSI_LOW_MSG_ABORT, SCSI_LOW_MSG_ALL, SCSI_LOW_MSG_NOOP, scsi_low_msgout_data, SCSI_LOW_MSGOUT_INIT, SCSI_LOW_MSGOUT_UNIFY, scsi_low_print(), ti, targ_info::ti_emsgflags, targ_info::ti_log_msgout, targ_info::ti_msgflags, targ_info::ti_msgoutlen, targ_info::ti_msgoutstr, and targ_info::ti_omsgflags.

Here is the call graph for this function:



7.27.3.23 void scsi_low_print (struct scsi_low_softc *, struct targ_info *)

Definition at line 4765 of file scsi_low.c.

References lun_info::li_discq, sc_p::scp_data, sc_p::scp_datalen, sc_p::scp_status, SCSI_LOW_BITS, SCSI_LOW_ERRORBITS, ti, and targ_info::ti_litab.

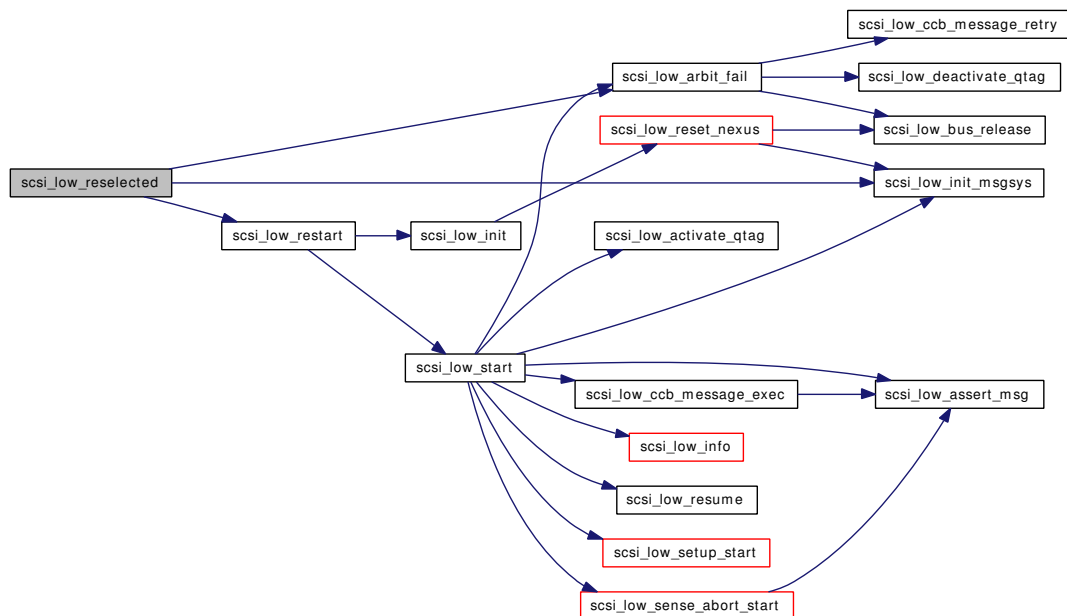
Referenced by `scsi_low_disconnected()`, `scsi_low_done()`, `scsi_low_info()`, `scsi_low_msgin()`, and `scsi_low_msgout()`.

7.27.3.24 `struct targ_info* scsi_low_reselected (struct scsi_low_softc *, u_int)`

Definition at line 3019 of file `scsi_low.c`.

References `PH_DISC`, `PH_NULL`, `PH_RESEL`, `scsi_low_arbit_fail()`, `scsi_low_init_msgsys()`, `scsi_low_restart()`, `SCSI_LOW_RESTART_HARD`, `SCSI_LOW_SETUP_PHASE`, `ti`, and `targ_info::ti_phase`.

Here is the call graph for this function:



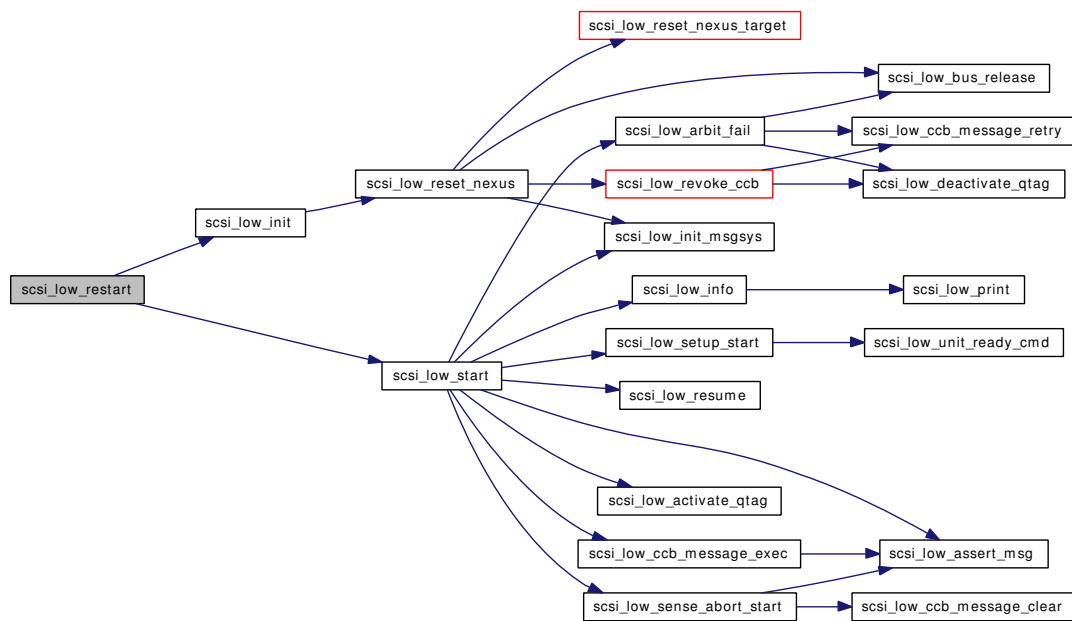
7.27.3.25 `int scsi_low_restart (struct scsi_low_softc *, int, u_char *)`

Definition at line 2934 of file `scsi_low.c`.

References `scsi_low_init()`, and `scsi_low_start()`.

Referenced by `scsi_low_activate()`, and `scsi_low_reselected()`.

Here is the call graph for this function:



7.27.3.26 `static __inline int scsi_low_statusin (struct scsi_low_softc *, struct targ_info *, u_int) [static]`

Referenced by `scsi_low_disconnected()`.

7.27.3.27 `TAILQ_HEAD (targ_info_tab, targ_info)`

7.27.4 Variable Documentation

7.27.4.1 `u_int c`

Definition at line 829 of file `scsi_low.h`.

Referenced by `copy_voltag()`.

7.27.4.2 `u_int mask`

Definition at line 765 of file `scsi_low.h`.

7.27.4.3 `u_int msg`

Definition at line 796 of file `scsi_low.h`.

7.27.4.4 `int now`

Definition at line 797 of file `scsi_low.h`.

7.27.4.5 struct `targ_info*` `ti`

Definition at line 828 of file `scsi_low.h`.

7.27.4.6 struct `targ_info*` `ti`

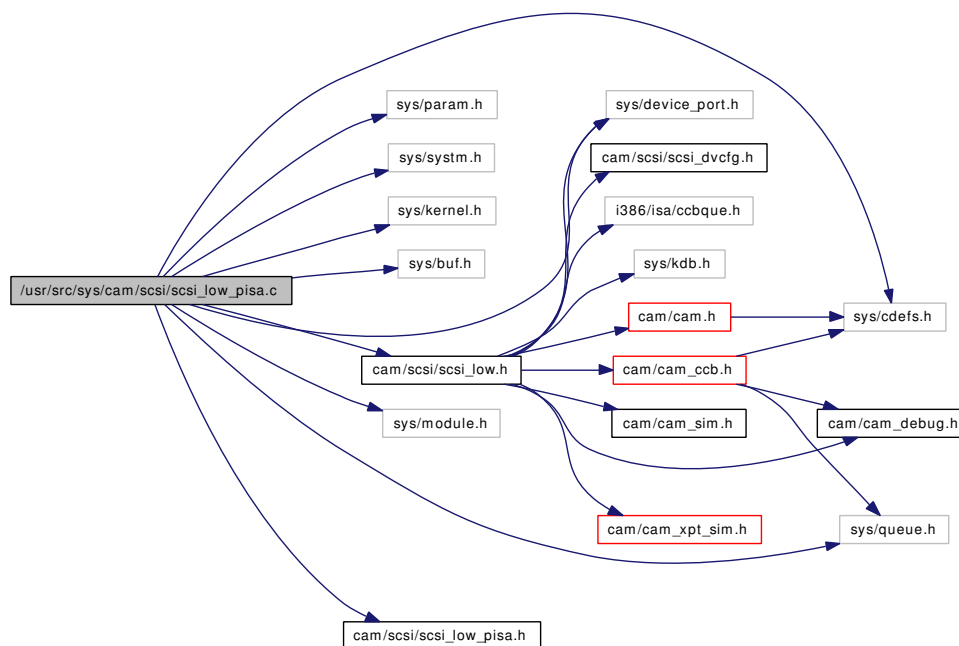
Definition at line 795 of file `scsi_low.h`.

Referenced by `scsi_low_abort_ccb()`, `scsi_low_alloc_ti()`, `scsi_low_arbit_fail()`, `scsi_low_attach()`, `scsi_low_bus_release()`, `scsi_low_calcf_lun()`, `scsi_low_calcf_show()`, `scsi_low_cmd()`, `scsi_low_data()`, `scsi_low_disconnected()`, `scsi_low_enqueue()`, `scsi_low_errfunc_synch()`, `scsi_low_errfunc_wide()`, `scsi_low_find_ccb()`, `scsi_low_free_ti()`, `scsi_low_info()`, `scsi_low_init_msgsys()`, `scsi_low_message_enqueue()`, `scsi_low_msgfunc_identify()`, `scsi_low_msgfunc_qtag()`, `scsi_low_msgfunc_synch()`, `scsi_low_msgfunc_wide()`, `scsi_low_msgin()`, `scsi_low_msginfunc_ext()`, `scsi_low_msginfunc_i_wide_residue()`, `scsi_low_msginfunc_lcc()`, `scsi_low_msginfunc_msg_reject()`, `scsi_low_msginfunc_parity()`, `scsi_low_msginfunc_rejop()`, `scsi_low_msginfunc_simple_qtag()`, `scsi_low_msgout()`, `scsi_low_print()`, `scsi_low_reselected()`, `scsi_low_reset_nexus()`, `scsi_low_reset_nexus_target()`, `scsi_low_revoke_ccb()`, `scsi_low_sense_abort_start()`, `scsi_low_setup_done()`, `scsi_low_start()`, `scsi_low_start_up()`, `scsi_low_synch()`, `scsi_low_test_attn()`, `scsi_low_timeout_check()`, and `scsi_low_wide()`.

7.28 /usr/src/sys/cam/scsi/scsi_low_pisa.c File Reference

```
#include <sys/cdefs.h>
#include <sys/param.h>
#include <sys/system.h>
#include <sys/kernel.h>
#include <sys/buf.h>
#include <sys/queue.h>
#include <sys/device_port.h>
#include <sys/module.h>
#include <cam/scsi/scsi_low.h>
#include <cam/scsi/scsi_low_pisa.h>
```

Include dependency graph for scsi_low_pisa.c:



Functions

- `__FBSDID` ("\$FreeBSD: src/sys/cam/scsi/scsi_low_pisa.c,v 1.8 2005/01/05 22:34:34 imp Exp \$")
- `int scsi_low_deactivate_pisa` (struct `scsi_low_softc` *sc)
- `int scsi_low_activate_pisa` (struct `scsi_low_softc` *sc, int flags)
- `DECLARE_MODULE` (scsi_low, `scsi_low_moduledata`, SI_SUB_DRIVERS, SI_ORDER_MIDDLE)
- `MODULE_VERSION` (scsi_low, 1)
- `MODULE_DEPEND` (scsi_low, cam, 1, 1, 1)

Variables

- static moduledata_t [scsi_low_moduledata](#)

7.28.1 Function Documentation

7.28.1.1 `__FBSDID` ("\$FreeBSD: src/sys/cam/scsi/scsi_low_pisa.c, v 1.8 2005/01/05 22:34:34 imp Exp \$")

7.28.1.2 `DECLARE_MODULE` (scsi_low, [scsi_low_moduledata](#), SI_SUB_DRIVERS, SI_ORDER_MIDDLE)

7.28.1.3 `MODULE_DEPEND` (scsi_low, cam, 1, 1, 1)

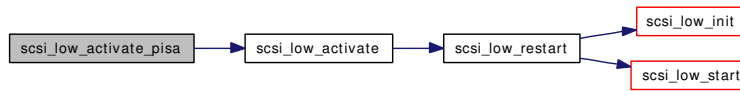
7.28.1.4 `MODULE_VERSION` (scsi_low, 1)

7.28.1.5 `int scsi_low_activate_pisa` (struct [scsi_low_softc](#) * *sc*, int *flags*)

Definition at line 146 of file `scsi_low_pisa.c`.

References `scsi_low_activate()`, and `scsi_low_softc::sl_cfgflags`.

Here is the call graph for this function:



7.28.1.6 `int scsi_low_deactivate_pisa` (struct [scsi_low_softc](#) * *sc*)

Definition at line 136 of file `scsi_low_pisa.c`.

References `scsi_low_deactivate()`.

Here is the call graph for this function:



7.28.2 Variable Documentation

7.28.2.1 `moduledata_t scsi_low_moduledata` [static]

Initial value:

```

{
    "scsi_low",
    NULL,
    NULL
}

```

Definition at line 159 of file `scsi_low_pisa.c`.

7.29 /usr/src/sys/cam/scsi/scsi_low_pisa.h File Reference

This graph shows which files directly or indirectly include this file:



Functions

- [int scsi_low_activate_pisa](#) (struct [scsi_low_softc](#) *, int)
- [int scsi_low_deactivate_pisa](#) (struct [scsi_low_softc](#) *)

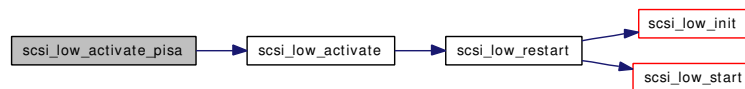
7.29.1 Function Documentation

7.29.1.1 int scsi_low_activate_pisa (struct [scsi_low_softc](#) *, int)

Definition at line 146 of file [scsi_low_pisa.c](#).

References [scsi_low_activate\(\)](#), and [scsi_low_softc::sl_cfgflags](#).

Here is the call graph for this function:



7.29.1.2 int scsi_low_deactivate_pisa (struct [scsi_low_softc](#) *)

Definition at line 136 of file [scsi_low_pisa.c](#).

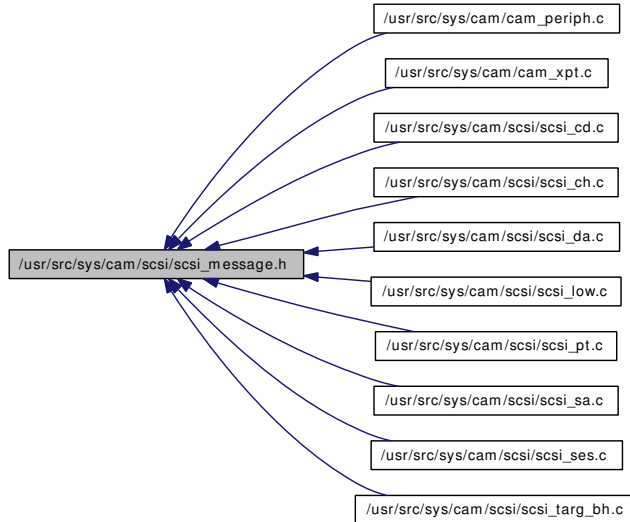
References [scsi_low_deactivate\(\)](#).

Here is the call graph for this function:



7.30 /usr/src/sys/cam/scsi/scsi_message.h File Reference

This graph shows which files directly or indirectly include this file:



Defines

- #define [MSG_CMDCOMPLETE](#) 0x00
- #define [MSG_TASK_COMPLETE](#) 0x00 /* M/M */
- #define [MSG_EXTENDED](#) 0x01
- #define [MSG_SAVEDATAPOINTER](#) 0x02
- #define [MSG_RESTOREPOINTERS](#) 0x03
- #define [MSG_DISCONNECT](#) 0x04
- #define [MSG_INITIATOR_DET_ERR](#) 0x05
- #define [MSG_ABORT](#) 0x06
- #define [MSG_ABORT_TASK_SET](#) 0x06 /* O/M */
- #define [MSG_MESSAGE_REJECT](#) 0x07
- #define [MSG_NOOP](#) 0x08
- #define [MSG_PARITY_ERROR](#) 0x09
- #define [MSG_LINK_CMD_COMPLETE](#) 0x0a
- #define [MSG_LINK_CMD_COMPLETEF](#) 0x0b /* O/O */
- #define [MSG_BUS_DEV_RESET](#) 0x0c
- #define [MSG_TARGET_RESET](#) 0x0c /* O/M */
- #define [MSG_ABORT_TAG](#) 0x0d
- #define [MSG_ABORT_TASK](#) 0x0d /* O/O */
- #define [MSG_CLEAR_QUEUE](#) 0x0e
- #define [MSG_CLEAR_TASK_SET](#) 0x0e /* O/O */
- #define [MSG_INIT_RECOVERY](#) 0x0f /* O/O */
- #define [MSG_REL_RECOVERY](#) 0x10 /* O/O */
- #define [MSG_TERM_IO_PROC](#) 0x11 /* O/O */
- #define [MSG_CLEAR_ACA](#) 0x16 /* O/O */
- #define [MSG_LOGICAL_UNIT_RESET](#) 0x17 /* O/O */
- #define [MSG_QAS_REQUEST](#) 0x55 /* O/O */

- #define `MSG_SIMPLE_Q_TAG` 0x20
- #define `MSG_SIMPLE_TASK` 0x20 /* O/O */
- #define `MSG_HEAD_OF_Q_TAG` 0x21
- #define `MSG_HEAD_OF_QUEUE_TASK` 0x21 /* O/O */
- #define `MSG_ORDERED_Q_TAG` 0x22
- #define `MSG_ORDERED_TASK` 0x22 /* O/O */
- #define `MSG_IGN_WIDE_RESIDUE` 0x23
- #define `MSG_ACA_TASK` 0x24 /* 0/0 */
- #define `MSG_IDENTIFYFLAG` 0x80
- #define `MSG_IDENTIFY_DISCFLAG` 0x40
- #define `MSG_IDENTIFY`(lun, disc) (((disc) ? 0xc0 : MSG_IDENTIFYFLAG) | (lun))
- #define `MSG_ISIDENTIFY`(m) ((m) & MSG_IDENTIFYFLAG)
- #define `MSG_IDENTIFY_LUNMASK` 0x3F
- #define `MSG_EXT_SDTR` 0x01
- #define `MSG_EXT_SDTR_LEN` 0x03
- #define `MSG_EXT_WDTR` 0x03
- #define `MSG_EXT_WDTR_LEN` 0x02
- #define `MSG_EXT_WDTR_BUS_8_BIT` 0x00
- #define `MSG_EXT_WDTR_BUS_16_BIT` 0x01
- #define `MSG_EXT_WDTR_BUS_32_BIT` 0x02
- #define `MSG_EXT_PPR` 0x04
- #define `MSG_EXT_PPR_LEN` 0x06
- #define `MSG_EXT_PPR_PCOMP_EN` 0x80
- #define `MSG_EXT_PPR_RTI` 0x40
- #define `MSG_EXT_PPR_RD_STRM` 0x20
- #define `MSG_EXT_PPR_WR_FLOW` 0x10
- #define `MSG_EXT_PPR_HOLD_MCS` 0x08
- #define `MSG_EXT_PPR_QAS_REQ` 0x04
- #define `MSG_EXT_PPR_DT_REQ` 0x02
- #define `MSG_EXT_PPR_IU_REQ` 0x01

7.30.1 Define Documentation

7.30.1.1 #define `MSG_ABORT` 0x06

Definition at line 14 of file `scsi_message.h`.

7.30.1.2 #define `MSG_ABORT_TAG` 0x0d

Definition at line 23 of file `scsi_message.h`.

7.30.1.3 #define `MSG_ABORT_TASK` 0x0d /* O/O */

Definition at line 24 of file `scsi_message.h`.

7.30.1.4 #define `MSG_ABORT_TASK_SET` 0x06 /* O/M */

Definition at line 15 of file `scsi_message.h`.

7.30.1.5 #define MSG_ACA_TASK 0x24 /* 0/0 */

Definition at line 42 of file scsi_message.h.

7.30.1.6 #define MSG_BUS_DEV_RESET 0x0c

Definition at line 21 of file scsi_message.h.

7.30.1.7 #define MSG_CLEAR_ACA 0x16 /* O/O */

Definition at line 30 of file scsi_message.h.

7.30.1.8 #define MSG_CLEAR_QUEUE 0x0e

Definition at line 25 of file scsi_message.h.

7.30.1.9 #define MSG_CLEAR_TASK_SET 0x0e /* O/O */

Definition at line 26 of file scsi_message.h.

7.30.1.10 #define MSG_CMDCOMPLETE 0x00

Definition at line 7 of file scsi_message.h.

7.30.1.11 #define MSG_DISCONNECT 0x04

Definition at line 12 of file scsi_message.h.

7.30.1.12 #define MSG_EXT_PPR 0x04

Definition at line 61 of file scsi_message.h.

7.30.1.13 #define MSG_EXT_PPR_DT_REQ 0x02

Definition at line 69 of file scsi_message.h.

Referenced by xpt_announce_periph(), and xpt_set_transfer_settings().

7.30.1.14 #define MSG_EXT_PPR_HOLD_MCS 0x08

Definition at line 67 of file scsi_message.h.

7.30.1.15 #define MSG_EXT_PPR_IU_REQ 0x01

Definition at line 70 of file scsi_message.h.

Referenced by xpt_set_transfer_settings().

7.30.1.16 #define MSG_EXT_PPR_LEN 0x06

Definition at line 62 of file scsi_message.h.

7.30.1.17 #define MSG_EXT_PPR_PCOMP_EN 0x80

Definition at line 63 of file scsi_message.h.

7.30.1.18 #define MSG_EXT_PPR_QAS_REQ 0x04

Definition at line 68 of file scsi_message.h.

Referenced by xpt_set_transfer_settings().

7.30.1.19 #define MSG_EXT_PPR_RD_STRM 0x20

Definition at line 65 of file scsi_message.h.

7.30.1.20 #define MSG_EXT_PPR_RTI 0x40

Definition at line 64 of file scsi_message.h.

7.30.1.21 #define MSG_EXT_PPR_WR_FLOW 0x10

Definition at line 66 of file scsi_message.h.

7.30.1.22 #define MSG_EXT_SDTR 0x01

Definition at line 52 of file scsi_message.h.

7.30.1.23 #define MSG_EXT_SDTR_LEN 0x03

Definition at line 53 of file scsi_message.h.

7.30.1.24 #define MSG_EXT_WDTR 0x03

Definition at line 55 of file scsi_message.h.

7.30.1.25 #define MSG_EXT_WDTR_BUS_16_BIT 0x01

Definition at line 58 of file scsi_message.h.

Referenced by xpt_set_transfer_settings().

7.30.1.26 #define MSG_EXT_WDTR_BUS_32_BIT 0x02

Definition at line 59 of file scsi_message.h.

Referenced by xpt_set_transfer_settings().

7.30.1.27 #define MSG_EXT_WDTR_BUS_8_BIT 0x00

Definition at line 57 of file scsi_message.h.

Referenced by xpt_set_transfer_settings().

7.30.1.28 #define MSG_EXT_WDTR_LEN 0x02

Definition at line 56 of file scsi_message.h.

7.30.1.29 #define MSG_EXTENDED 0x01

Definition at line 9 of file scsi_message.h.

7.30.1.30 #define MSG_HEAD_OF_Q_TAG 0x21

Definition at line 37 of file scsi_message.h.

7.30.1.31 #define MSG_HEAD_OF_QUEUE_TASK 0x21 /* O/O */

Definition at line 38 of file scsi_message.h.

7.30.1.32 #define MSG_IDENTIFY(lun, disc) (((disc) ? 0xc0 : MSG_IDENTIFYFLAG) | (lun))

Definition at line 47 of file scsi_message.h.

7.30.1.33 #define MSG_IDENTIFY_DISCFLAG 0x40

Definition at line 46 of file scsi_message.h.

7.30.1.34 #define MSG_IDENTIFY_LUNMASK 0x3F

Definition at line 49 of file scsi_message.h.

7.30.1.35 #define MSG_IDENTIFYFLAG 0x80

Definition at line 45 of file scsi_message.h.

7.30.1.36 #define MSG_IGN_WIDE_RESIDUE 0x23

Definition at line 41 of file scsi_message.h.

7.30.1.37 #define MSG_INIT_RECOVERY 0x0f /* O/O */

Definition at line 27 of file scsi_message.h.

7.30.1.38 #define MSG_INITIATOR_DET_ERR 0x05

Definition at line 13 of file scsi_message.h.

7.30.1.39 #define MSG_ISIDENTIFY(m) ((m) & MSG_IDENTIFYFLAG)

Definition at line 48 of file scsi_message.h.

7.30.1.40 #define MSG_LINK_CMD_COMPLETE 0x0a

Definition at line 19 of file scsi_message.h.

7.30.1.41 #define MSG_LINK_CMD_COMPLETEF 0x0b /* O/O */

Definition at line 20 of file scsi_message.h.

7.30.1.42 #define MSG_LOGICAL_UNIT_RESET 0x17 /* O/O */

Definition at line 31 of file scsi_message.h.

7.30.1.43 #define MSG_MESSAGE_REJECT 0x07

Definition at line 16 of file scsi_message.h.

7.30.1.44 #define MSG_NOOP 0x08

Definition at line 17 of file scsi_message.h.

7.30.1.45 #define MSG_ORDERED_Q_TAG 0x22

Definition at line 39 of file scsi_message.h.

Referenced by dadump(), and dastart().

7.30.1.46 #define MSG_ORDERED_TASK 0x22 /* O/O */

Definition at line 40 of file scsi_message.h.

7.30.1.47 #define MSG_PARITY_ERROR 0x09

Definition at line 18 of file scsi_message.h.

7.30.1.48 #define MSG_QAS_REQUEST 0x55 /* O/O */

Definition at line 32 of file scsi_message.h.

7.30.1.49 #define MSG_REL_RECOVERY 0x10 /* O/O */

Definition at line 28 of file scsi_message.h.

7.30.1.50 #define MSG_RESTOREPOINTERS 0x03

Definition at line 11 of file scsi_message.h.

7.30.1.51 #define MSG_SAVEDATAPOINTER 0x02

Definition at line 10 of file scsi_message.h.

7.30.1.52 #define MSG_SIMPLE_Q_TAG 0x20

Definition at line 35 of file scsi_message.h.

Referenced by camperiphscsisenseerror(), cdgetmode(), cdpause(), cdplay(), cdplaymsf(), cdplaytracks(), cdprevent(), cdreadvdstructure(), cdreadsubchannel(), cdreadtoc(), cdreportkey(), cdsendkey(), cdsetmode(), cdsetspeed(), cdsizes(), cdstart(), cdstartunit(), cdstopunit(), chexchange(), chgetelemstatus(), chgetparams(), chielem(), chmove(), chposition(), chstart(), daclose(), dadump(), dagetcapacity(), daprevent(), dashutdown(), dastart(), probestart(), ptstart(), saerase(), sagetparams(), saloadunload(), samount(), saprevent(), sardpos(), sareservereleaseunit(), saretension(), sarewind(), sasetparams(), sasetpos(), saspace(), sastart(), sawritefilemarks(), ses_runcmd(), and targbhstart().

7.30.1.53 #define MSG_SIMPLE_TASK 0x20 /* O/O */

Definition at line 36 of file scsi_message.h.

7.30.1.54 #define MSG_TARGET_RESET 0x0c /* O/M */

Definition at line 22 of file scsi_message.h.

7.30.1.55 #define MSG_TASK_COMPLETE 0x00 /* M/M */

Definition at line 8 of file scsi_message.h.

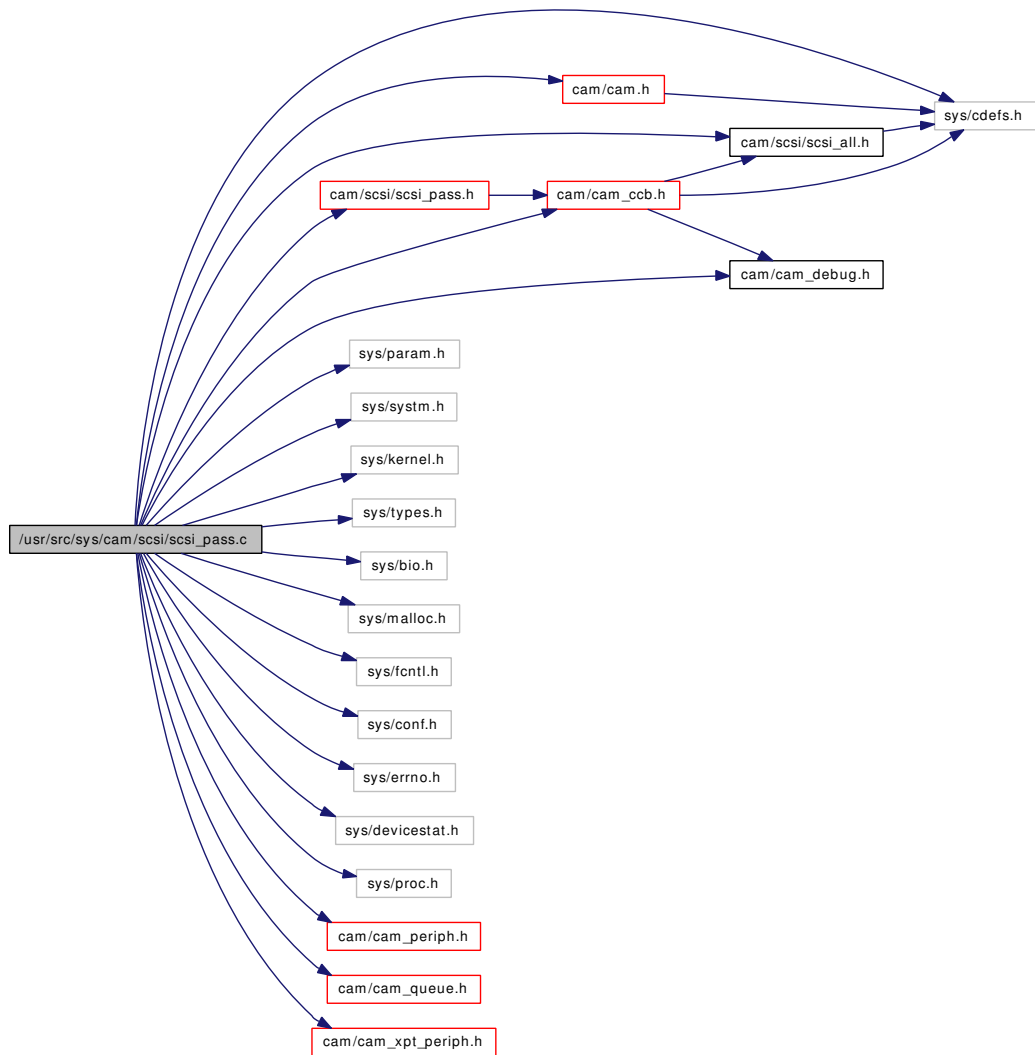
7.30.1.56 #define MSG_TERM_IO_PROC 0x11 /* O/O */

Definition at line 29 of file scsi_message.h.

7.31 /usr/src/sys/cam/scsi/scsi_pass.c File Reference

```
#include <sys/cdefs.h>
#include <sys/param.h>
#include <sys/system.h>
#include <sys/kernel.h>
#include <sys/types.h>
#include <sys/bio.h>
#include <sys/malloc.h>
#include <sys/fcntl.h>
#include <sys/conf.h>
#include <sys/errno.h>
#include <sys/devicestat.h>
#include <sys/proc.h>
#include <cam/cam.h>
#include <cam/cam_ccb.h>
#include <cam/cam_periph.h>
#include <cam/cam_queue.h>
#include <cam/cam_xpt_periph.h>
#include <cam/cam_debug.h>
#include <cam/scsi/scsi_all.h>
#include <cam/scsi/scsi_pass.h>
```

Include dependency graph for scsi_pass.c:



Data Structures

- struct [pass_softc](#)

Defines

- #define [ccb_type](#) ppriv_field0
- #define [ccb_bp](#) ppriv_ptr1

Enumerations

- enum [pass_flags](#) { [PASS_FLAG_OPEN](#) = 0x01, [PASS_FLAG_LOCKED](#) = 0x02, [PASS_FLAG_INVALID](#) = 0x04 }
- enum [pass_state](#) { [PASS_STATE_NORMAL](#) }

- enum `pass_ccb_types` { `PASS_CCB_BUFFER_IO`, `PASS_CCB_WAITING` }

Functions

- `__FBSDID` ("\$FreeBSD: src/sys/cam/scsi/scsi_pass.c,v 1.45 2006/12/05 07:45:28 mjacob Exp \$")
- static void `passasync` (void *callback_arg, u_int32_t code, struct `cam_path` *path, void *arg)
- static void `passdone` (struct `cam_periph` *periph, union `ccb` *done_ccb)
- static int `passerror` (union `ccb` *ccb, u_int32_t `cam_flags`, u_int32_t `sense_flags`)
- static int `passendccb` (struct `cam_periph` *periph, union `ccb` *ccb, union `ccb` *inccb)
- `PERIPHDRIVER_DECLARE` (pass, `passdriver`)
- static void `passinit` (void)
- static void `passoninvalidate` (struct `cam_periph` *periph)
- static void `passcleanup` (struct `cam_periph` *periph)
- static `cam_status` `passregister` (struct `cam_periph` *periph, void *arg)
- static int `passopen` (struct `cdev` *dev, int flags, int fmt, struct `thread` *td)
- static int `passclose` (struct `cdev` *dev, int flag, int fmt, struct `thread` *td)
- static void `passstart` (struct `cam_periph` *periph, union `ccb` *start_ccb)
- static int `passioctl` (struct `cdev` *dev, u_long cmd, `caddr_t` addr, int flag, struct `thread` *td)

Variables

- static `d_open_t` `passopen`
- static `d_close_t` `passclose`
- static `d_ioctl_t` `passioctl`
- static `periph_init_t` `passinit`
- static `periph_ctor_t` `passregister`
- static `periph_oninv_t` `passoninvalidate`
- static `periph_dtor_t` `passcleanup`
- static `periph_start_t` `passstart`
- static struct `periph_driver` `passdriver`
- static struct `cdevsw` `pass_cdevsw`

7.31.1 Define Documentation

7.31.1.1 #define `ccb_bp ppriv_ptr1`

Definition at line 69 of file `scsi_pass.c`.

7.31.1.2 #define `ccb_type ppriv_field0`

Definition at line 68 of file `scsi_pass.c`.

7.31.2 Enumeration Type Documentation

7.31.2.1 enum `pass_ccb_types`

Enumerator:

`PASS_CCB_BUFFER_IO`

PASS_CCB_WAITING

Definition at line 63 of file scsi_pass.c.

7.31.2.2 enum `pass_flags`**Enumerator:***PASS_FLAG_OPEN**PASS_FLAG_LOCKED**PASS_FLAG_INVALID*

Definition at line 53 of file scsi_pass.c.

7.31.2.3 enum `pass_state`**Enumerator:***PASS_STATE_NORMAL*

Definition at line 59 of file scsi_pass.c.

7.31.3 Function Documentation

7.31.3.1 `__FBSDID` ("FreeBSD: src/sys/cam/scsi/scsi_pass. c, v 1.45 2006/12/05 07:45:28 mjacob Exp \$")

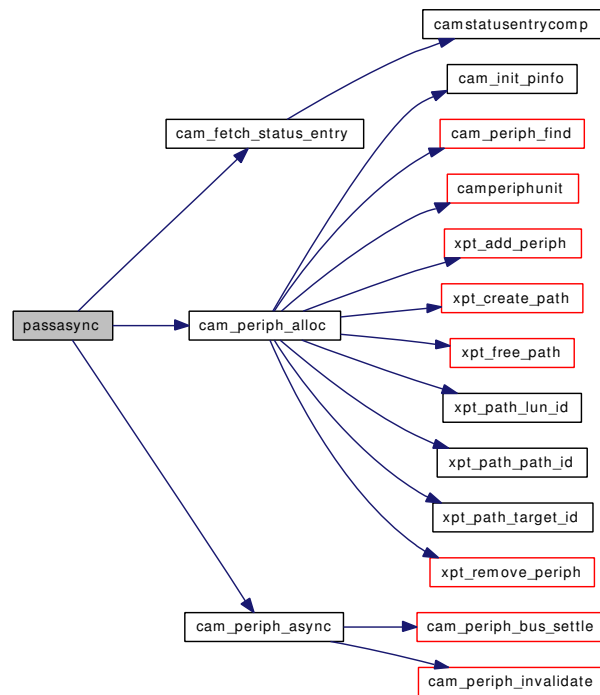
7.31.3.2 `static void passasync` (`void * callback_arg`, `u_int32_t code`, `struct cam_path * path`, `void * arg`) [`static`]

Definition at line 200 of file scsi_pass.c.

References `AC_FOUND_DEVICE`, `cam_fetch_status_entry()`, `cam_periph_alloc()`, `cam_periph_async()`, `CAM_PERIPH_BIO`, `CAM_REQ_CMP`, `CAM_REQ_INPROG`, `ccb_getdev::ccb_h`, `passcleanup`, `passoninvalidate`, `passregister`, `passstart`, `ccb_hdr::path`, and `cam_status_entry::status_text`.

Referenced by `passinit()`, `passoninvalidate()`, and `passregister()`.

Here is the call graph for this function:

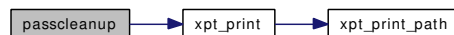


7.31.3.3 static void passcleanup (struct cam_periph * periph) [static]

Definition at line 183 of file scsi_pass.c.

References pass_softc::dev, pass_softc::device_stats, cam_periph::path, cam_periph::softc, and xpt_print().

Here is the call graph for this function:

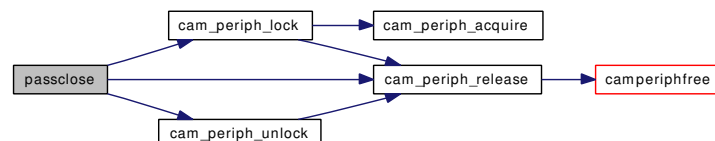


7.31.3.4 static int passclose (struct cdev * dev, int flag, int fmt, struct thread * td) [static]

Definition at line 385 of file scsi_pass.c.

References cam_periph_lock(), cam_periph_release(), cam_periph_unlock(), pass_softc::flags, PASS_FLAG_OPEN, and cam_periph::softc.

Here is the call graph for this function:



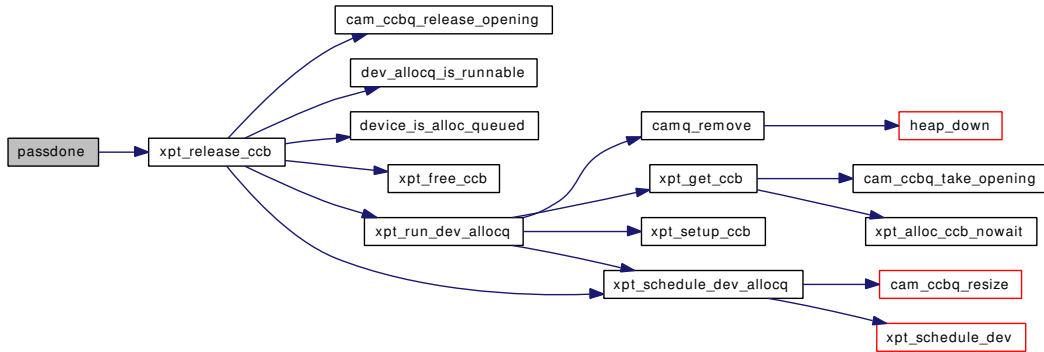
7.31.3.5 static void passdone (struct cam_periph * periph, union ccb * done_ccb) [static]

Definition at line 430 of file scsi_pass.c.

References ccb_hdr::cbfcnp, ccb::ccb_h, ccb_scsiio::ccb_h, ccb::csio, PASS_CCB_WAITING, cam_periph::softc, and xpt_release_ccb().

Referenced by passsendccb().

Here is the call graph for this function:



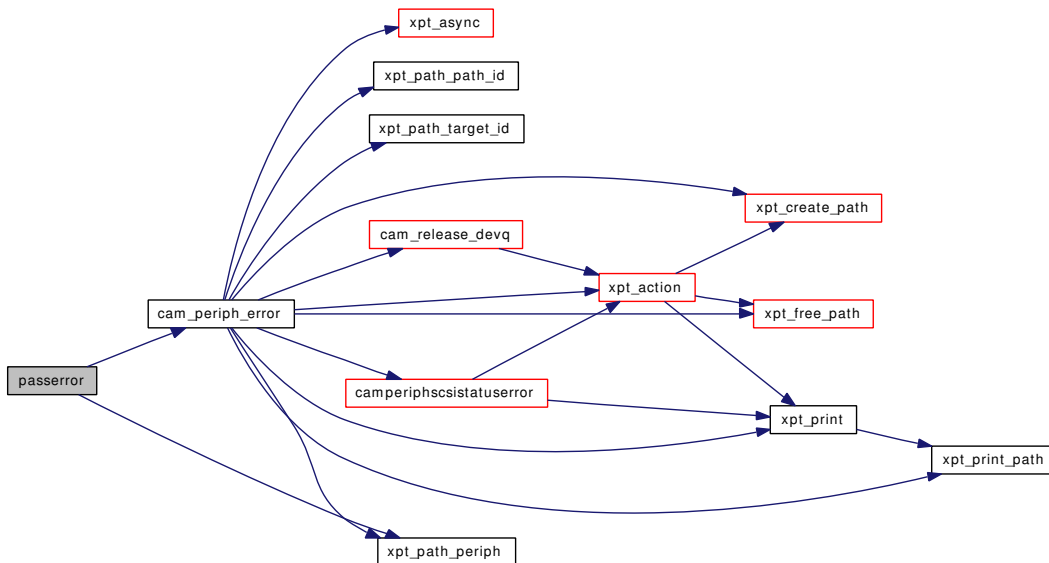
7.31.3.6 static int passerror (union ccb * ccb, u_int32_t cam_flags, u_int32_t sense_flags) [static]

Definition at line 610 of file scsi_pass.c.

References cam_periph_error(), ccb::ccb_h, ccb_hdr::path, pass_softc::saved_ccb, cam_periph::softc, and xpt_path_periph().

Referenced by passioctl(), and passsendccb().

Here is the call graph for this function:

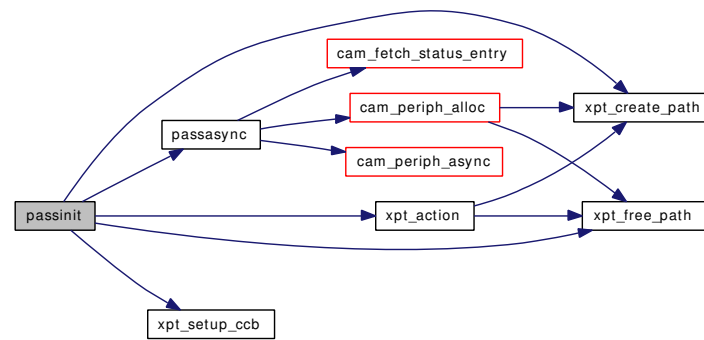


7.31.3.7 static void passinit (void) [static]

Definition at line 117 of file scsi_pass.c.

References AC_FOUND_DEVICE, ccb_setasync::callback, ccb_setasync::callback_arg, CAM_LUN_WILDCARD, CAM_REQ_CMP, CAM_TARGET_WILDCARD, CAM_XPT_PATH_ID, ccb::ccb_h, ccb_setasync::ccb_h, ccb_setasync::event_enable, ccb_hdr::func_code, passasync(), ccb_hdr::status, xpt_action(), xpt_create_path(), xpt_free_path(), XPT_SASYNC_CB, and xpt_setup_ccb().

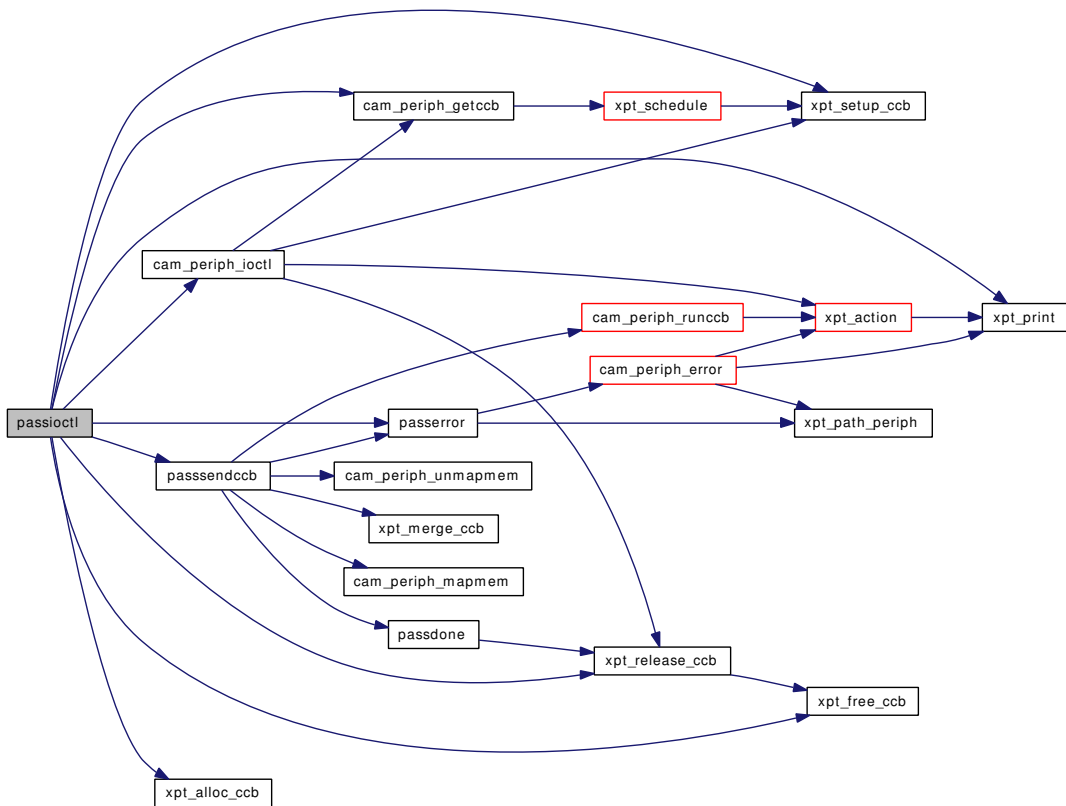
Here is the call graph for this function:

**7.31.3.8 static int passioctl (struct cdev * dev, u_long cmd, caddr_t addr, int flag, struct thread * td) [static]**

Definition at line 447 of file scsi_pass.c.

References `cam_periph_getccb()`, `cam_periph_ioctl()`, CAMIOCOMMAND, `ccb::ccb_h`, `ccb_hdr::func_code`, `passerror()`, `passsendccb()`, `cam_periph::path`, `ccb_hdr::pinfo`, `cam_pinfo::priority`, `cam_periph::softc`, `xpt_alloc_ccb()`, XPT_FC_QUEUED, XPT_FC_USER_CCB, XPT_FC_XPT_ONLY, `xpt_free_ccb()`, `xpt_print()`, `xpt_release_ccb()`, and `xpt_setup_ccb()`.

Here is the call graph for this function:

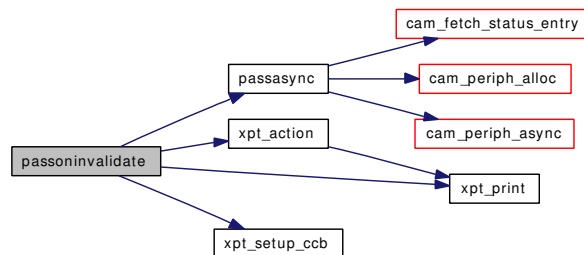


7.31.3.9 static void passoninvalidate (struct `cam_periph` * `periph`) [static]

Definition at line 150 of file `scsi_pass.c`.

References `ccb_setasync::callback`, `ccb_setasync::callback_arg`, `ccb_setasync::ccb_h`, `ccb_setasync::event_enable`, `ccb_hdr::func_code`, `PASS_FLAG_INVALID`, `passasync()`, `cam_periph::path`, `cam_periph::softc`, `xpt_action()`, `xpt_print()`, `XPT_SASYNC_CB`, and `xpt_setup_ccb()`.

Here is the call graph for this function:

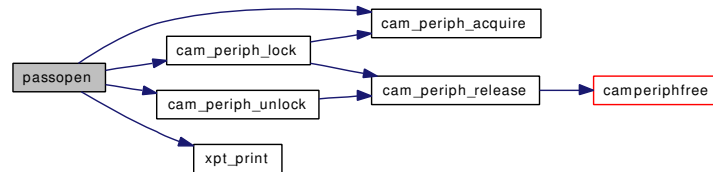


7.31.3.10 static int passopen (struct cdev * dev, int flags, int fmt, struct thread * td) [static]

Definition at line 319 of file scsi_pass.c.

References `cam_periph_acquire()`, `cam_periph_lock()`, `cam_periph_unlock()`, `CAM_REQ_CMP`, `pass_softc::flags`, `PASS_FLAG_INVALID`, `PASS_FLAG_OPEN`, `cam_periph::path`, `cam_periph::softc`, and `xpt_print()`.

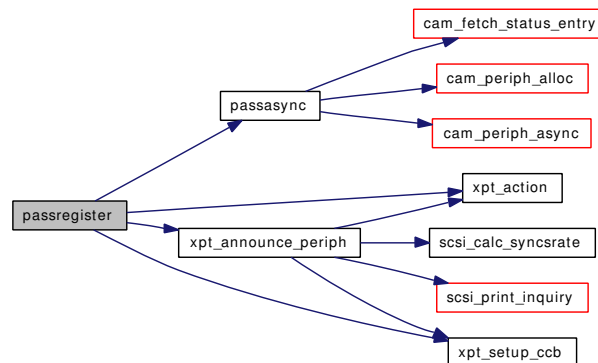
Here is the call graph for this function:

**7.31.3.11 static cam_status passregister (struct cam_periph * periph, void * arg) [static]**

Definition at line 247 of file scsi_pass.c.

References `AC_LOST_DEVICE`, `ccb_setasync::callback`, `ccb_setasync::callback_arg`, `CAM_REQ_CMP`, `CAM_REQ_CMP_ERR`, `ccb_setasync::ccb_h`, `ccb_setasync::event_enable`, `scsi_inquiry_data::flags`, `ccb_hdr::func_code`, `ccb_getdev::inq_data`, `pass_cdevsw`, `PASS_STATE_NORMAL`, `passasync()`, `cam_periph::path`, `cam_periph::periph_name`, `SID_CmdQue`, `SID_TYPE`, `cam_periph::softc`, `cam_periph::unit_number`, `xpt_action()`, `xpt_announce_periph()`, `XPT_SASYNC_CB`, and `xpt_setup_ccb()`.

Here is the call graph for this function:

**7.31.3.12 static int passsendccb (struct cam_periph * periph, union ccb * ccb, union ccb * inccb) [static]**

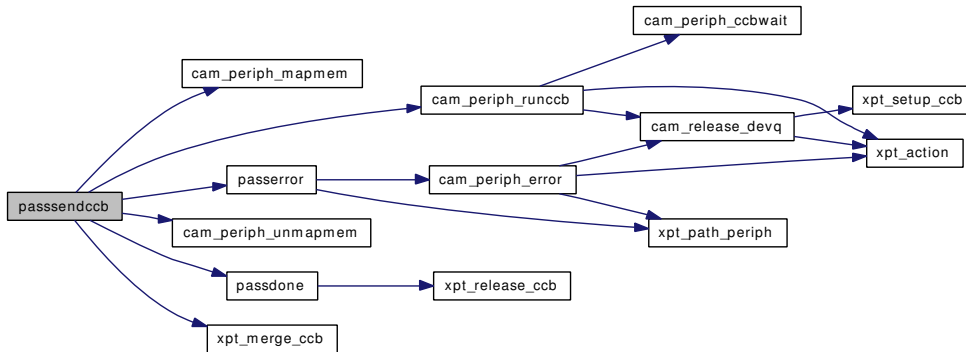
Definition at line 531 of file scsi_pass.c.

References `CAM_DATA_PHYS`, `CAM_DIR_MASK`, `CAM_DIR_NONE`, `CAM_PASS_ERR_RECOVER`, `cam_periph_mapmem()`, `cam_periph_runccb()`, `cam_periph_unmapmem()`, `CAM_RETRY_SELTO`, `ccb_hdr::cbfcnp`, `ccb::ccb_h`, `pass_softc::device_stats`, `ccb_hdr::flags`, `ccb_hdr::func_code`,

passdone(), passerror(), ccb_hdr::periph_priv, SF_RETRY_UA, cam_periph::softc, XPT_DEV_MATCH, xpt_merge_ccb(), and XPT_SCSI_IO.

Referenced by passioctl().

Here is the call graph for this function:



7.31.3.13 static void passtart (struct [cam_periph](#) * *periph*, union [ccb](#) * *start_ccb*) [static]

Definition at line 409 of file `scsi_pass.c`.

References `CAM_PRIORITY_NONE`, `ccb::ccb_h`, `cam_periph::immediate_priority`, `PASS_CCB_WAITING`, `PASS_STATE_NORMAL`, `cam_periph::softc`, and `pass_softc::state`.

7.31.3.14 PERIPHDRIIVER_DECLARE (pass, [passdriver](#))

7.31.4 Variable Documentation

7.31.4.1 struct [cdevsw](#) [pass_cdevsw](#) [static]

Initial value:

```

{
    .d_version =    D_VERSION,
    .d_flags =     D_NEEDGIANT,
    .d_open =      passopen,
    .d_close =     passclose,
    .d_ioctl =     passioctl,
    .d_name =      "pass",
}

```

Definition at line 107 of file `scsi_pass.c`.

Referenced by `passregister()`.

7.31.4.2 [periph_dtor_t](#) [passcleanup](#) [static]

Definition at line 88 of file `scsi_pass.c`.

Referenced by `passasync()`.

7.31.4.3 d_close_t passclose [static]

Definition at line 82 of file scsi_pass.c.

7.31.4.4 struct [periph_driver](#) passdriver [static]

Initial value:

```
{
    passinit, "pass",
    TAILQ_HEAD_INITIALIZER(passdriver.units), 0
}
```

Definition at line 99 of file scsi_pass.c.

7.31.4.5 [periph_init_t](#) passinit [static]

Definition at line 85 of file scsi_pass.c.

7.31.4.6 d_ioctl_t passioctl [static]

Definition at line 83 of file scsi_pass.c.

7.31.4.7 [periph_oninv_t](#) passoninvalidate [static]

Definition at line 87 of file scsi_pass.c.

Referenced by passasync().

7.31.4.8 d_open_t passopen [static]

Definition at line 81 of file scsi_pass.c.

7.31.4.9 [periph_ctor_t](#) passregister [static]

Definition at line 86 of file scsi_pass.c.

Referenced by passasync().

7.31.4.10 [periph_start_t](#) passstart [static]

Definition at line 89 of file scsi_pass.c.

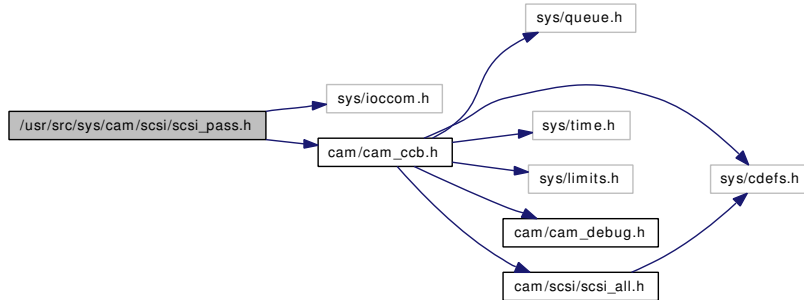
Referenced by passasync().

7.32 /usr/src/sys/cam/scsi/scsi_pass.h File Reference

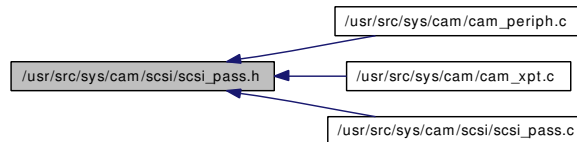
```
#include <sys/ioccom.h>
```

```
#include <cam/cam_ccb.h>
```

Include dependency graph for scsi_pass.h:



This graph shows which files directly or indirectly include this file:



Defines

- #define [_SCSI_PASS_H](#) 1
- #define [CAMIOCOMMAND_IOWR\(CAM_VERSION, 2, union ccb\)](#)
- #define [CAMGETPASSTHRU_IOWR\(CAM_VERSION, 3, union ccb\)](#)

7.32.1 Define Documentation

7.32.1.1 #define _SCSI_PASS_H 1

Definition at line 29 of file scsi_pass.h.

7.32.1.2 #define CAMGETPASSTHRU_IOWR(CAM_VERSION, 3, union ccb)

Definition at line 40 of file scsi_pass.h.

Referenced by `cam_periph_ioctl()`, and `xptioctl()`.

7.32.1.3 #define CAMIOCOMMAND_IOWR(CAM_VERSION, 2, union ccb)

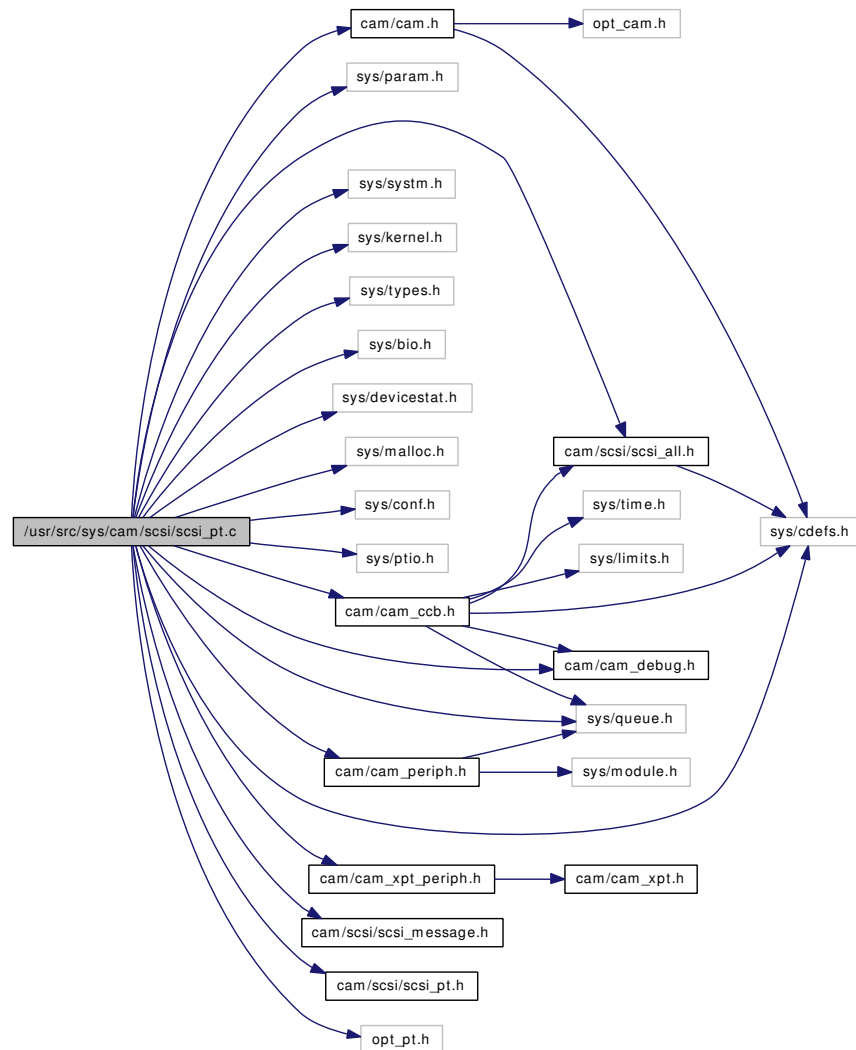
Definition at line 39 of file scsi_pass.h.

Referenced by `passioctl()`, and `xptioctl()`.

7.33 /usr/src/sys/cam/scsi/scsi_pt.c File Reference

```
#include <sys/cdefs.h>
#include <sys/param.h>
#include <sys/queue.h>
#include <sys/system.h>
#include <sys/kernel.h>
#include <sys/types.h>
#include <sys/bio.h>
#include <sys/devicestat.h>
#include <sys/malloc.h>
#include <sys/conf.h>
#include <sys/ptio.h>
#include <cam/cam.h>
#include <cam/cam_ccb.h>
#include <cam/cam_periph.h>
#include <cam/cam_xpt_periph.h>
#include <cam/cam_debug.h>
#include <cam/scsi/scsi_all.h>
#include <cam/scsi/scsi_message.h>
#include <cam/scsi/scsi_pt.h>
#include "opt_pt.h"
```

Include dependency graph for scsi_pt.c:



Data Structures

- struct [pt_softc](#)

Defines

- #define [ccb_state](#) ppriv_field0
- #define [ccb_bp](#) ppriv_ptr1
- #define [SCSI_PT_DEFAULT_TIMEOUT](#) 60

Enumerations

- enum [pt_state](#) { [PT_STATE_PROBE](#), [PT_STATE_NORMAL](#) }
- enum [pt_flags](#) { [PT_FLAG_NONE](#) = 0x00, [PT_FLAG_OPEN](#) = 0x01, [PT_FLAG_DEVICE_INVALID](#) = 0x02, [PT_FLAG_RETRY_UA](#) = 0x04 }

- enum `pt_ccb_state` { `PT_CCB_BUFFER_IO` = 0x01, `PT_CCB_WAITING` = 0x02, `PT_CCB_RETRY_UA` = 0x04, `PT_CCB_BUFFER_IO_UA` = `PT_CCB_BUFFER_IO|PT_CCB_RETRY_UA` }

Functions

- `__FBSDID` ("\$FreeBSD: src/sys/cam/scsi/scsi_pt.c,v 1.44 2006/12/05 07:45:28 mjacob Exp \$")
- static void `ptasync` (void *callback_arg, u_int32_t code, struct `cam_path` *path, void *arg)
- static void `ptdone` (struct `cam_periph` *periph, union `ccb` *done_ccb)
- static int `pterror` (union `ccb` *ccb, u_int32_t `cam_flags`, u_int32_t `sense_flags`)
- void `scsi_send_receive` (struct `ccb_scsiio` *csio, u_int32_t `retries`, void(*cbfcn)(struct `cam_periph` *, union `ccb` *), u_int tag_action, int readop, u_int byte2, u_int32_t `xfer_len`, u_int8_t *data_ptr, u_int8_t `sense_len`, u_int32_t `timeout`)
- `PERIPHDRIIVER_DECLARE` (pt, `ptdriver`)
- static int `ptopen` (struct `cdev` *dev, int flags, int fmt, struct thread *td)
- static int `ptclose` (struct `cdev` *dev, int flag, int fmt, struct thread *td)
- static void `ptstrategy` (struct `bio` *bp)
- static void `ptinit` (void)
- static `cam_status` `ptctor` (struct `cam_periph` *periph, void *arg)
- static void `ptoninvalidate` (struct `cam_periph` *periph)
- static void `ptdtor` (struct `cam_periph` *periph)
- static void `ptstart` (struct `cam_periph` *periph, union `ccb` *start_ccb)
- static int `ptioctl` (struct `cdev` *dev, u_long cmd, `caddr_t` addr, int flag, struct thread *td)

Variables

- static `d_open_t` `ptopen`
- static `d_close_t` `ptclose`
- static `d_strategy_t` `ptstrategy`
- static `periph_init_t` `ptinit`
- static `periph_ctor_t` `ptctor`
- static `periph_oninv_t` `ptoninvalidate`
- static `periph_dtor_t` `ptdtor`
- static `periph_start_t` `ptstart`
- static `d_ioctl_t` `ptioctl`
- static struct `periph_driver` `ptdriver`
- static struct `cdevsw` `pt_cdevsw`

7.33.1 Define Documentation

7.33.1.1 #define `ccb_bp ppriv_ptr1`

Definition at line 76 of file `scsi_pt.c`.

7.33.1.2 #define `ccb_state ppriv_field0`

Definition at line 75 of file `scsi_pt.c`.

7.33.1.3 #define SCSI_PT_DEFAULT_TIMEOUT 60

Definition at line 133 of file scsi_pt.c.

Referenced by ptctor().

7.33.2 Enumeration Type Documentation

7.33.2.1 enum [pt_ccb_state](#)

Enumerator:

PT_CCB_BUFFER_IO
PT_CCB_WAITING
PT_CCB_RETRY_UA
PT_CCB_BUFFER_IO_UA

Definition at line 67 of file scsi_pt.c.

7.33.2.2 enum [pt_flags](#)

Enumerator:

PT_FLAG_NONE
PT_FLAG_OPEN
PT_FLAG_DEVICE_INVALID
PT_FLAG_RETRY_UA

Definition at line 60 of file scsi_pt.c.

7.33.2.3 enum [pt_state](#)

Enumerator:

PT_STATE_PROBE
PT_STATE_NORMAL

Definition at line 55 of file scsi_pt.c.

7.33.3 Function Documentation

7.33.3.1 `__FBSDID` ("FreeBSD: src/sys/cam/scsi/scsi_pt. c, v 1.44 2006/12/05 07:45:28 mjacob Exp \$")

7.33.3.2 `PERIPHDRIVER_DECLARE` (pt, [ptdriver](#))

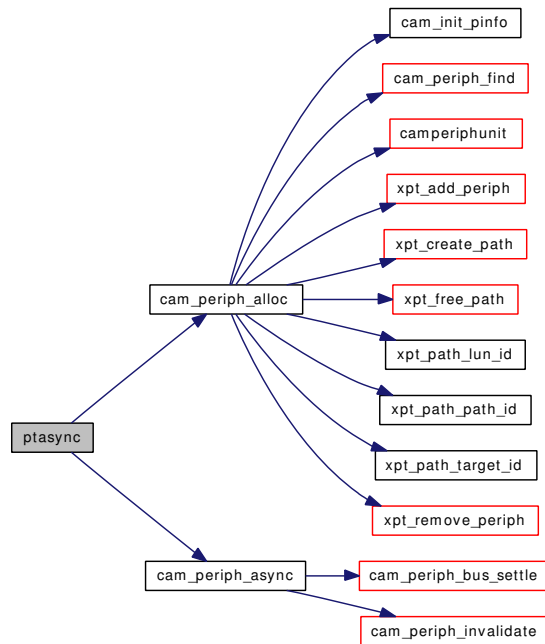
7.33.3.3 `static void ptasync` (void * *callback_arg*, u_int32_t *code*, struct [cam_path](#) * *path*, void * *arg*) [*static*]

Definition at line 409 of file scsi_pt.c.

References AC_BUS_RESET, AC_FOUND_DEVICE, AC_SENT_BDR, cam_periph_alloc(), cam_periph_async(), CAM_PERIPH_BIO, CAM_REQ_CMP, CAM_REQ_INPROG, ccb_getdev::ccb_h, ccb_getdev::inq_data, ccb_hdr::path, PT_CCB_RETRY_UA, PT_FLAG_RETRY_UA, ptctor, ptdtor, ptoninvalidate, ptstart, SID_TYPE, cam_periph::softc, and T_PROCESSOR.

Referenced by ptctor(), ptinit(), and ptoninvalidate().

Here is the call graph for this function:

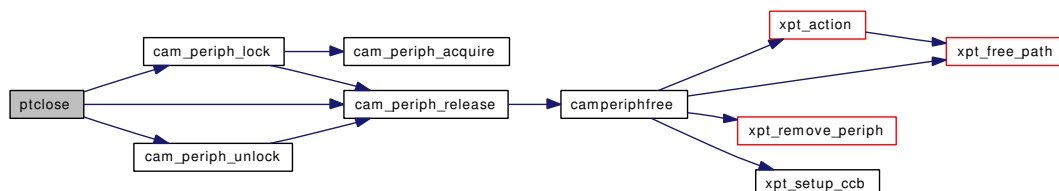


7.33.3.4 static int ptclose (struct cdev * dev, int flag, int fmt, struct thread * td) [static]

Definition at line 181 of file scsi_pt.c.

References cam_periph_lock(), cam_periph_release(), cam_periph_unlock(), PT_FLAG_OPEN, and cam_periph::softc.

Here is the call graph for this function:

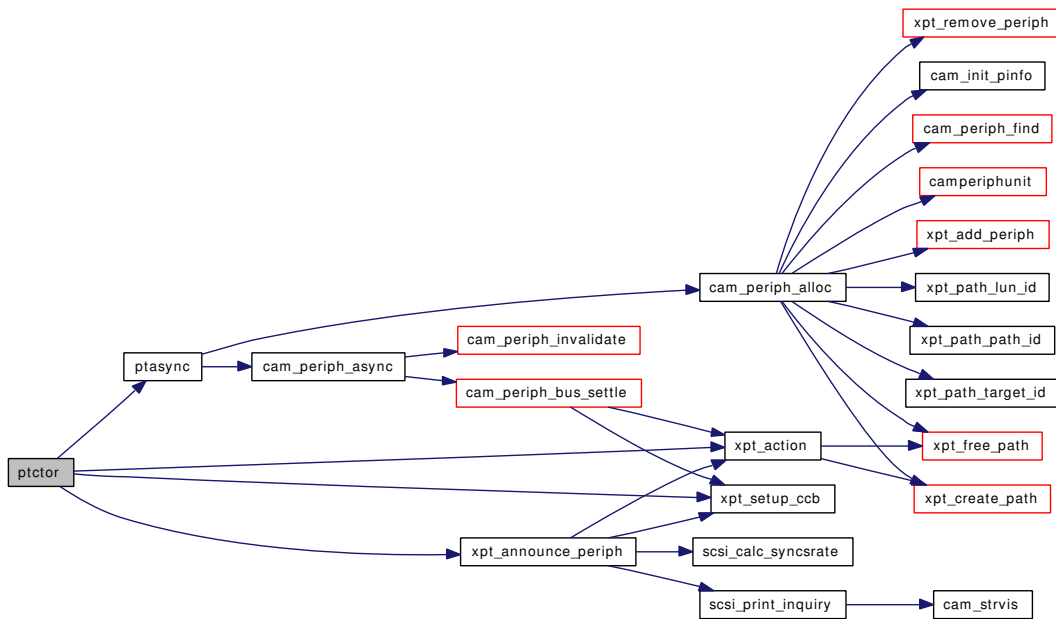


7.33.3.5 static cam_status ptctor (struct cam_periph * periph, void * arg) [static]

Definition at line 286 of file scsi_pt.c.

References AC_BUS_RESET, AC_LOST_DEVICE, AC_SENT_BDR, ccb_setasync::callback, ccb_setasync::callback_arg, CAM_REQ_CMP, CAM_REQ_CMP_ERR, ccb_setasync::ccb_h, ccb_setasync::event_enable, ccb_hdr::func_code, ccb_getdev::inq_data, cam_periph::path, cam_periph::periph_name, PT_STATE_NORMAL, ptasync(), SCSI_PT_DEFAULT_TIMEOUT, SID_TYPE, cam_periph::softc, cam_periph::unit_number, xpt_action(), xpt_announce_periph(), XPT_SASYNC_CB, and xpt_setup_ccb().

Here is the call graph for this function:



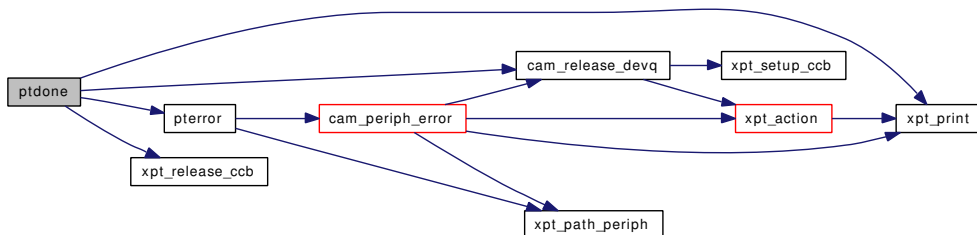
7.33.3.6 static void ptdone (struct [cam_periph](#) * *periph*, union [ccb](#) * *done_ccb*) [static]

Definition at line 537 of file `scsi_pt.c`.

References `pt_softc::bio_queue`, `CAM_DEV_QFRZN`, `cam_release_devq()`, `CAM_REQ_CMP`, `CAM_RETRY_SELTO`, `CAM_STATUS_MASK`, `ccb_hdr::cbfnp`, `ccb::ccb_h`, `ccb_scsiio::ccb_h`, `ccb::csio`, `pt_softc::device_stats`, `ccb_hdr::path`, `cam_periph::path`, `PT_CCB_BUFFER_IO`, `PT_CCB_BUFFER_IO_UA`, `PT_CCB_RETRY_UA`, `PT_CCB_WAITING`, `PT_FLAG_DEVICE_INVALID`, `pterror()`, `ccb_scsiio::resid`, `SF_RETRY_UA`, `cam_periph::softc`, `ccb_hdr::status`, `xpt_print()`, and `xpt_release_ccb()`.

Referenced by `ptstart()`.

Here is the call graph for this function:

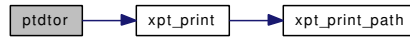


7.33.3.7 static void ptdtor (struct cam_periph * periph) [static]

Definition at line 394 of file scsi_pt.c.

References pt_softc::device_stats, cam_periph::path, cam_periph::softc, and xpt_print().

Here is the call graph for this function:

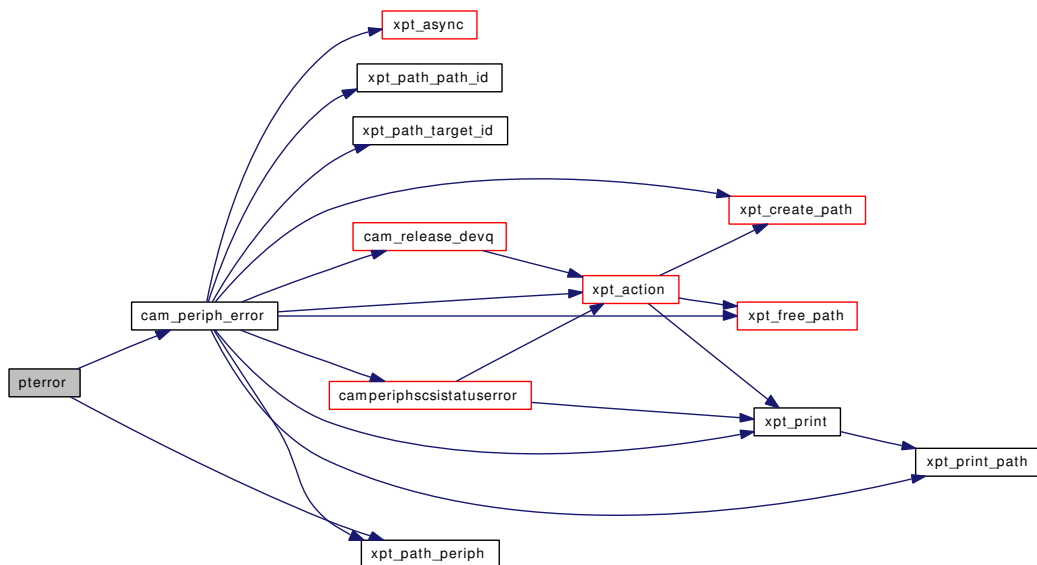
**7.33.3.8 static int pterror (union ccb * ccb, u_int32_t cam_flags, u_int32_t sense_flags) [static]**

Definition at line 633 of file scsi_pt.c.

References cam_periph_error(), ccb::ccb_h, ccb_hdr::path, cam_periph::softc, and xpt_path_periph().

Referenced by ptdone(), and ptioctl().

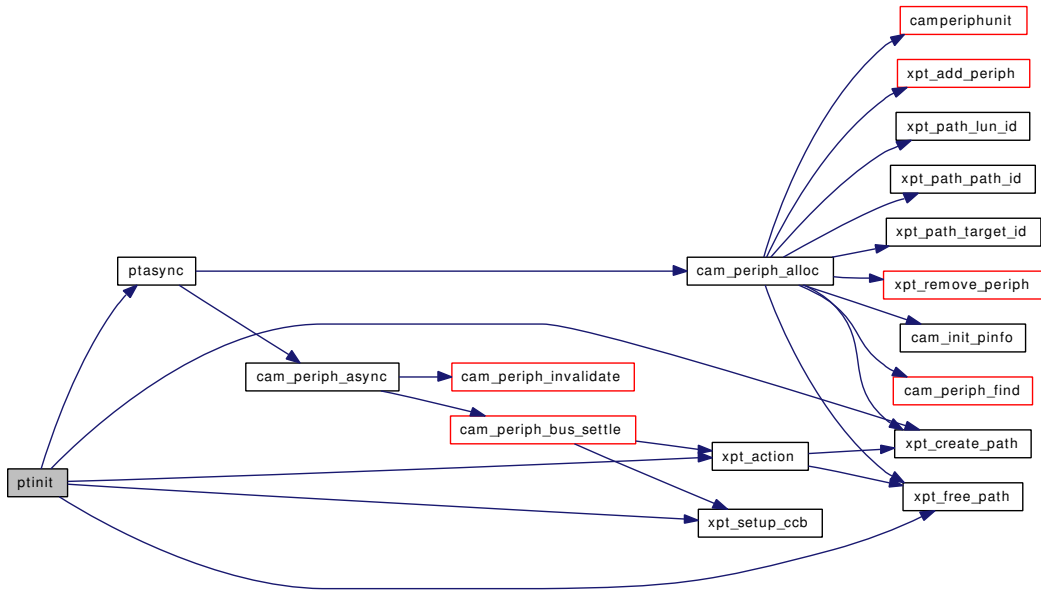
Here is the call graph for this function:

**7.33.3.9 static void ptinit (void) [static]**

Definition at line 254 of file scsi_pt.c.

References AC_FOUND_DEVICE, ccb_setasync::callback, ccb_setasync::callback_arg, CAM_LUN_WILDCARD, CAM_REQ_CMP, CAM_TARGET_WILDCARD, CAM_XPT_PATH_ID, ccb::ccb_h, ccb_setasync::ccb_h, ccb_setasync::event_enable, ccb_hdr::func_code, ptasync(), ccb_hdr::status, xpt_action(), xpt_create_path(), xpt_free_path(), XPT_SASYNC_CB, and xpt_setup_ccb().

Here is the call graph for this function:

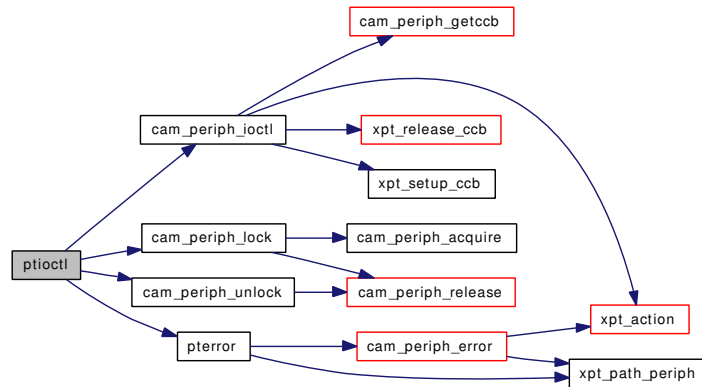


7.33.3.10 `static int ptioctl (struct cdev * dev, u_long cmd, caddr_t addr, int flag, struct thread * td)`
`[static]`

Definition at line 646 of file scsi_pt.c.

References `cam_periph_ioctl()`, `cam_periph_lock()`, `cam_periph_unlock()`, `pterror()`, and `cam_periph::softc`.

Here is the call graph for this function:



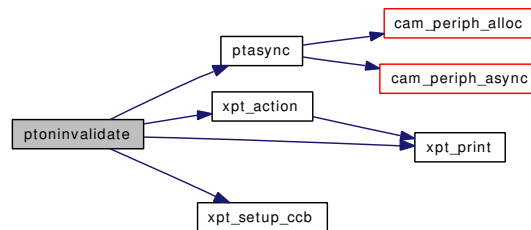
7.33.3.11 `static void ptoninvalidate (struct cam_periph * periph)` `[static]`

Definition at line 353 of file scsi_pt.c.

References `ccb_setasync::callback`, `ccb_setasync::callback_arg`, `ccb_setasync::ccb_h`, `ccb_setasync::event_enable`, `ccb_hdr::func_code`, `cam_periph::path`, `PT_FLAG_DEVICE_INVALID`,

ptasync(), cam_periph::softc, xpt_action(), xpt_print(), XPT_SASYNC_CB, and xpt_setup_ccb().

Here is the call graph for this function:

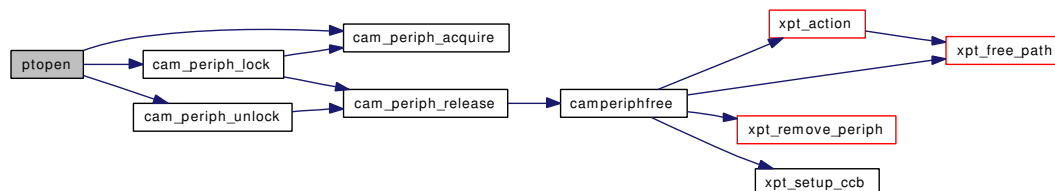


7.33.3.12 static int ptopen (struct cdev * dev, int flags, int fmt, struct thread * td) [static]

Definition at line 137 of file scsi_pt.c.

References CAM_DEBUG, CAM_DEBUG_TRACE, cam_periph_acquire(), cam_periph_lock(), cam_periph_unlock(), CAM_REQ_CMP, cam_periph::path, PT_FLAG_DEVICE_INVALID, PT_FLAG_OPEN, and cam_periph::softc.

Here is the call graph for this function:

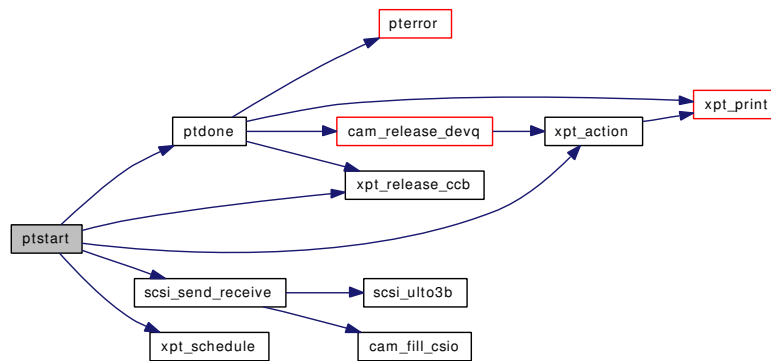


7.33.3.13 static void ptstart (struct cam_periph * periph, union ccb * start_ccb) [static]

Definition at line 469 of file scsi_pt.c.

References pt_softc::bio_queue, CAM_DEBUG_PRINT, CAM_DEBUG_SUBTRACE, CAM_PRIORITY_NONE, ccb::ccb_h, ccb::csio, pt_softc::device_stats, cam_periph::immediate_priority, MSG_SIMPLE_Q_TAG, cam_periph::pinfo, cam_pinfo::priority, PT_CCB_BUFFER_IO_UA, PT_CCB_WAITING, ptdone(), scsi_send_receive(), cam_periph::softc, SSD_FULL_SIZE, xpt_action(), xpt_release_ccb(), and xpt_schedule().

Here is the call graph for this function:

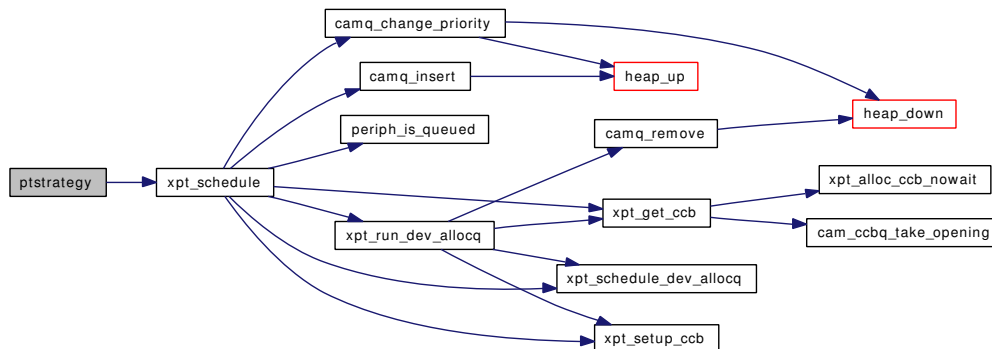


7.33.3.14 static void ptstrategy (struct bio * bp) [static]

Definition at line 208 of file scsi_pt.c.

References pt_softc::bio_queue, PT_FLAG_DEVICE_INVALID, cam_periph::softc, and xpt_schedule().

Here is the call graph for this function:



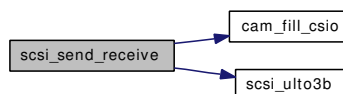
7.33.3.15 void scsi_send_receive (struct ccb_scsiio * csio, u_int32_t retries, void(*) (struct cam_periph *, union ccb *) cbfn, u_int tag_action, int readop, u_int byte2, u_int32_t xfer_len, u_int8_t * data_ptr, u_int8_t sense_len, u_int32_t timeout)

Definition at line 695 of file scsi_pt.c.

References scsi_send_receive::byte2, CAM_DIR_IN, CAM_DIR_OUT, cam_fill_csio(), cdb_t::cdb_bytes, ccb_scsiio::cdb_io, scsi_send_receive::control, scsi_send_receive::opcode, RECEIVE, scsi_ulto3b(), SEND, and scsi_send_receive::xfer_len.

Referenced by ptstart().

Here is the call graph for this function:



7.33.4 Variable Documentation

7.33.4.1 struct cdevsw [pt_cdevsw](#) [static]

Initial value:

```
{
    .d_version =    D_VERSION,
    .d_flags =     D_NEEDGIANT,
    .d_open =      ptopen,
    .d_close =     ptclose,
    .d_read =      physread,
    .d_write =     physwrite,
    .d_ioctl =     ptioctl,
    .d_strategy =  ptstrategy,
    .d_name =      "pt",
}
```

Definition at line 120 of file scsi_pt.c.

7.33.4.2 d_close_t [ptclose](#) [static]

Definition at line 90 of file scsi_pt.c.

7.33.4.3 [periph_ctor_t](#) [ptctor](#) [static]

Definition at line 95 of file scsi_pt.c.

Referenced by ptasync().

7.33.4.4 struct [periph_driver](#) [ptdriver](#) [static]

Initial value:

```
{
    ptinit, "pt",
    TAILQ_HEAD_INITIALIZER(ptdriver.units), 0
}
```

Definition at line 111 of file scsi_pt.c.

7.33.4.5 [periph_dtor_t](#) [ptdtor](#) [static]

Definition at line 97 of file scsi_pt.c.

Referenced by ptasync().

7.33.4.6 [periph_init_t](#) [ptinit](#) [static]

Definition at line 92 of file scsi_pt.c.

7.33.4.7 d_ioctl_t [ptioctl](#) [static]

Definition at line 101 of file scsi_pt.c.

7.33.4.8 `periph_oninv_t ptoninvalidate` [static]

Definition at line 96 of file `scsi_pt.c`.

Referenced by `ptasync()`.

7.33.4.9 `d_open_t ptopen` [static]

Definition at line 89 of file `scsi_pt.c`.

7.33.4.10 `periph_start_t ptstart` [static]

Definition at line 98 of file `scsi_pt.c`.

Referenced by `ptasync()`.

7.33.4.11 `d_strategy_t ptstrategy` [static]

Definition at line 91 of file `scsi_pt.c`.

7.34 /usr/src/sys/cam/scsi/scsi_pt.h File Reference

This graph shows which files directly or indirectly include this file:



Data Structures

- struct [scsi_send_receive](#)

Defines

- #define [_SCSI_SCSI_PT_H](#) 1
- #define [RECEIVE](#) 0x08
- #define [SEND](#) 0x0A

7.34.1 Define Documentation

7.34.1.1 #define _SCSI_SCSI_PT_H 1

Definition at line 32 of file scsi_pt.h.

7.34.1.2 #define RECEIVE 0x08

Definition at line 45 of file scsi_pt.h.

Referenced by [scsi_send_receive\(\)](#).

7.34.1.3 #define SEND 0x0A

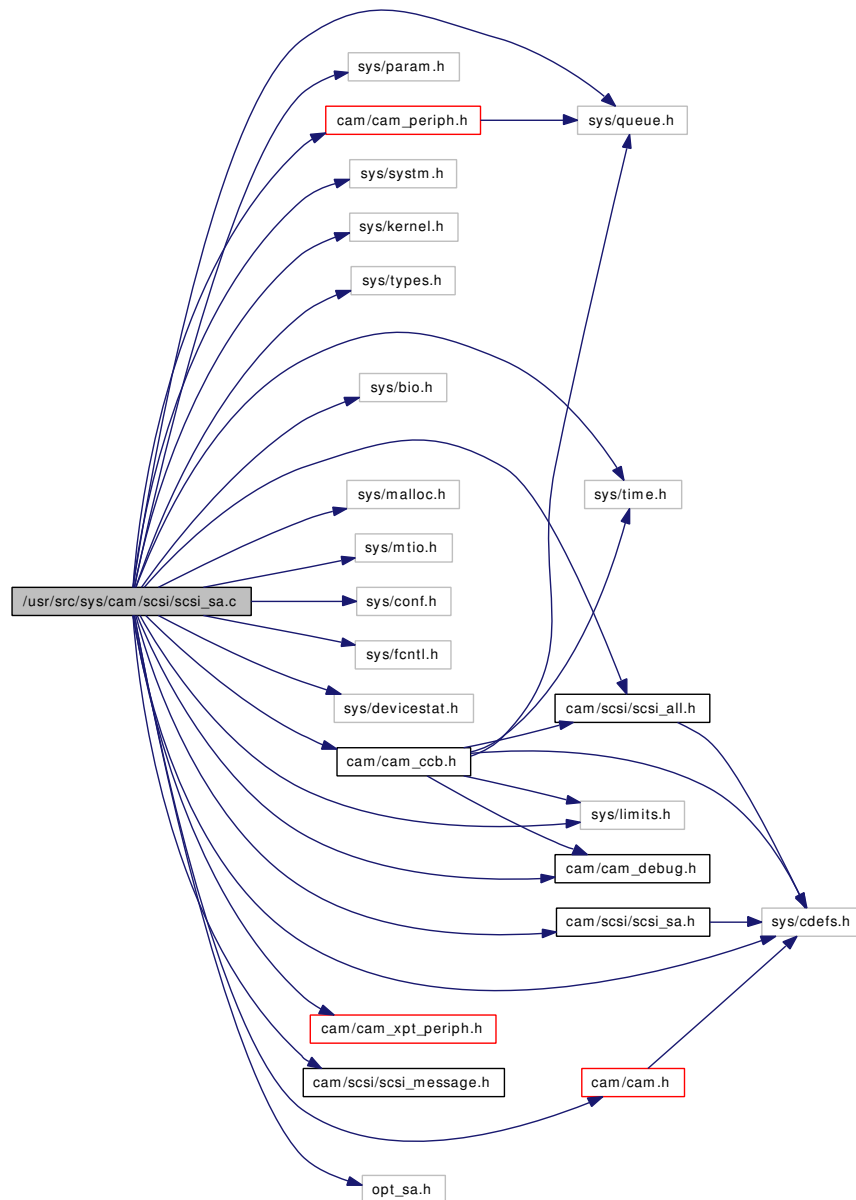
Definition at line 46 of file scsi_pt.h.

Referenced by [scsi_send_receive\(\)](#).

7.35 /usr/src/sys/cam/scsi/scsi_sa.c File Reference

```
#include <sys/cdefs.h>
#include <sys/param.h>
#include <sys/queue.h>
#include <sys/system.h>
#include <sys/kernel.h>
#include <sys/types.h>
#include <sys/time.h>
#include <sys/bio.h>
#include <sys/limits.h>
#include <sys/malloc.h>
#include <sys/mtio.h>
#include <sys/conf.h>
#include <sys/fcntl.h>
#include <sys/devicestat.h>
#include <cam/cam.h>
#include <cam/cam_ccb.h>
#include <cam/cam_periph.h>
#include <cam/cam_xpt_periph.h>
#include <cam/cam_debug.h>
#include <cam/scsi/scsi_all.h>
#include <cam/scsi/scsi_message.h>
#include <cam/scsi/scsi_sa.h>
#include <opt_sa.h>
```

Include dependency graph for scsi_sa.c:



Data Structures

- struct [sa_devs](#)
- struct [sa_devs::sa_mode_devs](#)
- struct [sa_softc](#)
- struct [sa_quirk_entry](#)

Defines

- #define [SA_IO_TIMEOUT](#) 4
- #define [SA_SPACE_TIMEOUT](#) 1 * 60

- #define SA_REWIND_TIMEOUT 2 * 60
- #define SA_ERASE_TIMEOUT 4 * 60
- #define SCSIOP_TIMEOUT (60 * 1000)
- #define IO_TIMEOUT (SA_IO_TIMEOUT * 60 * 1000)
- #define REWIND_TIMEOUT (SA_REWIND_TIMEOUT * 60 * 1000)
- #define ERASE_TIMEOUT (SA_ERASE_TIMEOUT * 60 * 1000)
- #define SPACE_TIMEOUT (SA_SPACE_TIMEOUT * 60 * 1000)
- #define UNUSED_PARAMETER(x) x = x
- #define QFRLS(ccb)
- #define ccb_pflags ppriv_field0
- #define ccb_bp ppriv_ptr1
- #define SA_CCB_BUFFER_IO 0x0
- #define SA_CCB_WAITING 0x1
- #define SA_CCB_TPEMASK 0x1
- #define SA_POSITION_UPDATED 0x2
- #define Set_CCB_Type(x, type)
- #define CCB_Type(x) (x → ccb_h.ccb_pflags & SA_CCB_TPEMASK)
- #define SAUNIT(DEV) (((minor(DEV) & 0xF0) >> 4) | ((minor(DEV) & 0x3f0000) >> 16))
- #define SAMODE(z) ((minor(z) & 0x3))
- #define SADENSITY(z) (((minor(z) >> 2) & 0x3))
- #define SA_IS_CTRL(z) (minor(z) & (1 << 29))
- #define SA_NOT_CTLDEV 0
- #define SA_CTLDEV 1
- #define SA_ATYPE_R 0
- #define SA_ATYPE_NR 1
- #define SA_ATYPE_ER 2
- #define SAMINOR(ctl, unit, mode, access)
- #define SA_NUM_MODES 4
- #define last_io_sense errinfo._last_io_sense
- #define last_io_resid errinfo._last_io_resid
- #define last_io_cdb errinfo._last_io_cdb
- #define last_ctl_sense errinfo._last_ctl_sense
- #define last_ctl_resid errinfo._last_ctl_resid
- #define last_ctl_cdb errinfo._last_ctl_cdb
- #define D_TAPE 0
- #define PENDING_MOUNT_CHECK(softc, periph, dev)

Enumerations

- enum sa_state { SA_STATE_NORMAL, SA_STATE_ABNORMAL }
- enum sa_flags {
 - SA_FLAG_OPEN = 0x0001, SA_FLAG_FIXED = 0x0002, SA_FLAG_TAPE_LOCKED = 0x0004, SA_FLAG_TAPE_MOUNTED = 0x0008,
 - SA_FLAG_TAPE_WP = 0x0010, SA_FLAG_TAPE_WRITTEN = 0x0020, SA_FLAG_EOM_PENDING = 0x0040, SA_FLAG_EIO_PENDING = 0x0080,
 - SA_FLAG_EOF_PENDING = 0x0100, SA_FLAG_ERR_PENDING, SA_FLAG_INVALID = 0x0200, SA_FLAG_COMP_ENABLED = 0x0400,
 - SA_FLAG_COMP_SUPP = 0x0800, SA_FLAG_COMP_UNSUPP = 0x1000, SA_FLAG_TAPE_FROZEN = 0x2000 }

- enum `sa_mode` { `SA_MODE_REWIND` = 0x00, `SA_MODE_NOREWIND` = 0x01, `SA_MODE_OFFLINE` = 0x02 }
- enum `sa_params` {
`SA_PARAM_NONE` = 0x00, `SA_PARAM_BLOCKSIZE` = 0x01, `SA_PARAM_DENSITY` = 0x02,
`SA_PARAM_COMPRESSION` = 0x04,
`SA_PARAM_BUFF_MODE` = 0x08, `SA_PARAM_NUMBLOCKS` = 0x10, `SA_PARAM_WP` =
0x20, `SA_PARAM_SPEED` = 0x40,
`SA_PARAM_ALL` = 0x7f }
- enum `sa_quirks` {
`SA_QUIRK_NONE` = 0x00, `SA_QUIRK_NOCOMP` = 0x01, `SA_QUIRK_FIXED` = 0x02, `SA_QUIRK_VARIABLE` = 0x04,
`SA_QUIRK_2FM` = 0x08, `SA_QUIRK_1FM` = 0x10, `SA_QUIRK_NODREAD` = 0x20, `SA_QUIRK_NO_MODESEL` = 0x40,
`SA_QUIRK_NO_CPAGE` = 0x80 }

Functions

- `__FBSDDID` ("FreeBSD: src/sys/cam/scsi/scsi_sa.c,v 1.107 2006/12/05 07:45:28 mjacob Exp \$")
- `MALLOC_DEFINE` (M_SCSISA, "SCSI sa", "SCSI sequential access buffers")
- static void `saasync` (void *callback_arg, u_int32_t code, struct `cam_path` *path, void *arg)
- static void `sadone` (struct `cam_periph` *periph, union `ccb` *start_ccb)
- static int `saerror` (union `ccb` *ccb, u_int32_t `cam_flags`, u_int32_t `sense_flags`)
- static int `samarkswanted` (struct `cam_periph` *)
- static int `sacheckeed` (struct `cam_periph` *periph)
- static int `sagetparams` (struct `cam_periph` *periph, `sa_params` `params_to_get`, u_int32_t *blocksize, u_int8_t *density, u_int32_t *numblocks, int *buff_mode, u_int8_t *write_protect, u_int8_t *speed, int *comp_supported, int *comp_enabled, u_int32_t *comp_algorithm, `sa_comp_t` *comp_page)
- static int `sasetparams` (struct `cam_periph` *periph, `sa_params` `params_to_set`, u_int32_t blocksize, u_int8_t density, u_int32_t comp_algorithm, u_int32_t `sense_flags`)
- static void `saprevent` (struct `cam_periph` *periph, int action)
- static int `sarewind` (struct `cam_periph` *periph)
- static int `saspace` (struct `cam_periph` *periph, int count, `scsi_space_code` code)
- static int `samount` (struct `cam_periph` *, int, struct `cdev` *)
- static int `saretension` (struct `cam_periph` *periph)
- static int `sareservereleaseunit` (struct `cam_periph` *periph, int reserve)
- static int `saloadunload` (struct `cam_periph` *periph, int load)
- static int `saerase` (struct `cam_periph` *periph, int longerase)
- static int `sawritefilemarks` (struct `cam_periph` *periph, int nmarks, int setmarks)
- static int `sardpos` (struct `cam_periph` *periph, int, u_int32_t *)
- static int `sasetpos` (struct `cam_periph` *periph, int, u_int32_t *)
- `PERIPHDRIVER_DECLARE` (sa, `sadriver`)
- static int `saopen` (struct `cdev` *dev, int flags, int fmt, struct `thread` *td)
- static int `saclose` (struct `cdev` *dev, int flag, int fmt, struct `thread` *td)
- static void `sastrategy` (struct `bio` *bp)
- static int `saioclt` (struct `cdev` *dev, u_long cmd, `caddr_t` arg, int flag, struct `thread` *td)
- static void `sainit` (void)
- static void `saoninvalidate` (struct `cam_periph` *periph)
- static void `sacleanup` (struct `cam_periph` *periph)
- static `cam_status` `saregister` (struct `cam_periph` *periph, void *arg)

- static void `sastart` (struct `cam_periph` *periph, union `ccb` *start_ccb)
- void `scsi_read_block_limits` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcn)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, struct `scsi_read_block_limits_data` *rlimit_buf, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_sa_read_write` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcn)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, int readop, int sli, int fixed, u_int32_t length, u_int8_t *data_ptr, u_int32_t dxfer_len, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_load_unload` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcn)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, int immediate, int eot, int reten, int load, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_rewind` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcn)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, int immediate, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_space` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcn)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, `scsi_space_code` code, u_int32_t count, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_write_filemarks` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcn)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, int immediate, int setmark, u_int32_t num_marks, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_reserve_release_unit` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcn)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, int third_party, int third_party_id, u_int8_t sense_len, u_int32_t timeout, int reserve)
- void `scsi_erase` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcn)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, int immediate, int long_erase, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_read_position` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcn)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, int hardsoft, struct `scsi_tape_position_data` *sbp, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_set_position` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcn)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, int hardsoft, u_int32_t blkno, u_int8_t sense_len, u_int32_t timeout)

Variables

- static struct `sa_quirk_entry` `sa_quirk_table` []
- static d_open_t `saopen`
- static d_close_t `saclose`
- static d_strategy_t `sastrategy`
- static d_ioctl_t `saioclt`
- static `periph_init_t` `sainit`
- static `periph_ctor_t` `saregister`
- static `periph_oninv_t` `saoninvalidate`
- static `periph_dtor_t` `sacleanup`
- static `periph_start_t` `sastart`
- static struct `periph_driver` `sadriver`
- static struct `cdevsw` `sa_cdevsw`

7.35.1 Define Documentation

7.35.1.1 #define `ccb_bp ppriv_ptr1`

Definition at line 112 of file `scsi_sa.c`.

7.35.1.2 #define ccb_pflags ppriv_field0

Definition at line 111 of file scsi_sa.c.

7.35.1.3 #define CCB_Type(x) (x → ccb_h.ccb_pflags & SA_CCB_TPEMASK)

Definition at line 123 of file scsi_sa.c.

Referenced by sadone(), and saerror().

7.35.1.4 #define D_TAPE 0

Definition at line 427 of file scsi_sa.c.

7.35.1.5 #define ERASE_TIMEOUT (SA_ERASE_TIMEOUT * 60 * 1000)

Definition at line 86 of file scsi_sa.c.

Referenced by saerase(), and saretension().

7.35.1.6 #define IO_TIMEOUT (SA_IO_TIMEOUT * 60 * 1000)

Definition at line 84 of file scsi_sa.c.

Referenced by samount(), sastart(), and sawritefilemarks().

7.35.1.7 #define last_ctl_cdb errinfo._last_ctl_cdb

Definition at line 253 of file scsi_sa.c.

7.35.1.8 #define last_ctl_resid errinfo._last_ctl_resid

Definition at line 252 of file scsi_sa.c.

7.35.1.9 #define last_ctl_sense errinfo._last_ctl_sense

Definition at line 251 of file scsi_sa.c.

7.35.1.10 #define last_io_cdb errinfo._last_io_cdb

Definition at line 250 of file scsi_sa.c.

7.35.1.11 #define last_io_resid errinfo._last_io_resid

Definition at line 249 of file scsi_sa.c.

7.35.1.12 #define last_io_sense errinfo._last_io_sense

Definition at line 248 of file scsi_sa.c.

7.35.1.13 #define PENDING_MOUNT_CHECK(softc, periph, dev)**Value:**

```

if (softc->open_pending_mount) {
    error = samount(periph, 0, dev);
    if (error) {
        break;
    }
    saprevent(periph, PR_PREVENT);
    softc->open_pending_mount = 0;
}

```

Definition at line 805 of file scsi_sa.c.

Referenced by saioctl().

7.35.1.14 #define QFRLS(ccb)**Value:**

```

if (((ccb)->ccb_h.status & CAM_DEV_QFRZN) != 0) \
    cam_release_devq((ccb)->ccb_h.path, 0, 0, 0, FALSE)

```

Definition at line 97 of file scsi_sa.c.

Referenced by saerror(), sagetparams(), saloadunload(), samount(), saprevent(), sareservereleaseunit(), and sasetparams().

7.35.1.15 #define REWIND_TIMEOUT (SA_REWIND_TIMEOUT * 60 * 1000)

Definition at line 85 of file scsi_sa.c.

Referenced by saloadunload(), samount(), and sarewind().

7.35.1.16 #define SA_ATYPE_ER 2

Definition at line 189 of file scsi_sa.c.

Referenced by saregister().

7.35.1.17 #define SA_ATYPE_NR 1

Definition at line 188 of file scsi_sa.c.

Referenced by saregister().

7.35.1.18 #define SA_ATYPE_R 0

Definition at line 187 of file scsi_sa.c.

Referenced by saregister().

7.35.1.19 #define SA_CCB_BUFFER_IO 0x0

Definition at line 114 of file scsi_sa.c.

Referenced by sadone(), saerror(), and sastart().

7.35.1.20 #define SA_CCB_TPEMASK 0x1

Definition at line 116 of file scsi_sa.c.

7.35.1.21 #define SA_CCB_WAITING 0x1

Definition at line 115 of file scsi_sa.c.

Referenced by sadone(), saerror(), and sastart().

7.35.1.22 #define SA_CTLDEV 1

Definition at line 185 of file scsi_sa.c.

Referenced by saregister().

7.35.1.23 #define SA_ERASE_TIMEOUT 4 * 60

Definition at line 79 of file scsi_sa.c.

7.35.1.24 #define SA_IO_TIMEOUT 4

Definition at line 70 of file scsi_sa.c.

7.35.1.25 #define SA_IS_CTRL(z) (minor(z) & (1 << 29))

Definition at line 182 of file scsi_sa.c.

Referenced by saclose(), saioctl(), saopen(), and sastrategy().

7.35.1.26 #define SA_NOT_CTLDEV 0

Definition at line 184 of file scsi_sa.c.

Referenced by saregister().

7.35.1.27 #define SA_NUM_MODES 4

Definition at line 195 of file scsi_sa.c.

Referenced by sacleanup(), and saregister().

7.35.1.28 #define SA_POSITION_UPDATED 0x2

Definition at line 117 of file scsi_sa.c.

Referenced by sadone(), saerror(), and sastart().

7.35.1.29 #define SA_REWIND_TIMEOUT 2 * 60

Definition at line 76 of file scsi_sa.c.

7.35.1.30 #define SA_SPACE_TIMEOUT 1 * 60

Definition at line 73 of file scsi_sa.c.

7.35.1.31 #define SADENSITY(z) (((minor(z) >> 2) & 0x3))

Definition at line 181 of file scsi_sa.c.

7.35.1.32 #define SAMINOR(ctl, unit, mode, access)**Value:**

```
((ctl << 29) | ((unit & 0x3f0) << 16) | ((unit & 0xf) << 4) | \
 (mode << 0x2) | (access & 0x3))
```

Definition at line 191 of file scsi_sa.c.

Referenced by saregister().

7.35.1.33 #define SAMODE(z) ((minor(z) & 0x3)

Definition at line 180 of file scsi_sa.c.

Referenced by saclose(), and saioctl().

7.35.1.34 #define SAUNIT(DEV) (((minor(DEV) & 0xF0) >> 4) | ((minor(DEV) & 0x3f0000) >> 16))

Definition at line 177 of file scsi_sa.c.

Referenced by saclose(), and saopen().

7.35.1.35 #define SCSIOP_TIMEOUT (60 * 1000)

Definition at line 82 of file scsi_sa.c.

Referenced by sasetparams(), samount(), saprevent(), sardpos(), sareserveunit(), and sasetparams().

7.35.1.36 #define Set_CCB_Type(x, type)**Value:**

```
x->ccb_h.ccb_pflags &= ~SA_CCB_TYEMASK;      \
x->ccb_h.ccb_pflags |= type
```

Definition at line 119 of file scsi_sa.c.

Referenced by sastart().

7.35.1.37 #define SPACE_TIMEOUT (SA_SPACE_TIMEOUT * 60 * 1000)

Definition at line 87 of file scsi_sa.c.

Referenced by sasetpos(), and saspace().

7.35.1.38 #define UNUSED_PARAMETER(x) x = x

Definition at line 94 of file scsi_sa.c.

Referenced by samount().

7.35.2 Enumeration Type Documentation**7.35.2.1 enum [sa_flags](#)****Enumerator:**

```
SA_FLAG_OPEN
SA_FLAG_FIXED
SA_FLAG_TAPE_LOCKED
SA_FLAG_TAPE_MOUNTED
SA_FLAG_TAPE_WP
SA_FLAG_TAPE_WRITTEN
SA_FLAG_EOM_PENDING
SA_FLAG_EIO_PENDING
SA_FLAG_EOF_PENDING
SA_FLAG_ERR_PENDING
SA_FLAG_INVALID
SA_FLAG_COMP_ENABLED
SA_FLAG_COMP_SUPP
SA_FLAG_COMP_UNSUPP
SA_FLAG_TAPE_FROZEN
```

Definition at line 127 of file scsi_sa.c.

7.35.2.2 enum [sa_mode](#)

Enumerator:

SA_MODE_REWIND
SA_MODE_NOREWIND
SA_MODE_OFFLINE

Definition at line 146 of file scsi_sa.c.

7.35.2.3 enum [sa_params](#)

Enumerator:

SA_PARAM_NONE
SA_PARAM_BLOCKSIZE
SA_PARAM_DENSITY
SA_PARAM_COMPRESSION
SA_PARAM_BUFF_MODE
SA_PARAM_NUMBLOCKS
SA_PARAM_WP
SA_PARAM_SPEED
SA_PARAM_ALL

Definition at line 152 of file scsi_sa.c.

7.35.2.4 enum [sa_quirks](#)

Enumerator:

SA_QUIRK_NONE
SA_QUIRK_NOCOMP
SA_QUIRK_FIXED
SA_QUIRK_VARIABLE
SA_QUIRK_2FM
SA_QUIRK_1FM
SA_QUIRK_NODREAD
SA_QUIRK_NO_MODESEL
SA_QUIRK_NO_CPAGE

Definition at line 164 of file scsi_sa.c.

7.35.2.5 enum [sa_state](#)

Enumerator:

SA_STATE_NORMAL
SA_STATE_ABNORMAL

Definition at line 107 of file scsi_sa.c.

7.35.3 Function Documentation

7.35.3.1 `__FBSDID` ("\$FreeBSD: src/sys/cam/scsi/scsi_sa.c, v 1.107 2006/12/05 07:45:28 mjacob Exp \$")

7.35.3.2 `MALLOC_DEFINE` (M_SCISISA, "SCSI sa", "SCSI sequential access buffers")

7.35.3.3 `PERIPHDRIVER_DECLARE` (sa, [sdriver](#))

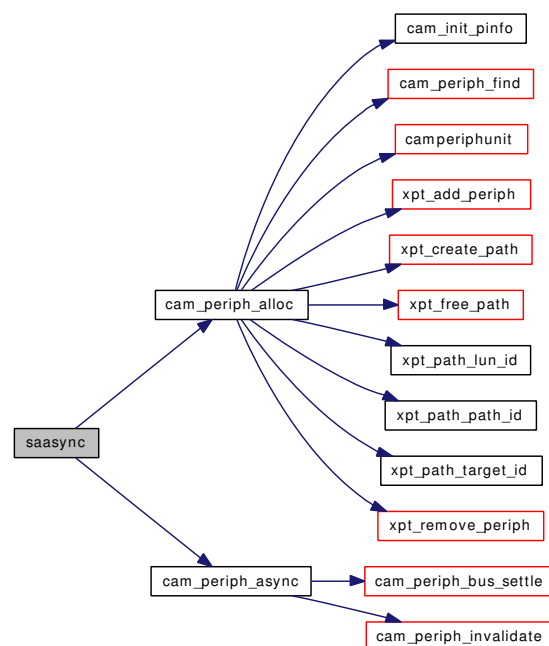
7.35.3.4 `static void saasync` (void * *callback_arg*, u_int32_t *code*, struct [cam_path](#) * *path*, void * *arg*) [static]

Definition at line 1434 of file `scsi_sa.c`.

References `AC_FOUND_DEVICE`, `cam_periph_alloc()`, `cam_periph_async()`, `CAM_PERIPH_BIO`, `CAM_REQ_CMP`, `CAM_REQ_INPROG`, `ccb_getdev::ccb_h`, `ccb_getdev::inq_data`, `ccb_hdr::path`, `SID_TYPE`, and `T_SEQUENTIAL`.

Referenced by `sainit()`, `saoninvalidate()`, and `saregister()`.

Here is the call graph for this function:



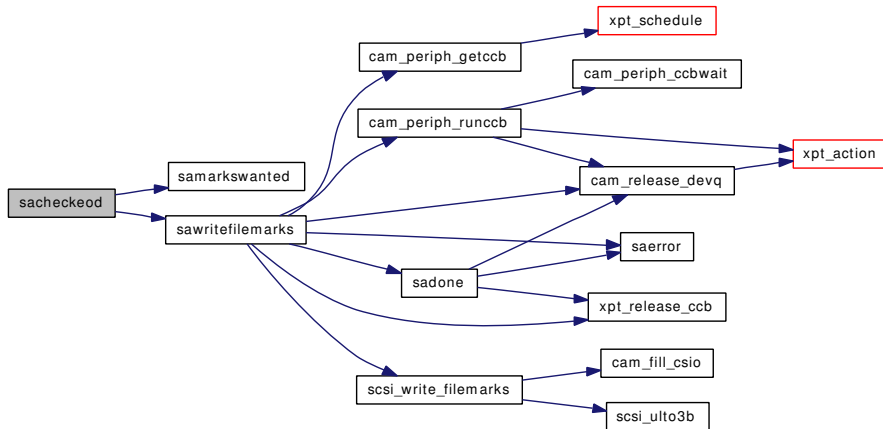
7.35.3.5 `static int sachekeod` (struct [cam_periph](#) * *periph*) [static]

Definition at line 2355 of file `scsi_sa.c`.

References `samarkswanted()`, and `sawritefilemarks()`.

Referenced by `saclose()`, and `saiocntl()`.

Here is the call graph for this function:

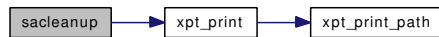


7.35.3.6 `static void sacleanup (struct cam_periph * periph) [static]`

Definition at line 1412 of file scsi_sa.c.

References sa_devs::ctl_dev, sa_softc::device_stats, sa_softc::devs, sa_devs::sa_mode_devs::er_dev, sa_devs::mode_devs, sa_devs::sa_mode_devs::nr_dev, cam_periph::path, sa_devs::sa_mode_devs::r_dev, SA_NUM_MODES, cam_periph::softc, and xpt_print().

Here is the call graph for this function:

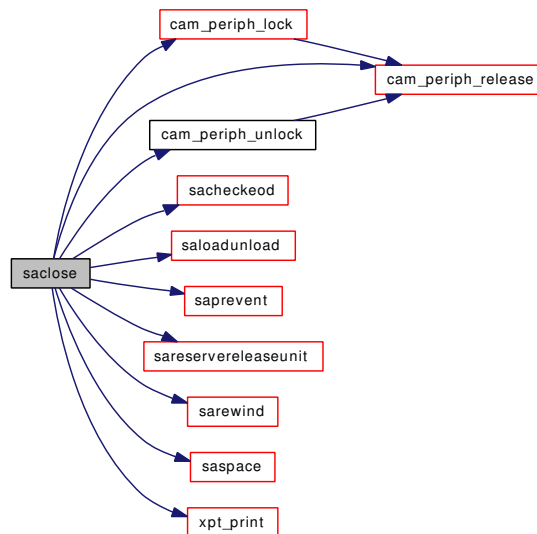


7.35.3.7 `static int saclose (struct cdev * dev, int flag, int fmt, struct thread * td) [static]`

Definition at line 518 of file scsi_sa.c.

References CAM_DEBUG, CAM_DEBUG_INFO, CAM_DEBUG_TRACE, cam_periph_lock(), cam_periph_release(), cam_periph_unlock(), sa_softc::ctrl_mode, sa_softc::filemarks, sa_softc::flags, sa_softc::open_pending_mount, sa_softc::open_ronly, cam_periph::path, PR_ALLOW, sa_softc::quirks, SA_FLAG_OPEN, SA_FLAG_TAPE_FROZEN, SA_FLAG_TAPE_MOUNTED, SA_FLAG_TAPE_UNWRITTEN, SA_IS_CTRL, SA_MODE_NOREWIND, SA_MODE_OFFLINE, SA_MODE_REWIND, SA_QUIRK_2FM, sachecked(), saloadunload(), SAMODE, saprevent(), sareservereleaseunit(), sarewind(), saspace(), SAUNIT, cam_periph::softc, SS_FILEMARKS, and xpt_print().

Here is the call graph for this function:



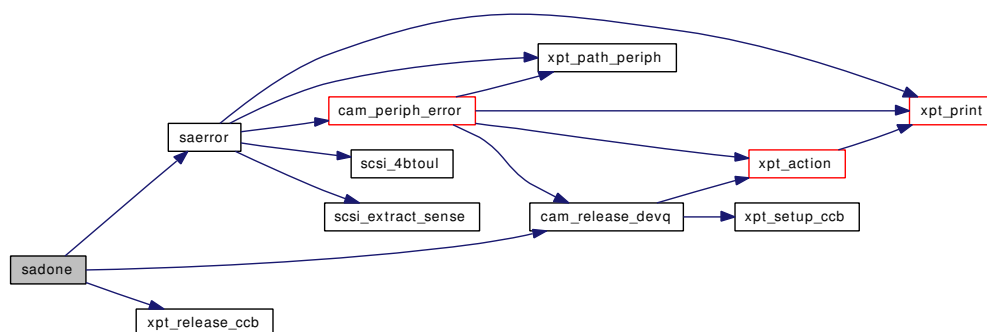
7.35.3.8 static void sadone (struct [cam_periph](#) * *periph*, union [ccb](#) * *start_ccb*) [static]

Definition at line 1762 of file `scsi_sa.c`.

References `sa_softc::bio_queue`, `sa_softc::blk_shift`, `sa_softc::blkno`, `CAM_DEBUG`, `CAM_DEBUG_INFO`, `cam_release_devq()`, `CAM_REQ_CMP`, `CAM_STATUS_MASK`, `ccb_hdr::cbfcnp`, `ccb_scsiio::ccb_h`, `ccb::ccb_h`, `CCB_Type`, `ccb::csio`, `sa_softc::device_stats`, `sa_softc::dsreg`, `sa_softc::filemarks`, `sa_softc::flags`, `sa_softc::media_blksize`, `cam_periph::path`, `ccb_hdr::path`, `ccb_scsiio::resid`, `SA_CCB_BUFFER_IO`, `SA_CCB_WAITING`, `SA_FLAG_ERR_PENDING`, `SA_FLAG_FIXED`, `SA_FLAG_TAPE_FROZEN`, `SA_FLAG_TAPE_WRITTEN`, `SA_POSITION_UPDATED`, `saerror()`, `cam_periph::softc`, `ccb_hdr::status`, and `xpt_release_ccb()`.

Referenced by `saerase()`, `sagetparams()`, `saloadunload()`, `samount()`, `saprevent()`, `sardpos()`, `sareservereleaseunit()`, `saretension()`, `sarewind()`, `sasetparams()`, `sasetpos()`, `saspace()`, `sastart()`, and `sawritefilemarks()`.

Here is the call graph for this function:



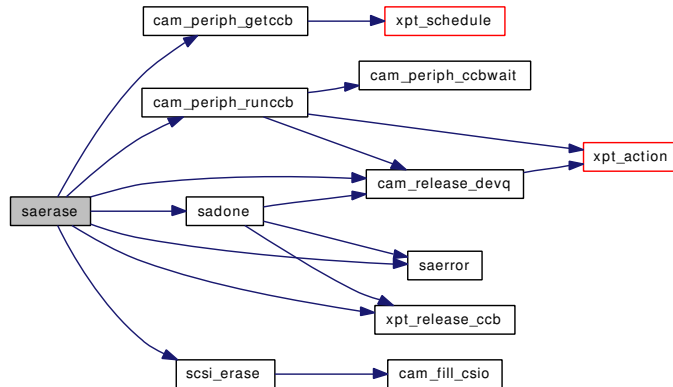
7.35.3.9 static int saerase (struct [cam_periph](#) * *periph*, int *longerase*) [static]

Definition at line 3430 of file `scsi_sa.c`.

References CAM_DEV_QFRZN, cam_periph_getccb(), cam_periph_runcccb(), cam_release_devq(), ccb::ccb_h, ccb::csio, sa_softc::device_stats, sa_softc::dsreg, ERASE_TIMEOUT, MSG_SIMPLE_QTAG, sa_softc::open_rdonly, ccb_hdr::path, sadone(), saerror(), scsi_erase(), cam_periph::softc, SSD_FULL_SIZE, ccb_hdr::status, and xpt_release_ccb().

Referenced by saioctl().

Here is the call graph for this function:



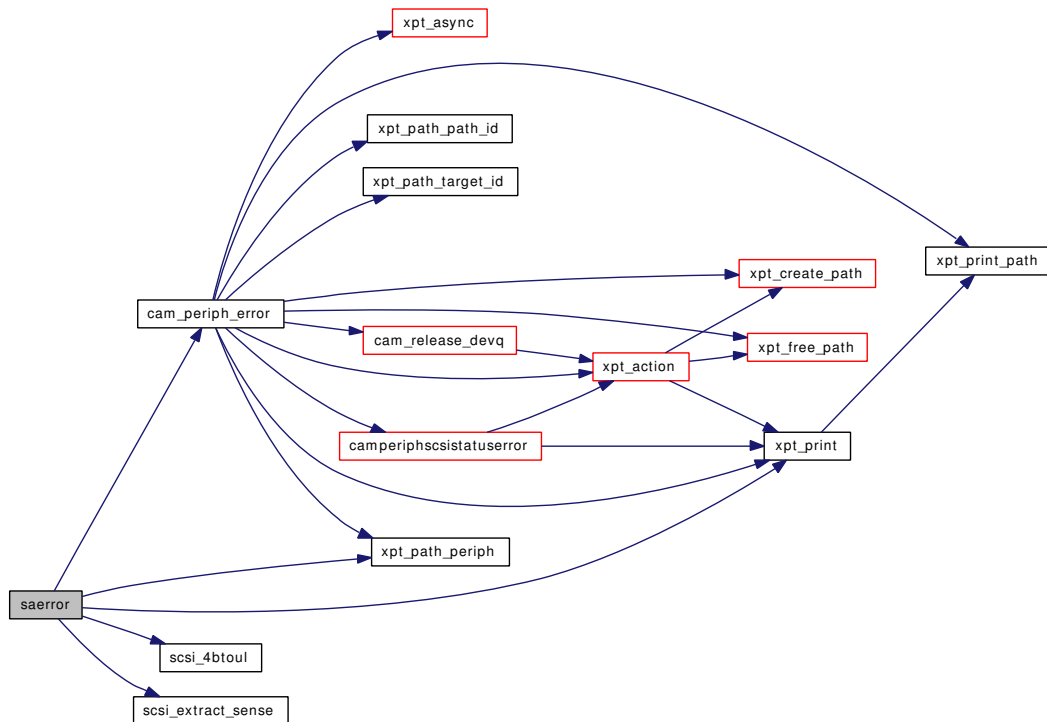
7.35.3.10 static int saerror (union ccb * ccb, u_int32_t cam_flags, u_int32_t sense_flags) [static]

Definition at line 2371 of file scsi_sa.c.

References sa_softc::blkno, CAM_BDR_SENT, CAM_DEBUG, CAM_DEBUG_INFO, cam_periph_error(), CAM_REQ_CMP, CAM SCSI BUS RESET, CAM SCSI STATUS_ERROR, CAM_STATUS_MASK, ccb_scsiio::ccb_h, ccb::ccb_h, CCB_Type, cdb_t::cdb_bytes, ccb_scsiio::cdb_io, ccb_scsiio::cdb_len, ccb::csio, ccb_scsiio::dxfer_len, scsi_sense_data::error_code, scsi_sense_data::extra_len, sa_softc::fileno, scsi_sense_data::flags, sa_softc::flags, scsi_sense_data::info, sa_softc::last_resid_was_io, sa_softc::media_blksize, cam_periph::path, ccb_hdr::path, QFRLS, sa_softc::quirks, ccb_scsiio::resid, ccb_hdr::retry_count, SA_CCB_BUFFER_IO, SA_CCB_WAITING, SA_FLAG_EIO_PENDING, SA_FLAG_EOF_PENDING, SA_FLAG_EOM_PENDING, SA_FLAG_FIXED, SA_POSITION_UPDATED, SA_QUIRK_1FM, SA_WRITE, sa_softc::saved_ccb, scsi_4btoul(), scsi_extract_sense(), ccb_scsiio::sense_data, cam_periph::softc, SSD_EOM, SSD_ERRCODE_VALID, SSD_FILEMARK, SSD_ILI, SSD_KEY_BLANK_CHECK, SSD_KEY_NO_SENSE, SSD_KEY_RESERVED, SSD_KEY_VOLUME_OVERFLOW, ccb_hdr::status, xpt_path_periph(), and xpt_print().

Referenced by sadone(), saerase(), sasetparams(), saioctl(), saloadunload(), samount(), saprevent(), sardpos(), sareserveunit(), saretension(), sarewind(), sasetparams(), sasetpos(), saspace(), and sawritefilemarks().

Here is the call graph for this function:



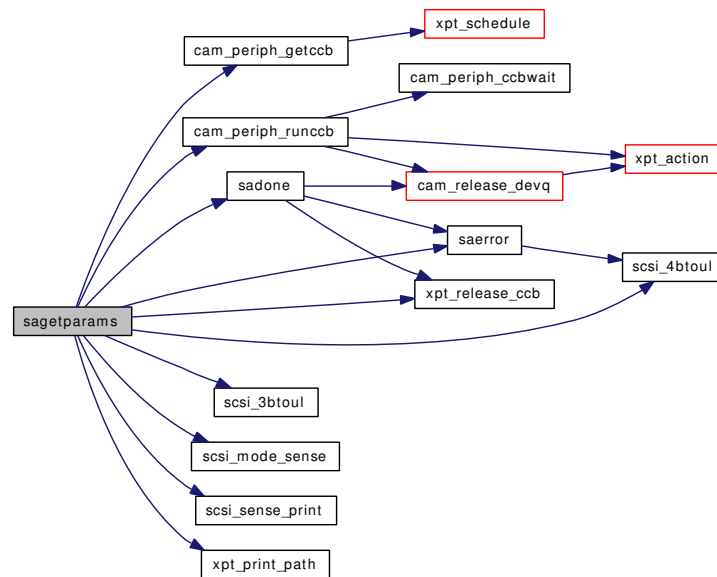
7.35.3.11 `static int sagetparams (struct cam_periph * periph, sa_params params_to_get, u_int32_t * blocksize, u_int8_t * density, u_int32_t * numblocks, int * buff_mode, u_int8_t * write_protect, u_int8_t * speed, int * comp_supported, int * comp_enabled, u_int32_t * comp_algorithm, sa_comp_t * comp_page) [static]`

Definition at line 2565 of file `scsi_sa.c`.

References `scsi_mode_header_6::blk_desc_len`, `CAM_DEBUG_INFO`, `CAM_DEBUGGED`, `cam_periph_getccb()`, `cam_periph_runccb()`, `CAM SCSI STATUS ERROR`, `CAM STATUS MASK`, `ccb::ccb_h`, `scsi_data_compression_page::comp_algorithm`, `ccb::csio`, `scsi_data_compression_page::dce_and_dcc`, `sa_comp_t::dcomp`, `sa_comp_t::dconf`, `scsi_mode_header_6::dev_spec`, `sa_softc::device_stats`, `MSG_SIMPLE_Q_TAG`, `cam_periph::path`, `QFRLS`, `sa_softc::quirks`, `SA_COMP_NONE`, `SA_DATA_COMPRESSION_PAGE`, `SA_DCP_DCC`, `SA_DCP_DCE`, `SA_DEVICE_CONFIGURATION_PAGE`, `SA_PARAM_BLOCKSIZE`, `SA_PARAM_BUFF_MODE`, `SA_PARAM_COMPRESSION`, `SA_PARAM_DENSITY`, `SA_PARAM_NUMBLOCKS`, `SA_PARAM_SPEED`, `SA_PARAM_WP`, `SA_QUIRK_NO_CPAGE`, `SA_QUIRK_NOCOMP`, `sadone()`, `saerror()`, `scsi_3btoul()`, `scsi_4btoul()`, `scsi_mode_sense()`, `scsi_sense_print()`, `SCSIOP_TIMEOUT`, `scsi_dev_conf_page::sel_comp_alg`, `SF_NO_PRINT`, `SMH_SA_BUF_MODE_MASK`, `SMH_SA_SPEED_MASK`, `SMH_SA_WP`, `SMS_PAGE_CTRL_CURRENT`, `SMS_VENDOR_SPECIFIC_PAGE`, `cam_periph::softc`, `SSD_FULL_SIZE`, `ccb_hdr::status`, `xpt_print_path()`, and `xpt_release_ccb()`.

Referenced by `saiocctl()`, `samount()`, and `sasetparams()`.

Here is the call graph for this function:

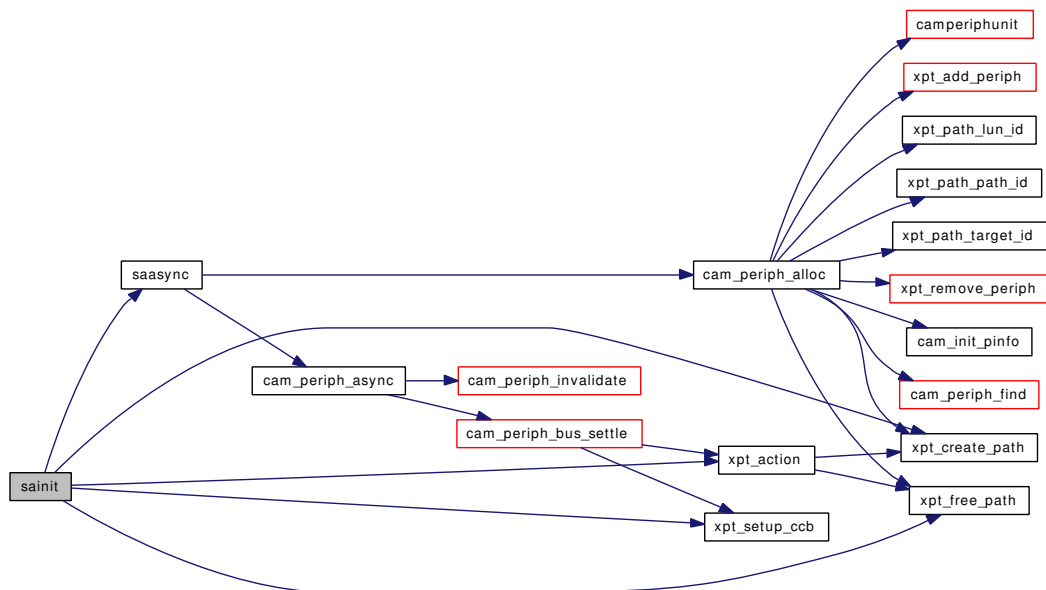


7.35.3.12 static void sainit (void) [static]

Definition at line 1336 of file scsi_sa.c.

References AC_FOUND_DEVICE, ccb_setasync::callback, ccb_setasync::callback_arg, CAM_LUN_WILDCARD, CAM_REQ_CMP, CAM_TARGET_WILDCARD, CAM_XPT_PATH_ID, ccb::ccb_h, ccb_setasync::ccb_h, ccb_setasync::event_enable, ccb_hdr::func_code, saasync(), ccb_hdr::status, xpt_action(), xpt_create_path(), xpt_free_path(), XPT_SASYNC_CB, and xpt_setup_ccb().

Here is the call graph for this function:

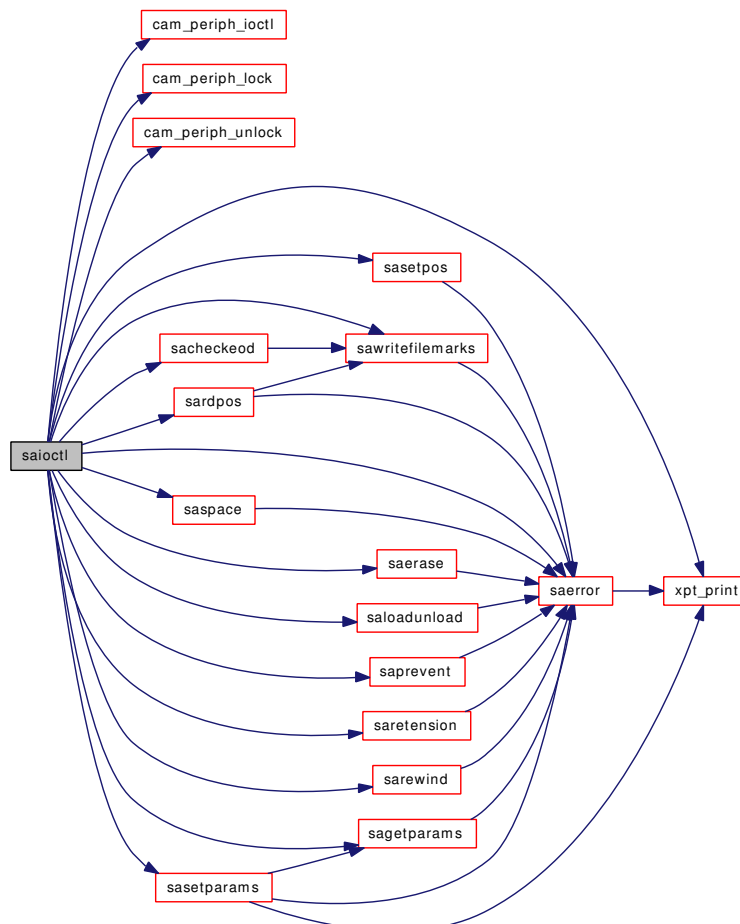


7.35.3.13 static int saioctl (struct cdev * dev, u_long cmd, caddr_t arg, int flag, struct thread * td) [static]

Definition at line 816 of file scsi_sa.c.

References sa_softc::blk_gran, sa_softc::blk_mask, sa_softc::blk_shift, sa_softc::blkno, sa_softc::buffer_mode, CAM_DEBUG, CAM_DEBUG_TRACE, cam_periph_ioctl(), cam_periph_lock(), CAM_PERIPH_LOCKED, cam_periph_unlock(), sa_softc::comp_algorithm, sa_softc::dsreg, sa_softc::errinfo, sa_softc::filemarks, sa_softc::fileno, sa_softc::flags, cam_periph::flags, sa_softc::last_media_blksize, sa_softc::last_resid_was_io, sa_softc::max_blk, sa_softc::media_blksize, sa_softc::media_density, sa_softc::media_numblks, sa_softc::open_pending_mount, cam_periph::path, PENDING_MOUNT_CHECK, PR_ALLOW, sa_softc::quirks, SA_FLAG_COMP_ENABLED, SA_FLAG_COMP_SUPP, SA_FLAG_COMP_UNSUPP, SA_FLAG_EIO_PENDING, SA_FLAG_ERR_PENDING, SA_FLAG_FIXED, SA_FLAG_TAPE_FROZEN, SA_FLAG_TAPE_MOUNTED, SA_FLAG_TAPE_WP, SA_FLAG_TAPE_WRITTEN, SA_IS_CTRL, SA_PARAM_ALL, SA_PARAM_BLOCKSIZE, SA_PARAM_COMPRESSION, SA_PARAM_DENSITY, SA_QUIRK_1FM, SA_QUIRK_2FM, SA_QUIRK_FIXED, SA_QUIRK_NOCOMP, SA_QUIRK_VARIABLE, sacheckeod(), saerase(), saerror(), sagetparams(), saloadunload(), SAMODE, saprevent(), sardpos(), saretension(), sarewind(), sasetparams(), sasetpos(), saspace(), sa_softc::saved_comp_algorithm, sawritefilemarks(), SF_NO_PRINT, cam_periph::softc, sa_softc::speed, SS_BLOCKS, SS_EOD, SS_FILEMARKS, SS_SETMARKS, and xpt_print().

Here is the call graph for this function:



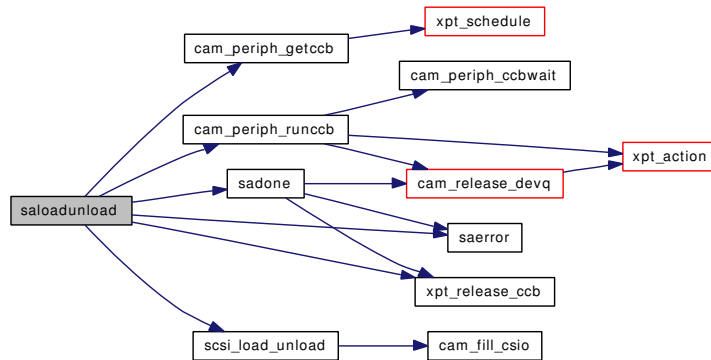
7.35.3.14 static int saloadunload (struct [cam_periph](#) * *periph*, int *load*) [static]

Definition at line 3402 of file `scsi_sa.c`.

References `sa_softc::blkno`, `cam_periph_getccb()`, `cam_periph_runccb()`, `ccb::csio`, `sa_softc::device_stats`, `sa_softc::dsreg`, `sa_softc::fileno`, `MSG_SIMPLE_Q_TAG`, `QFRLS`, `REWIND_TIMEOUT`, `sadone()`, `saerror()`, `scsi_load_unload()`, `cam_periph::softc`, `SSD_FULL_SIZE`, and `xpt_release_ccb()`.

Referenced by `saclose()`, and `saiocctl()`.

Here is the call graph for this function:



7.35.3.15 static int samarkswanted (struct [cam_periph](#) *) [static]

Definition at line 2338 of file `scsi_sa.c`.

References `sa_softc::filemarks`, `sa_softc::flags`, `sa_softc::quirks`, `SA_FLAG_TAPE_WRITTEN`, `SA_QUIRK_2FM`, and `cam_periph::softc`.

Referenced by `sacheckead()`.

7.35.3.16 static int samount (struct [cam_periph](#) *, int, struct [cdev](#) *) [static]

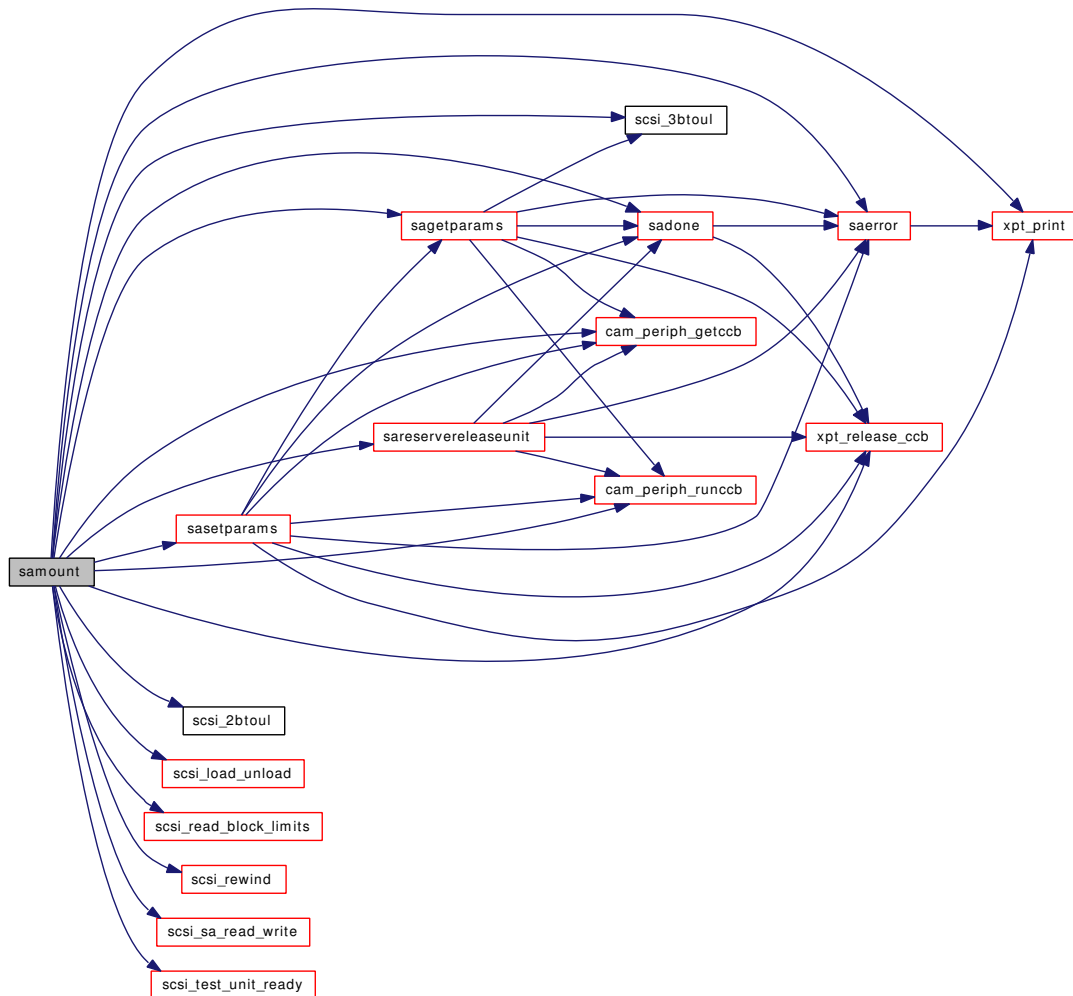
Definition at line 1870 of file `scsi_sa.c`.

References `sa_softc::blk_gran`, `sa_softc::buffer_mode`, `CAM_DEBUG_INFO`, `CAM_DEBUGGED`, `cam_periph_getccb()`, `cam_periph_runccb()`, `CAM_RETRY_SELTO`, `sa_softc::comp_algorithm`, `ccb::csio`, `sa_softc::device_stats`, `sa_softc::filemarks`, `sa_softc::flags`, `IO_TIMEOUT`, `sa_softc::last_media_blksize`, `sa_softc::max_blk`, `scsi_read_block_limits_data::maximum`, `sa_softc::media_blksize`, `sa_softc::media_density`, `sa_softc::media_numblks`, `sa_softc::min_blk`, `scsi_read_block_limits_data::minimum`, `MSG_SIMPLE_Q_TAG`, `cam_periph::path`, `QFRLS`, `sa_softc::quirks`, `RBL_GRAN`, `REWIND_TIMEOUT`, `SA_FLAG_COMP_ENABLED`, `SA_FLAG_COMP_SUPP`, `SA_FLAG_COMP_UNSUPP`, `SA_FLAG_ERR_PENDING`, `SA_FLAG_TAPE_MOUNTED`, `SA_FLAG_TAPE_WP`, `SA_FLAG_TAPE_WRITTEN`, `SA_PARAM_ALL`, `SA_PARAM_DENSITY`, `SA_QUIRK_1FM`, `SA_QUIRK_2FM`, `SA_QUIRK_FIXED`, `SA_QUIRK_NODREAD`, `SA_QUIRK_VARIABLE`, `sadone()`, `saerror()`, `sagetparams()`, `sareservereleaseunit()`, `sasetparams()`, `scsi_2btoul()`, `scsi_3btoul()`, `SCSI_DEFAULT_DENSITY`, `SCSI_DENSITY_HALFINCH_1600`, `SCSI_DENSITY_HALFINCH_6250`, `SCSI_DENSITY_HALFINCH_6250C`, `SCSI_DENSITY_HALFINCH_800`, `SCSI_DENSITY_HALFINCH_PE`, `SCSI_DENSITY_QIC_11_4TRK`, `SCSI_DENSITY_QIC_11_9TRK`, `SCSI_DENSITY_QIC_120`, `SCSI_DENSITY_QIC_1320`, `SCSI_DENSITY_QIC_150`, `SCSI_DENSITY_QIC_24`, `SCSI_DENSITY_QIC_2GB`, `SCSI_DENSITY_QIC_3080`, `SCSI_DENSITY_QIC_4GB`,

SCSI_DENSITY_QIC_525_320, scsi_load_unload(), scsi_read_block_limits(), sa_softc::scsi_rev, SCSI_REV_SPC, scsi_rewind(), scsi_sa_read_write(), scsi_test_unit_ready(), SCSIOP_TIMEOUT, SF_NO_PRINT, SF_RETRY_UA, cam_periph::softc, sa_softc::speed, SSD_FULL_SIZE, UNUSED_PARAMETER, xpt_print(), and xpt_release_ccb().

Referenced by saopen(), and sastrategy().

Here is the call graph for this function:

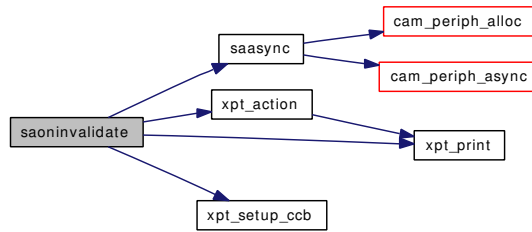


7.35.3.17 static void saoninvalidate (struct cam_periph *periph) [static]

Definition at line 1370 of file scsi_sa.c.

References sa_softc::bio_queue, ccb_setasync::callback, ccb_setasync::callback_arg, ccb_setasync::ccb_h, ccb_setasync::event_enable, sa_softc::flags, ccb_hdr::func_code, cam_periph::path, sa_softc::queue_count, SA_FLAG_INVALID, saasync(), cam_periph::softc, xpt_action(), xpt_print(), XPT_SASYNC_CB, and xpt_setup_ccb().

Here is the call graph for this function:

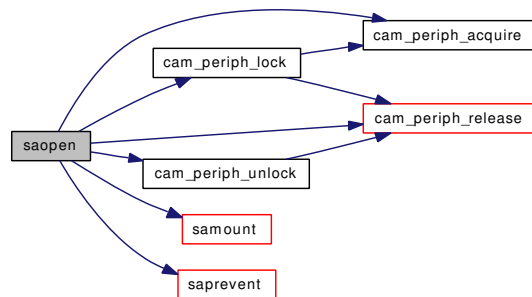


7.35.3.18 static int saopen (struct cdev * dev, int flags, int fmt, struct thread * td) [static]

Definition at line 444 of file scsi_sa.c.

References CAM_DEBUG, CAM_DEBUG_INFO, CAM_DEBUG_TRACE, cam_periph_acquire(), cam_periph_lock(), cam_periph_release(), cam_periph_unlock(), CAM_REQ_CMP, sa_softc::ctrl_mode, sa_softc::flags, sa_softc::open_pending_mount, sa_softc::open_ronly, cam_periph::path, PR_PREVENT, SA_FLAG_INVALID, SA_FLAG_OPEN, SA_IS_CTRL, samount(), saprevent(), SAUNIT, and cam_periph::softc.

Here is the call graph for this function:



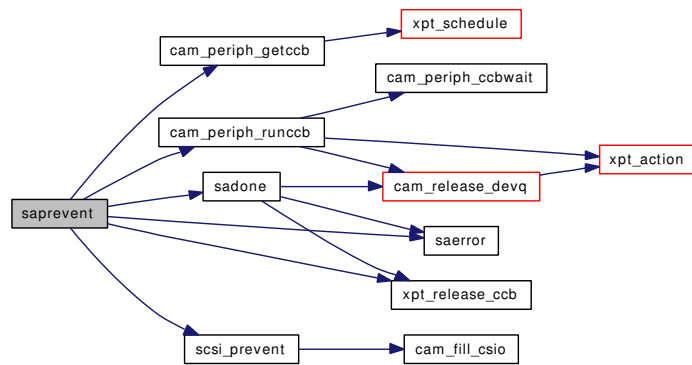
7.35.3.19 static void saprevent (struct cam_periph * periph, int action) [static]

Definition at line 3069 of file scsi_sa.c.

References CAM_DEBUG_INFO, CAM_DEBUGGED, cam_periph_getccb(), cam_periph_runccb(), ccb::csio, sa_softc::device_stats, sa_softc::flags, MSG_SIMPLE_Q_TAG, cam_periph::path, PR_ALLOW, PR_PREVENT, QFRLS, SA_FLAG_TAPE_LOCKED, sadone(), saerror(), scsi_prevent(), SCSIOP_TIMEOUT, SF_QUIET_IR, cam_periph::softc, SSD_FULL_SIZE, and xpt_release_ccb().

Referenced by saclose(), saioc1(), saopen(), and sastrategy().

Here is the call graph for this function:



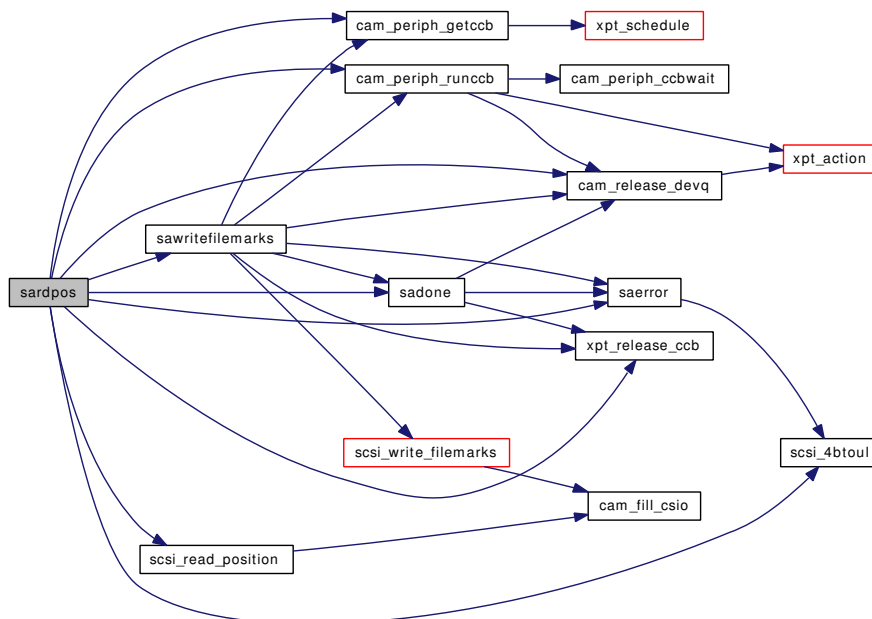
7.35.3.20 static int sardpos (struct **cam_periph** * *periph*, int, u_int32_t *) [static]

Definition at line 3261 of file scsi_sa.c.

References CAM_DEV_QFRZN, cam_periph_getccb(), cam_periph_runccb(), cam_release_devq(), ccb::ccb_h, ccb::csio, sa_softc::device_stats, sa_softc::dsreg, scsi_tape_position_data::firstblk, scsi_tape_position_data::flags, sa_softc::flags, MSG_SIMPLE_Q_TAG, ccb_hdr::path, SA_FLAG_TAPE_WRITTEN, SA_RPOS_UNCERTAIN, sadone(), saerror(), sawritefilemarks(), scsi_4btoul(), scsi_read_position(), SCSIOP_TIMEOUT, cam_periph::softc, SSD_FULL_SIZE, ccb_hdr::status, and xpt_release_ccb().

Referenced by saioctl().

Here is the call graph for this function:

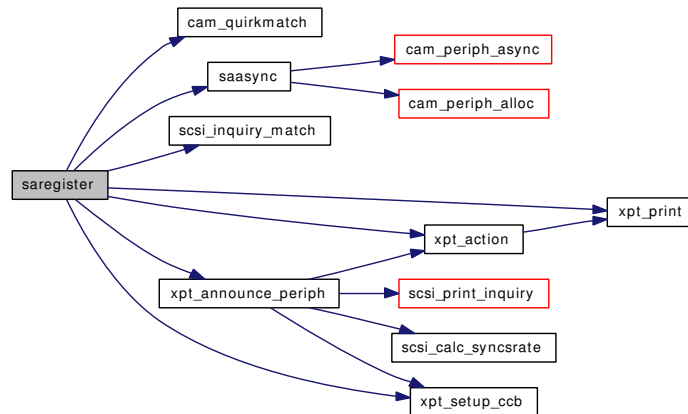


7.35.3.21 static `cam_status` `saregister` (`struct cam_periph *periph`, `void *arg`) [static]

Definition at line 1476 of file `scsi_sa.c`.

References `AC_LOST_DEVICE`, `ccb_setasync::callback`, `ccb_setasync::callback_arg`, `cam_quirkmatch()`, `CAM_REQ_CMP`, `CAM_REQ_CMP_ERR`, `ccb_setasync::ccb_h`, `ccb_setasync::event_enable`, `ccb_hdr::func_code`, `ccb_getdev::inq_data`, `cam_periph::path`, `cam_periph::periph_name`, `SA_ATYPE_ER`, `SA_ATYPE_NR`, `SA_ATYPE_R`, `SA_CTLDEV`, `SA_NOT_CTLDEV`, `SA_NUM_MODES`, `SA_QUIRK_NONE`, `sa_quirk_table`, `SA_STATE_NORMAL`, `saasync()`, `SAMINOR`, `scsi_inquiry_match()`, `SID_ANSI_REV`, `SID_TYPE`, `cam_periph::softc`, `cam_periph::unit_number`, `xpt_action()`, `xpt_announce_periph()`, `xpt_print()`, `XPT_SASYNC_CB`, and `xpt_setup_ccb()`.

Here is the call graph for this function:



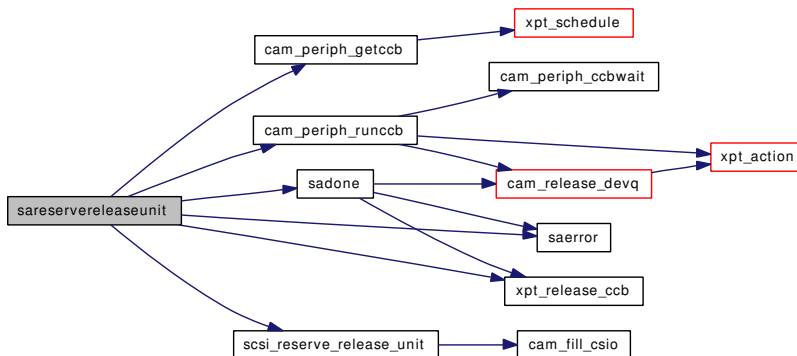
7.35.3.22 static `int` `sareservereleaseunit` (`struct cam_periph *periph`, `int reserve`) [static]

Definition at line 3371 of file `scsi_sa.c`.

References `cam_periph_getccb()`, `cam_periph_runccb()`, `ccb::csio`, `sa_softc::device_stats`, `sa_softc::dsreg`, `MSG_SIMPLE_Q_TAG`, `QFRLS`, `sadone()`, `saerror()`, `scsi_reserve_release_unit()`, `SCSIOP_TIMEOUT`, `SF_NO_PRINT`, `SF_RETRY_UA`, `cam_periph::softc`, `SSD_FULL_SIZE`, and `xpt_release_ccb()`.

Referenced by `saclose()`, and `samount()`.

Here is the call graph for this function:



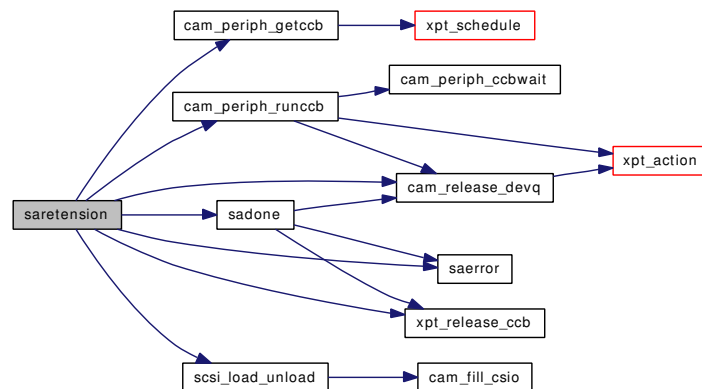
7.35.3.23 static int saretension (struct cam_periph * periph) [static]

Definition at line 3342 of file scsi_sa.c.

References sa_softc::blkno, CAM_DEV_QFRZN, cam_periph_getccb(), cam_periph_runccb(), cam_release_devq(), ccb::ccb_h, ccb::csio, sa_softc::device_stats, sa_softc::dsreg, ERASE_TIMEOUT, sa_softc::fileno, MSG_SIMPLE_Q_TAG, ccb_hdr::path, sadone(), saerror(), scsi_load_unload(), cam_periph::softc, SSD_FULL_SIZE, ccb_hdr::status, and xpt_release_ccb().

Referenced by saioctl().

Here is the call graph for this function:



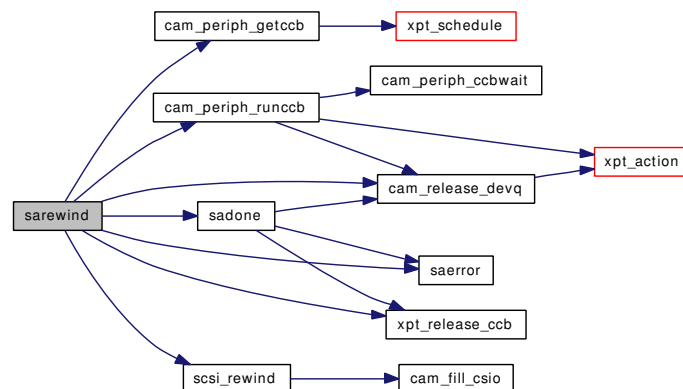
7.35.3.24 static int sarewind (struct cam_periph * periph) [static]

Definition at line 3109 of file scsi_sa.c.

References sa_softc::blkno, CAM_DEV_QFRZN, cam_periph_getccb(), cam_periph_runccb(), cam_release_devq(), ccb::ccb_h, ccb::csio, sa_softc::device_stats, sa_softc::dsreg, sa_softc::fileno, MSG_SIMPLE_Q_TAG, ccb_hdr::path, REWIND_TIMEOUT, sadone(), saerror(), scsi_rewind(), cam_periph::softc, SSD_FULL_SIZE, ccb_hdr::status, and xpt_release_ccb().

Referenced by saclose(), and saioctl().

Here is the call graph for this function:



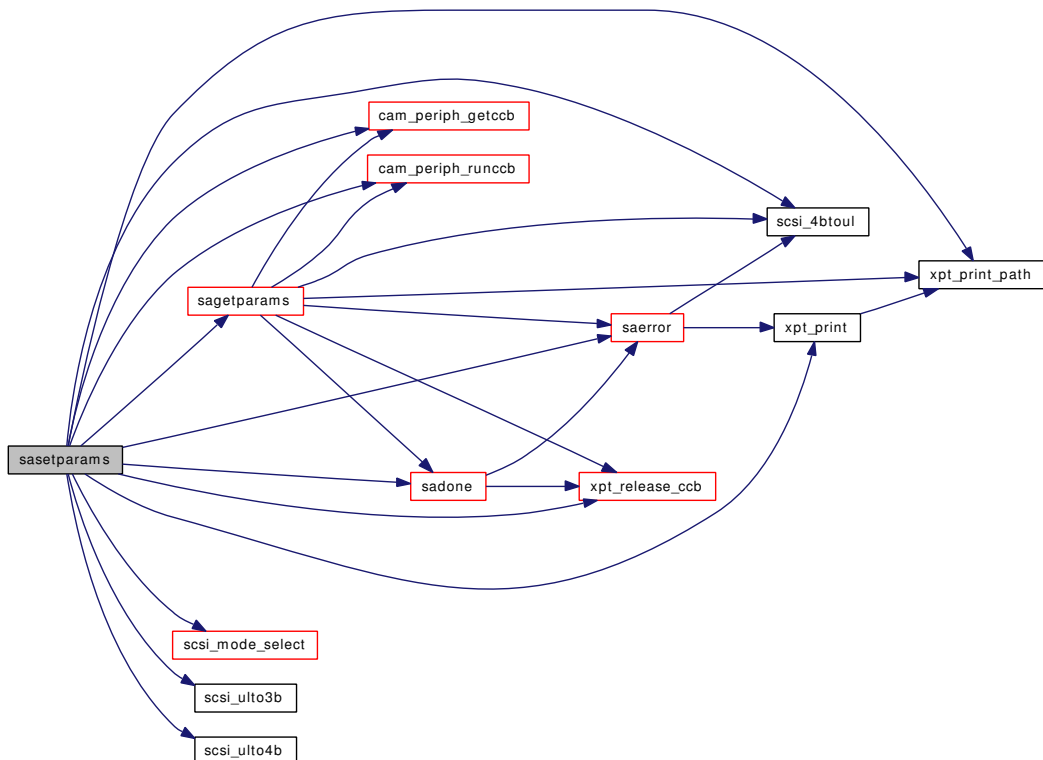
7.35.3.25 `static int sasetparams (struct cam_periph * periph, sa_params params_to_set, u_int32_t blocksize, u_int8_t density, u_int32_t comp_algorithm, u_int32_t sense_flags)`
`[static]`

Definition at line 2758 of file `scsi_sa.c`.

References `scsi_mode_header_6::blk_desc_len`, `sa_softc::buffer_mode`, `CAM_DEBUG_INFO`, `CAM_DEBUGGED`, `cam_periph_getccb()`, `cam_periph_runccb()`, `ccb::ccb_h`, `sa_softc::comp_algorithm`, `scsi_data_compression_page::comp_algorithm`, `ccb::csio`, `scsi_mode_header_6::data_length`, `scsi_data_compression_page::dce_and_dcc`, `sa_comp_t::dcomp`, `sa_comp_t::dconf`, `scsi_data_compression_page::dde_and_red`, `scsi_data_compression_page::decomp_algorithm`, `scsi_mode_header_6::dev_spec`, `sa_softc::device_stats`, `sa_softc::flags`, `sa_comp_t::hdr`, `sa_softc::media_density`, `scsi_mode_header_6::medium_type`, `MSG_SIMPLE_Q_TAG`, `sa_comp_t::pagecode`, `cam_periph::path`, `QFRLS`, `ccb_hdr::retry_count`, `SA_COMP_NONE`, `SA_DATA_COMPRESSION_PAGE`, `SA_DCP_DCC`, `SA_DCP_DCE`, `SA_DCP_DDE`, `SA_DEVICE_CONFIGURATION_PAGE`, `SA_FLAG_COMP_ENABLED`, `SA_PARAM_BLOCKSIZE`, `SA_PARAM_COMPRESSION`, `SA_PARAM_DENSITY`, `SA_PARAM_NONE`, `SA_PARAM_SPEED`, `sadone()`, `saerror()`, `sagetparams()`, `sa_softc::saved_comp_algorithm`, `scsi_4btoul()`, `scsi_mode_select()`, `sa_softc::scsi_rev`, `SCSI_REV_CCS`, `SCSI_SAME_DENSITY`, `scsi_ulto3b()`, `scsi_ulto4b()`, `SCSIOP_TIMEOUT`, `scsi_dev_conf_page::sel_comp_alg`, `SMH_SA_BUF_MODE_SIBUF`, `cam_periph::softc`, `SSD_FULL_SIZE`, `xpt_print()`, `xpt_print_path()`, and `xpt_release_ccb()`.

Referenced by `saiocctl()`, and `samount()`.

Here is the call graph for this function:



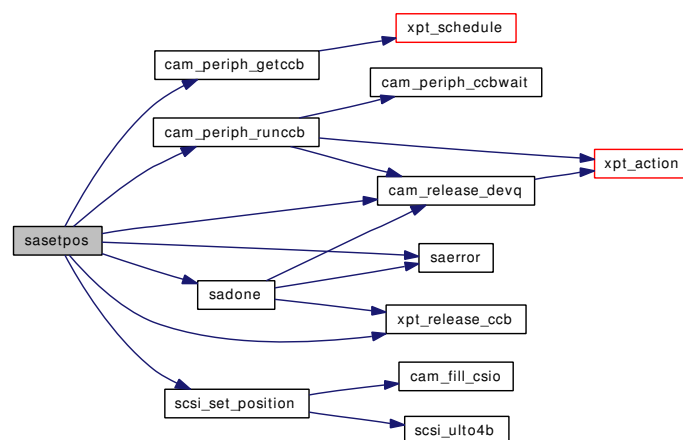
7.35.3.26 static int sasetpos (struct **cam_periph** * *periph*, int, u_int32_t *) [static]

Definition at line 3306 of file scsi_sa.c.

References sa_softc::blkno, CAM_DEV_QFRZN, cam_periph_getccb(), cam_periph_runccb(), cam_release_devq(), ccb::ccb_h, ccb::csio, sa_softc::device_stats, sa_softc::dsreg, sa_softc::fileno, MSG_SIMPLE_Q_TAG, ccb_hdr::path, sadone(), saerror(), scsi_set_position(), cam_periph::softc, SPACE_TIMEOUT, SSD_FULL_SIZE, ccb_hdr::status, and xpt_release_ccb().

Referenced by saioctl().

Here is the call graph for this function:



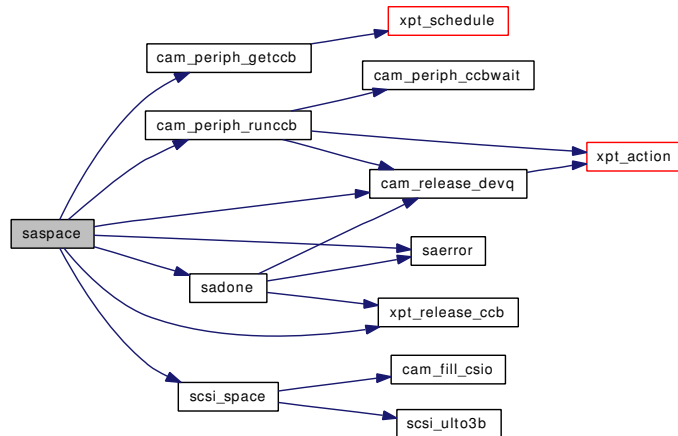
7.35.3.27 static int saspace (struct **cam_periph** * *periph*, int *count*, **scsi_space_code** *code*) [static]

Definition at line 3139 of file scsi_sa.c.

References sa_softc::blkno, CAM_DEV_QFRZN, cam_periph_getccb(), cam_periph_runccb(), cam_release_devq(), ccb::ccb_h, ccb::csio, sa_softc::device_stats, sa_softc::dsreg, sa_softc::fileno, MSG_SIMPLE_Q_TAG, ccb_hdr::path, sadone(), saerror(), scsi_space(), cam_periph::softc, SPACE_TIMEOUT, SS_BLOCKS, SS_EOD, SS_FILEMARKS, SS_SETMARKS, SSD_FULL_SIZE, ccb_hdr::status, and xpt_release_ccb().

Referenced by saclose(), and saioctl().

Here is the call graph for this function:

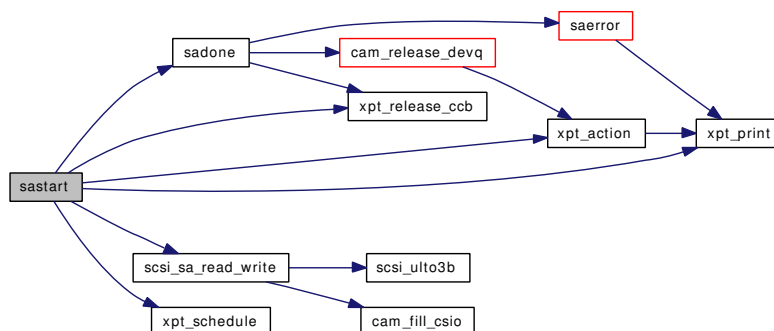


7.35.3.28 static void sastart (struct **cam_periph** * *periph*, union **ccb** * *start_ccb*) [static]

Definition at line 1598 of file scsi_sa.c.

References sa_softc::bio_queue, sa_softc::blk_shift, CAM_DEBUG, CAM_DEBUG_INFO, CAM_DEBUG_PRINT, CAM_DEBUG_SUBTRACE, CAM_DEBUG_TRACE, CAM_PRIORITY_NONE, ccb::ccb_h, ccb::csio, sa_softc::device_stats, sa_softc::dsreg, sa_softc::flags, cam_periph::immediate_priority, IO_TIMEOUT, sa_softc::media_blksize, MSG_SIMPLE_Q_TAG, ccb_hdr::path, cam_periph::path, cam_periph::pinfo, cam_pinfo::priority, sa_softc::queue_count, SA_CCB_BUFFER_IO, SA_CCB_WAITING, SA_FLAG_EIO_PENDING, SA_FLAG_EOF_PENDING, SA_FLAG_EOM_PENDING, SA_FLAG_ERR_PENDING, SA_FLAG_FIXED, SA_POSITION_UPDATED, SA_STATE_ABNORMAL, SA_STATE_NORMAL, sadone(), scsi_sa_read_write(), Set_CCB_Type, cam_periph::softc, SSD_FULL_SIZE, sa_softc::state, xpt_action(), xpt_print(), xpt_release_ccb(), and xpt_schedule().

Here is the call graph for this function:



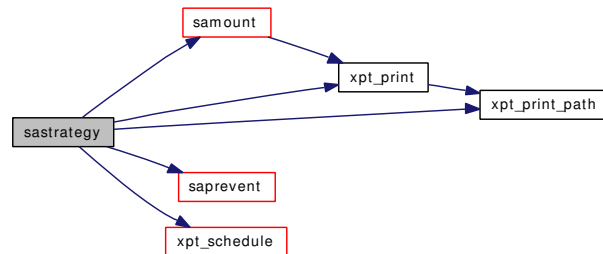
7.35.3.29 static void sastrategy (struct **bio** * *bp*) [static]

Definition at line 676 of file scsi_sa.c.

References sa_softc::bio_queue, sa_softc::blk_gran, sa_softc::blk_mask, CAM_DEBUG, CAM_DEBUG_INFO, sa_softc::flags, sa_softc::max_blk, sa_softc::min_blk, sa_softc::open_pending_mount,

sa_softc::open_rdonly, cam_periph::path, PR_PREVENT, sa_softc::queue_count, SA_FLAG_FIXED, SA_FLAG_INVALID, SA_FLAG_TAPE_FROZEN, SA_IS_CTRL, samount(), saprevent(), cam_periph::softc, xpt_print(), xpt_print_path(), and xpt_schedule().

Here is the call graph for this function:



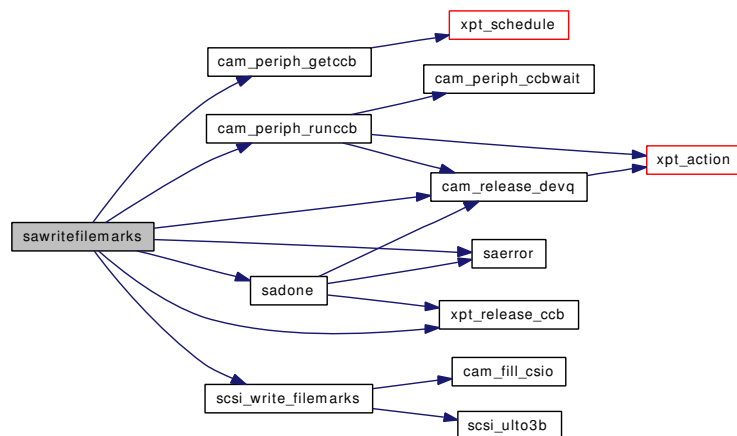
7.35.3.30 static int sawritefilemarks (struct **cam_periph** * *periph*, int *nmarks*, int *setmarks*) [static]

Definition at line 3212 of file scsi_sa.c.

References sa_softc::blkno, CAM_DEV_QFRZN, cam_periph_getccb(), cam_periph_runccb(), cam_release_devq(), ccb::ccb_h, ccb::csio, sa_softc::device_stats, sa_softc::dsreg, sa_softc::filemarks, sa_softc::fileno, IO_TIMEOUT, MSG_SIMPLE_Q_TAG, sa_softc::open_rdonly, ccb_hdr::path, sadone(), saerror(), scsi_write_filemarks(), cam_periph::softc, SSD_FULL_SIZE, ccb_hdr::status, and xpt_release_ccb().

Referenced by sacheckeod(), saioctl(), and sardpos().

Here is the call graph for this function:



7.35.3.31 void **scsi_erase** (struct **ccb_scsiio** * *csio*, u_int32_t *retries*, void(*) (struct **cam_periph** *, union **ccb** *) *cbfcn*, u_int8_t *tag_action*, int *immediate*, int *long_erase*, u_int8_t *sense_len*, u_int32_t *timeout*)

Definition at line 3618 of file scsi_sa.c.

References CAM_DIR_NONE, cam_fill_csio(), cdb_t::cdb_bytes, ccb_scsiio::cdb_io, ERASE, SE_IMMED, and SE_LONG.

Referenced by saerase().

Here is the call graph for this function:



7.35.3.32 void **scsi_load_unload** (struct **ccb_scsiio** * *csio*, u_int32_t *retries*, void(*) (struct **cam_periph** *, union **ccb** *) *cbfcnp*, u_int8_t *tag_action*, int *immediate*, int *eot*, int *reten*, int *load*, u_int8_t *sense_len*, u_int32_t *timeout*)

Definition at line 3504 of file scsi_sa.c.

References CAM_DIR_NONE, cam_fill_csio(), cdb_t::cdb_bytes, ccb_scsiio::cdb_io, LOAD_UNLOAD, SLU_EOT, SLU_IMMED, SLU_LOAD, and SLU_RETEN.

Referenced by saloadunload(), samount(), and saretension().

Here is the call graph for this function:



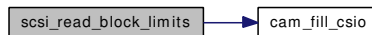
7.35.3.33 void **scsi_read_block_limits** (struct **ccb_scsiio** * *csio*, u_int32_t *retries*, void(*) (struct **cam_periph** *, union **ccb** *) *cbfcnp*, u_int8_t *tag_action*, struct **scsi_read_block_limits_data** * *rlimit_buf*, u_int8_t *sense_len*, u_int32_t *timeout*)

Definition at line 3462 of file scsi_sa.c.

References CAM_DIR_IN, cam_fill_csio(), cdb_t::cdb_bytes, ccb_scsiio::cdb_io, and READ_BLOCK_LIMITS.

Referenced by samount().

Here is the call graph for this function:



7.35.3.34 void **scsi_read_position** (struct **ccb_scsiio** * *csio*, u_int32_t *retries*, void(*) (struct **cam_periph** *, union **ccb** *) *cbfcnp*, u_int8_t *tag_action*, int *hardsoft*, struct **scsi_tape_position_data** * *sbp*, u_int8_t *sense_len*, u_int32_t *timeout*)

Definition at line 3644 of file scsi_sa.c.

References CAM_DIR_IN, cam_fill_csio(), cdb_t::cdb_bytes, ccb_scsiio::cdb_io, and READ_POSITION.

Referenced by sardpos().

Here is the call graph for this function:



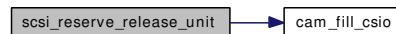
7.35.3.35 void **scsi_reserve_release_unit** (struct **ccb_scsiio** * *csio*, u_int32_t *retries*, void(*) (struct **cam_periph** *, union **ccb** *) *cbfcnp*, u_int8_t *tag_action*, int *third_party*, int *third_party_id*, u_int8_t *sense_len*, u_int32_t *timeout*, int *reserve*)

Definition at line 3591 of file *scsi_sa.c*.

References CAM_DIR_NONE, *cam_fill_csio()*, *cdb_t::cdb_bytes*, *ccb_scsiio::cdb_io*, RELEASE_UNIT, RESERVE_UNIT, SRRU_3RD_MASK, SRRU_3RD_PARTY, and SRRU_3RD_SHAMT.

Referenced by *sareservereleaseunit()*.

Here is the call graph for this function:



7.35.3.36 void **scsi_rewind** (struct **ccb_scsiio** * *csio*, u_int32_t *retries*, void(*) (struct **cam_periph** *, union **ccb** *) *cbfcnp*, u_int8_t *tag_action*, int *immediate*, u_int8_t *sense_len*, u_int32_t *timeout*)

Definition at line 3529 of file *scsi_sa.c*.

References CAM_DIR_NONE, *cam_fill_csio()*, *cdb_t::cdb_bytes*, *ccb_scsiio::cdb_io*, REWIND, and SREW_IMMED.

Referenced by *samount()*, and *sarewind()*.

Here is the call graph for this function:



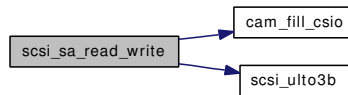
7.35.3.37 void **scsi_sa_read_write** (struct **ccb_scsiio** * *csio*, u_int32_t *retries*, void(*) (struct **cam_periph** *, union **ccb** *) *cbfcnp*, u_int8_t *tag_action*, int *readop*, int *sli*, int *fixed*, u_int32_t *length*, u_int8_t * *data_ptr*, u_int32_t *dxfer_len*, u_int8_t *sense_len*, u_int32_t *timeout*)

Definition at line 3480 of file *scsi_sa.c*.

References CAM_DIR_IN, CAM_DIR_OUT, *cam_fill_csio()*, *cdb_t::cdb_bytes*, *ccb_scsiio::cdb_io*, *scsi_sa_rw::control*, *scsi_sa_rw::length*, *scsi_sa_rw::opcode*, SA_READ, SA_WRITE, SAR_SLI, SARW_FIXED, *scsi_ulto3b()*, and *scsi_sa_rw::sli_fixed*.

Referenced by *samount()*, and *sastart()*.

Here is the call graph for this function:



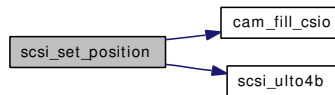
7.35.3.38 void `scsi_set_position` (struct `ccb_scsiio` * *csio*, `u_int32_t` *retries*, void(*) (struct `cam_periph` *, union `ccb` *) *cbfcnp*, `u_int8_t` *tag_action*, int *hardsoft*, `u_int32_t` *blkno*, `u_int8_t` *sense_len*, `u_int32_t` *timeout*)

Definition at line 3664 of file `scsi_sa.c`.

References `CAM_DIR_NONE`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `LOCATE`, `SA_SPOS_BT`, and `scsi_ulito4b()`.

Referenced by `sasetpos()`.

Here is the call graph for this function:



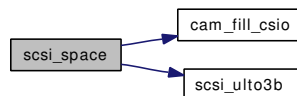
7.35.3.39 void `scsi_space` (struct `ccb_scsiio` * *csio*, `u_int32_t` *retries*, void(*) (struct `cam_periph` *, union `ccb` *) *cbfcnp*, `u_int8_t` *tag_action*, `scsi_space_code` *code*, `u_int32_t` *count*, `u_int8_t` *sense_len*, `u_int32_t` *timeout*)

Definition at line 3547 of file `scsi_sa.c`.

References `CAM_DIR_NONE`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `scsi_space::code`, `scsi_space::control`, `scsi_space::count`, `scsi_space::opcode`, `scsi_ulito3b()`, and `SPACE`.

Referenced by `saspace()`.

Here is the call graph for this function:



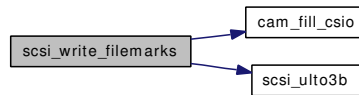
7.35.3.40 void `scsi_write_filemarks` (struct `ccb_scsiio` * *csio*, `u_int32_t` *retries*, void(*) (struct `cam_periph` *, union `ccb` *) *cbfcnp*, `u_int8_t` *tag_action*, int *immediate*, int *setmark*, `u_int32_t` *num_marks*, `u_int8_t` *sense_len*, `u_int32_t` *timeout*)

Definition at line 3565 of file `scsi_sa.c`.

References `CAM_DIR_NONE`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `scsi_ulito3b()`, `SWFMRK_IMMED`, `SWFMRK_WSMK`, and `WRITE_FILEMARKS`.

Referenced by `sawritefilemarks()`.

Here is the call graph for this function:



7.35.4 Variable Documentation

7.35.4.1 struct `cdevsw sa_cdevsw` [static]

Initial value:

```

{
    .d_version =    D_VERSION,
    .d_open =      saopen,
    .d_close =     saclose,
    .d_read =      physread,
    .d_write =     physwrite,
    .d_ioctl =     saioctl,
    .d_strategy =  sastrategy,
    .d_name =      "sa",
    .d_flags =     D_TAPE | D_NEEDGIANT,
}
  
```

Definition at line 431 of file `scsi_sa.c`.

7.35.4.2 struct `sa_quirk_entry sa_quirk_table[]` [static]

Definition at line 271 of file `scsi_sa.c`.

Referenced by `saregister()`.

7.35.4.3 `periph_dtor_t sacleanup` [static]

Definition at line 378 of file `scsi_sa.c`.

7.35.4.4 `d_close_t saclose` [static]

Definition at line 372 of file `scsi_sa.c`.

7.35.4.5 struct `periph_driver sadriver` [static]

Initial value:

```

{
    sainit, "sa",
    TAILQ_HEAD_INITIALIZER(sadriver.units), 0
}
  
```

Definition at line 417 of file `scsi_sa.c`.

7.35.4.6 `periph_init_t sainit` [static]

Definition at line 375 of file `scsi_sa.c`.

7.35.4.7 `d_ioctl_t saioctl` [static]

Definition at line 374 of file `scsi_sa.c`.

7.35.4.8 `periph_oninv_t saoninvalidate` [static]

Definition at line 377 of file `scsi_sa.c`.

7.35.4.9 `d_open_t saopen` [static]

Definition at line 371 of file `scsi_sa.c`.

7.35.4.10 `periph_ctor_t saregister` [static]

Definition at line 376 of file `scsi_sa.c`.

7.35.4.11 `periph_start_t sastart` [static]

Definition at line 379 of file `scsi_sa.c`.

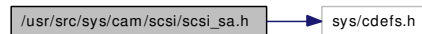
7.35.4.12 `d_strategy_t sastrategy` [static]

Definition at line 373 of file `scsi_sa.c`.

7.36 /usr/src/sys/cam/scsi/scsi_sa.h File Reference

```
#include <sys/cdefs.h>
```

Include dependency graph for scsi_sa.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- struct [scsi_read_block_limits](#)
- struct [scsi_read_block_limits_data](#)
- struct [scsi_sa_rw](#)
- struct [scsi_load_unload](#)
- struct [scsi_rewind](#)
- struct [scsi_space](#)
- struct [scsi_write_filemarks](#)
- struct [scsi_reserve_release_unit](#)
- struct [scsi_erase](#)
- struct [scsi_dev_conf_page](#)
- struct [scsi_data_compression_page](#)
- union [sa_comp_t](#)
- struct [scsi_tape_read_position](#)
- struct [scsi_tape_position_data](#)
- struct [scsi_tape_locate](#)

Defines

- #define [_SCSI SCSI_SA_H](#) 1
- #define [RBL_GRAN_MASK](#) 0x1F
- #define [RBL_GRAN](#)(rblim) ((rblim) → gran & RBL_GRAN_MASK)
- #define [SAR_SLI](#) 0x02
- #define [SARW_FIXED](#) 0x01
- #define [SLU_IMMED](#) 0x01
- #define [SLU_EOT](#) 0x04
- #define [SLU_RETEN](#) 0x02
- #define [SLU_LOAD](#) 0x01
- #define [SREW_IMMED](#) 0x01
- #define [SREW_IMMED](#) 0x01
- #define [SWFMRK_IMMED](#) 0x01
- #define [SWFMRK_WSMK](#) 0x02
- #define [SRRU_LUN_MASK](#) 0xE0
- #define [SRRU_3RD_PARTY](#) 0x10

- #define `SRRU_3RD_SHAMT` 1
- #define `SRRU_3RD_MASK` 0xE
- #define `SE_LUN_MASK` 0xE0
- #define `SE_LONG` 0x1
- #define `SE_IMMED` 0x2
- #define `SMH_SA_WP` 0x80
- #define `SMH_SA_BUF_MODE_MASK` 0x70
- #define `SMH_SA_BUF_MODE_NOBUF` 0x00
- #define `SMH_SA_BUF_MODE_SIBUF` 0x10
- #define `SMH_SA_BUF_MODE_MIBUF` 0x20
- #define `SMH_SA_SPEED_MASK` 0x0F
- #define `SMH_SA_SPEED_DEFAULT` 0x00
- #define `SA_DEVICE_CONFIGURATION_PAGE` 0x10
- #define `SA_MEDIUM_PARTITION_PAGE_1` 0x11
- #define `SA_MEDIUM_PARTITION_PAGE_2` 0x12
- #define `SA_MEDIUM_PARTITION_PAGE_3` 0x13
- #define `SA_MEDIUM_PARTITION_PAGE_4` 0x14
- #define `SA_DATA_COMPRESSION_PAGE` 0x0f
- #define `SA_DBR` 0x80
- #define `SA_BIS` 0x40
- #define `SA_RSMK` 0x20
- #define `SA_AVC` 0x10
- #define `SA_SOCF_MASK` 0xc0
- #define `SA_RBO` 0x20
- #define `SA_REW` 0x10
- #define `SA_COMP_NONE` 0x00
- #define `SA_COMP_DEFAULT` 0x01
- #define `SA_ASOC_WP` 0x04
- #define `SA_PERS_WP` 0x02
- #define `SA_PERM_WP` 0x01
- #define `SA_DCP_DCE` 0x80
- #define `SA_DCP_DCC` 0x40
- #define `SA_DCP_DDE` 0x80
- #define `SA_DCP_RED_MASK` 0x60
- #define `SA_DCP_RED_SHAMT` 5
- #define `SA_DCP_RED_0` 0x00
- #define `SA_DCP_RED_1` 0x20
- #define `SA_DCP_RED_2` 0x40
- #define `SA_RPOS_BOP` 0x80
- #define `SA_RPOS_EOP` 0x40
- #define `SA_RPOS_BCU` 0x20
- #define `SA_RPOS_BYCU` 0x10
- #define `SA_RPOS_BPU` 0x04
- #define `SA_RPOS_PERR` 0x02
- #define `SA_RPOS_UNCERTAIN` `SA_RPOS_BPU`
- #define `SA_SPOS_IMMED` 0x01
- #define `SA_SPOS_CP` 0x02
- #define `SA_SPOS_BT` 0x04
- #define `REWIND` 0x01
- #define `READ_BLOCK_LIMITS` 0x05

- #define SA_READ 0x08
- #define SA_WRITE 0x0A
- #define WRITE_FILEMARKS 0x10
- #define SPACE 0x11
- #define RESERVE_UNIT 0x16
- #define RELEASE_UNIT 0x17
- #define ERASE 0x19
- #define LOAD_UNLOAD 0x1B
- #define LOCATE 0x2B
- #define READ_POSITION 0x34
- #define SCSI_DENSITY_HALFINCH_800 0x01
- #define SCSI_DENSITY_HALFINCH_1600 0x02
- #define SCSI_DENSITY_HALFINCH_6250 0x03
- #define SCSI_DENSITY_HALFINCH_6250C 0xC3
- #define SCSI_DENSITY_QIC_11_4TRK 0x04
- #define SCSI_DENSITY_QIC_11_9TRK 0x84
- #define SCSI_DENSITY_QIC_24 0x05
- #define SCSI_DENSITY_HALFINCH_PE 0x06
- #define SCSI_DENSITY_QIC_120 0x0f
- #define SCSI_DENSITY_QIC_150 0x10
- #define SCSI_DENSITY_QIC_525_320 0x11
- #define SCSI_DENSITY_QIC_1320 0x12
- #define SCSI_DENSITY_QIC_2GB 0x22
- #define SCSI_DENSITY_QIC_4GB 0x26
- #define SCSI_DENSITY_QIC_3080 0x29

Enumerations

- enum `scsi_space_code` {
 SS_BLOCKS, SS_FILEMARKS, SS_SEQFILEMARKS, SS_EOD,
 SS_SETMARKS, SS_SEQSETMARKS }

Functions

- `__BEGIN_DECLS` void `scsi_read_block_limits` (struct `ccb_scsiio` *, u_int32_t, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t, struct `scsi_read_block_limits_data` *, u_int8_t, u_int32_t)
- void `scsi_sa_read_write` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, int readop, int sli, int fixed, u_int32_t length, u_int8_t *data_ptr, u_int32_t dxfer_len, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_rewind` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, int immediate, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_space` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, `scsi_space_code` code, u_int32_t count, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_load_unload` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, int immediate, int eot, int reten, int load, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_write_filemarks` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, int immediate, int setmark, u_int32_t num_marks, u_int8_t sense_len, u_int32_t timeout)

- void `scsi_reserve_release_unit` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, int third_party, int third_party_id, u_int8_t sense_len, u_int32_t timeout, int reserve)
- void `scsi_erase` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, int immediate, int long_erase, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_data_comp_page` (struct `scsi_data_compression_page` *page, u_int8_t dce, u_int8_t dde, u_int8_t red, u_int32_t comp_algorithm, u_int32_t decomp_algorithm)
- void `scsi_read_position` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, int hardsoft, struct `scsi_tape_position_data` *sbp, u_int8_t sense_len, u_int32_t timeout)
- void `scsi_set_position` (struct `ccb_scsiio` *csio, u_int32_t retries, void(*cbfcnp)(struct `cam_periph` *, union `ccb` *), u_int8_t tag_action, int hardsoft, u_int32_t blkno, u_int8_t sense_len, u_int32_t timeout)

7.36.1 Define Documentation

7.36.1.1 #define _SCSI_SCSI_SA_H 1

Definition at line 33 of file `scsi_sa.h`.

7.36.1.2 #define ERASE 0x19

Definition at line 272 of file `scsi_sa.h`.

Referenced by `scsi_erase()`.

7.36.1.3 #define LOAD_UNLOAD 0x1B

Definition at line 273 of file `scsi_sa.h`.

Referenced by `scsi_load_unload()`.

7.36.1.4 #define LOCATE 0x2B

Definition at line 274 of file `scsi_sa.h`.

Referenced by `scsi_set_position()`.

7.36.1.5 #define RBL_GRAN(rblim) ((rblim) → gran & RBL_GRAN_MASK)

Definition at line 49 of file `scsi_sa.h`.

Referenced by `samount()`.

7.36.1.6 #define RBL_GRAN_MASK 0x1F

Definition at line 48 of file `scsi_sa.h`.

7.36.1.7 #define READ_BLOCK_LIMITS 0x05

Definition at line 265 of file scsi_sa.h.

Referenced by scsi_read_block_limits().

7.36.1.8 #define READ_POSITION 0x34

Definition at line 275 of file scsi_sa.h.

Referenced by scsi_read_position().

7.36.1.9 #define RELEASE_UNIT 0x17

Definition at line 271 of file scsi_sa.h.

Referenced by scsi_reserve_release_unit().

7.36.1.10 #define RESERVE_UNIT 0x16

Definition at line 270 of file scsi_sa.h.

Referenced by scsi_reserve_release_unit().

7.36.1.11 #define REWIND 0x01

Definition at line 264 of file scsi_sa.h.

Referenced by scsi_rewind().

7.36.1.12 #define SA_ASOC_WP 0x04

Definition at line 194 of file scsi_sa.h.

7.36.1.13 #define SA_AVC 0x10

Definition at line 182 of file scsi_sa.h.

7.36.1.14 #define SA_BIS 0x40

Definition at line 180 of file scsi_sa.h.

7.36.1.15 #define SA_COMP_DEFAULT 0x01

Definition at line 191 of file scsi_sa.h.

7.36.1.16 #define SA_COMP_NONE 0x00

Definition at line 190 of file scsi_sa.h.

Referenced by sasetparams(), and sasetparams().

7.36.1.17 #define SA_DATA_COMPRESSION_PAGE 0x0f

Definition at line 163 of file scsi_sa.h.

Referenced by sasetparams(), and sasetparams().

7.36.1.18 #define SA_DBR 0x80

Definition at line 179 of file scsi_sa.h.

7.36.1.19 #define SA_DCP_DCC 0x40

Definition at line 205 of file scsi_sa.h.

Referenced by sasetparams(), and sasetparams().

7.36.1.20 #define SA_DCP_DCE 0x80

Definition at line 204 of file scsi_sa.h.

Referenced by sasetparams(), and sasetparams().

7.36.1.21 #define SA_DCP_DDE 0x80

Definition at line 207 of file scsi_sa.h.

Referenced by sasetparams().

7.36.1.22 #define SA_DCP_RED_0 0x00

Definition at line 210 of file scsi_sa.h.

7.36.1.23 #define SA_DCP_RED_1 0x20

Definition at line 211 of file scsi_sa.h.

7.36.1.24 #define SA_DCP_RED_2 0x40

Definition at line 212 of file scsi_sa.h.

7.36.1.25 #define SA_DCP_RED_MASK 0x60

Definition at line 208 of file scsi_sa.h.

7.36.1.26 #define SA_DCP_RED_SHAMT 5

Definition at line 209 of file scsi_sa.h.

7.36.1.27 #define SA_DEVICE_CONFIGURATION_PAGE 0x10

Definition at line 158 of file scsi_sa.h.

Referenced by sasetparams(), and sasetparams().

7.36.1.28 #define SA_MEDIUM_PARTITION_PAGE_1 0x11

Definition at line 159 of file scsi_sa.h.

7.36.1.29 #define SA_MEDIUM_PARTITION_PAGE_2 0x12

Definition at line 160 of file scsi_sa.h.

7.36.1.30 #define SA_MEDIUM_PARTITION_PAGE_3 0x13

Definition at line 161 of file scsi_sa.h.

7.36.1.31 #define SA_MEDIUM_PARTITION_PAGE_4 0x14

Definition at line 162 of file scsi_sa.h.

7.36.1.32 #define SA_PERM_WP 0x01

Definition at line 196 of file scsi_sa.h.

7.36.1.33 #define SA_PERS_WP 0x02

Definition at line 195 of file scsi_sa.h.

7.36.1.34 #define SA_RBO 0x20

Definition at line 184 of file scsi_sa.h.

7.36.1.35 #define SA_READ 0x08

Definition at line 266 of file scsi_sa.h.

Referenced by scsi_sa_read_write().

7.36.1.36 #define SA_REW 0x10

Definition at line 185 of file scsi_sa.h.

7.36.1.37 #define SA_RPOS_BCU 0x20

Definition at line 234 of file scsi_sa.h.

7.36.1.38 #define SA_RPOS_BOP 0x80

Definition at line 232 of file scsi_sa.h.

7.36.1.39 #define SA_RPOS_BPU 0x04

Definition at line 236 of file scsi_sa.h.

7.36.1.40 #define SA_RPOS_BYCU 0x10

Definition at line 235 of file scsi_sa.h.

7.36.1.41 #define SA_RPOS_EOP 0x40

Definition at line 233 of file scsi_sa.h.

7.36.1.42 #define SA_RPOS_PERR 0x02

Definition at line 237 of file scsi_sa.h.

7.36.1.43 #define SA_RPOS_UNCERTAIN SA_RPOS_BPU

Definition at line 238 of file scsi_sa.h.

Referenced by sardpos().

7.36.1.44 #define SA_RSMK 0x20

Definition at line 181 of file scsi_sa.h.

7.36.1.45 #define SA_SOCF_MASK 0xc0

Definition at line 183 of file scsi_sa.h.

7.36.1.46 #define SA_SPOS_BT 0x04

Definition at line 253 of file scsi_sa.h.

Referenced by scsi_set_position().

7.36.1.47 #define SA_SPOS_CP 0x02

Definition at line 252 of file scsi_sa.h.

7.36.1.48 #define SA_SPOS_IMMED 0x01

Definition at line 251 of file scsi_sa.h.

7.36.1.49 #define SA_WRITE 0x0A

Definition at line 267 of file scsi_sa.h.

Referenced by saerror(), and scsi_sa_read_write().

7.36.1.50 #define SAR_SLI 0x02

Definition at line 58 of file scsi_sa.h.

Referenced by scsi_sa_read_write().

7.36.1.51 #define SARW_FIXED 0x01

Definition at line 59 of file scsi_sa.h.

Referenced by scsi_sa_read_write().

7.36.1.52 #define SCSI_DENSITY_HALFINCH_1600 0x02

Definition at line 283 of file scsi_sa.h.

Referenced by samount().

7.36.1.53 #define SCSI_DENSITY_HALFINCH_6250 0x03

Definition at line 284 of file scsi_sa.h.

Referenced by samount().

7.36.1.54 #define SCSI_DENSITY_HALFINCH_6250C 0xC3

Definition at line 285 of file scsi_sa.h.

Referenced by samount().

7.36.1.55 #define SCSI_DENSITY_HALFINCH_800 0x01

Definition at line 282 of file scsi_sa.h.

Referenced by samount().

7.36.1.56 #define SCSI_DENSITY_HALFINCH_PE 0x06

Definition at line 289 of file scsi_sa.h.

Referenced by samount().

7.36.1.57 #define SCSI_DENSITY_QIC_11_4TRK 0x04

Definition at line 286 of file scsi_sa.h.

Referenced by samount().

7.36.1.58 #define SCSI_DENSITY_QIC_11_9TRK 0x84

Definition at line 287 of file scsi_sa.h.

Referenced by samount().

7.36.1.59 #define SCSI_DENSITY_QIC_120 0x0f

Definition at line 290 of file scsi_sa.h.

Referenced by samount().

7.36.1.60 #define SCSI_DENSITY_QIC_1320 0x12

Definition at line 293 of file scsi_sa.h.

Referenced by samount().

7.36.1.61 #define SCSI_DENSITY_QIC_150 0x10

Definition at line 291 of file scsi_sa.h.

Referenced by samount().

7.36.1.62 #define SCSI_DENSITY_QIC_24 0x05

Definition at line 288 of file scsi_sa.h.

Referenced by samount().

7.36.1.63 #define SCSI_DENSITY_QIC_2GB 0x22

Definition at line 294 of file scsi_sa.h.

Referenced by samount().

7.36.1.64 #define SCSI_DENSITY_QIC_3080 0x29

Definition at line 296 of file scsi_sa.h.

Referenced by samount().

7.36.1.65 #define SCSI_DENSITY_QIC_4GB 0x26

Definition at line 295 of file scsi_sa.h.

Referenced by samount().

7.36.1.66 #define SCSI_DENSITY_QIC_525_320 0x11

Definition at line 292 of file scsi_sa.h.

Referenced by samount().

7.36.1.67 #define SE_IMMED 0x2

Definition at line 139 of file scsi_sa.h.

Referenced by scsi_erase().

7.36.1.68 #define SE_LONG 0x1

Definition at line 138 of file scsi_sa.h.

Referenced by scsi_erase().

7.36.1.69 #define SE_LUN_MASK 0xE0

Definition at line 137 of file scsi_sa.h.

7.36.1.70 #define SLU_EOT 0x04

Definition at line 71 of file scsi_sa.h.

Referenced by scsi_load_unload().

7.36.1.71 #define SLU_IMMED 0x01

Definition at line 68 of file scsi_sa.h.

Referenced by scsi_load_unload().

7.36.1.72 #define SLU_LOAD 0x01

Definition at line 73 of file scsi_sa.h.

Referenced by scsi_load_unload().

7.36.1.73 #define SLU_RETEN 0x02

Definition at line 72 of file scsi_sa.h.

Referenced by scsi_load_unload().

7.36.1.74 #define SMH_SA_BUF_MODE_MASK 0x70

Definition at line 148 of file scsi_sa.h.

Referenced by sasetparams().

7.36.1.75 #define SMH_SA_BUF_MODE_MIBUF 0x20

Definition at line 151 of file scsi_sa.h.

7.36.1.76 #define SMH_SA_BUF_MODE_NOBUF 0x00

Definition at line 149 of file scsi_sa.h.

7.36.1.77 #define SMH_SA_BUF_MODE_SIBUF 0x10

Definition at line 150 of file scsi_sa.h.

Referenced by sasetparams().

7.36.1.78 #define SMH_SA_SPEED_DEFAULT 0x00

Definition at line 153 of file scsi_sa.h.

7.36.1.79 #define SMH_SA_SPEED_MASK 0x0F

Definition at line 152 of file scsi_sa.h.

Referenced by sagetparams().

7.36.1.80 #define SMH_SA_WP 0x80

Definition at line 147 of file scsi_sa.h.

Referenced by sagetparams().

7.36.1.81 #define SPACE 0x11

Definition at line 269 of file scsi_sa.h.

Referenced by scsi_space().

7.36.1.82 #define SREW_IMMED 0x01

Definition at line 99 of file scsi_sa.h.

7.36.1.83 #define SREW_IMMED 0x01

Definition at line 99 of file scsi_sa.h.

Referenced by scsi_rewind().

7.36.1.84 #define SRRU_3RD_MASK 0xE

Definition at line 125 of file scsi_sa.h.

Referenced by scsi_reserve_release_unit().

7.36.1.85 #define SRRU_3RD_PARTY 0x10

Definition at line 123 of file scsi_sa.h.

Referenced by scsi_reserve_release_unit().

7.36.1.86 #define SRRU_3RD_SHAMT 1

Definition at line 124 of file scsi_sa.h.

Referenced by scsi_reserve_release_unit().

7.36.1.87 #define SRRU_LUN_MASK 0xE0

Definition at line 122 of file scsi_sa.h.

7.36.1.88 #define SWFMRK_IMMED 0x01

Definition at line 108 of file scsi_sa.h.

Referenced by scsi_write_filemarks().

7.36.1.89 #define SWFMRK_WSMK 0x02

Definition at line 109 of file scsi_sa.h.

Referenced by scsi_write_filemarks().

7.36.1.90 #define WRITE_FILEMARKS 0x10

Definition at line 268 of file scsi_sa.h.

Referenced by scsi_write_filemarks().

7.36.2 Enumeration Type Documentation**7.36.2.1 enum [scsi_space_code](#)**

Enumerator:

SS_BLOCKS

SS_FILEMARKS

SS_SEQFILEMARKS

SS_EOD

SS_SETMARKS

SS_SEQSETMARKS

Definition at line 86 of file scsi_sa.h.

7.36.3 Function Documentation

7.36.3.1 void `scsi_data_comp_page` (struct `scsi_data_compression_page` * *page*, `u_int8_t` *dce*, `u_int8_t` *dde*, `u_int8_t` *red*, `u_int32_t` *comp_algorithm*, `u_int32_t` *decomp_algorithm*)

7.36.3.2 void `scsi_erase` (struct `ccb_scsiio` * *csio*, `u_int32_t` *retries*, void(*) (struct `cam_periph` *, union `ccb` *) *cbfnp*, `u_int8_t` *tag_action*, int *immediate*, int *long_erase*, `u_int8_t` *sense_len*, `u_int32_t` *timeout*)

Definition at line 3618 of file `scsi_sa.c`.

References `CAM_DIR_NONE`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `ERASE`, `SE_IMMED`, and `SE_LONG`.

Referenced by `saerase()`.

Here is the call graph for this function:



7.36.3.3 void `scsi_load_unload` (struct `ccb_scsiio` * *csio*, `u_int32_t` *retries*, void(*) (struct `cam_periph` *, union `ccb` *) *cbfnp*, `u_int8_t` *tag_action*, int *immediate*, int *eot*, int *reten*, int *load*, `u_int8_t` *sense_len*, `u_int32_t` *timeout*)

Definition at line 3504 of file `scsi_sa.c`.

References `CAM_DIR_NONE`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `LOAD_UNLOAD`, `SLU_EOT`, `SLU_IMMED`, `SLU_LOAD`, and `SLU_RETEN`.

Referenced by `saloadunload()`, `samount()`, and `saretension()`.

Here is the call graph for this function:



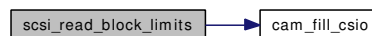
7.36.3.4 `__BEGIN_DECLS` void `scsi_read_block_limits` (struct `ccb_scsiio` *, `u_int32_t`, void(*) (struct `cam_periph` *, union `ccb` *) *cbfnp*, `u_int8_t`, struct `scsi_read_block_limits_data` *, `u_int8_t`, `u_int32_t`)

Definition at line 3462 of file `scsi_sa.c`.

References `CAM_DIR_IN`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, and `READ_BLOCK_LIMITS`.

Referenced by `samount()`.

Here is the call graph for this function:



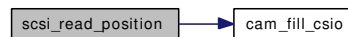
7.36.3.5 void `scsi_read_position` (struct `ccb_scsiio` * *csio*, u_int32_t *retries*, void(*) (struct `cam_periph` *, union `ccb` *) *cbfcn*, u_int8_t *tag_action*, int *hardsoft*, struct `scsi_tape_position_data` * *sbp*, u_int8_t *sense_len*, u_int32_t *timeout*)

Definition at line 3644 of file `scsi_sa.c`.

References `CAM_DIR_IN`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, and `READ_POSITION`.

Referenced by `sardpos()`.

Here is the call graph for this function:



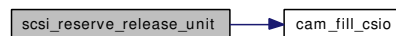
7.36.3.6 void `scsi_reserve_release_unit` (struct `ccb_scsiio` * *csio*, u_int32_t *retries*, void(*) (struct `cam_periph` *, union `ccb` *) *cbfcn*, u_int8_t *tag_action*, int *third_party*, int *third_party_id*, u_int8_t *sense_len*, u_int32_t *timeout*, int *reserve*)

Definition at line 3591 of file `scsi_sa.c`.

References `CAM_DIR_NONE`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `RELEASE_UNIT`, `RESERVE_UNIT`, `SRRU_3RD_MASK`, `SRRU_3RD_PARTY`, and `SRRU_3RD_SHAMT`.

Referenced by `sareservereleaseunit()`.

Here is the call graph for this function:



7.36.3.7 void `scsi_rewind` (struct `ccb_scsiio` * *csio*, u_int32_t *retries*, void(*) (struct `cam_periph` *, union `ccb` *) *cbfcn*, u_int8_t *tag_action*, int *immediate*, u_int8_t *sense_len*, u_int32_t *timeout*)

Definition at line 3529 of file `scsi_sa.c`.

References `CAM_DIR_NONE`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `REWIND`, and `SREW_IMMED`.

Referenced by `samount()`, and `sarewind()`.

Here is the call graph for this function:



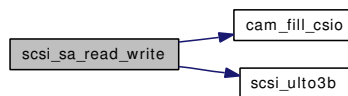
7.36.3.8 `void scsi_sa_read_write (struct ccb_scsiio * csio, u_int32_t retries, void(*)(struct cam_periph *, union ccb *) cbfcnp, u_int8_t tag_action, int readop, int sli, int fixed, u_int32_t length, u_int8_t * data_ptr, u_int32_t dxfer_len, u_int8_t sense_len, u_int32_t timeout)`

Definition at line 3480 of file `scsi_sa.c`.

References `CAM_DIR_IN`, `CAM_DIR_OUT`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `scsi_sa_rw::control`, `scsi_sa_rw::length`, `scsi_sa_rw::opcode`, `SA_READ`, `SA_WRITE`, `SAR_SLI`, `SARW_FIXED`, `scsi_ulto3b()`, and `scsi_sa_rw::sli_fixed`.

Referenced by `samount()`, and `sastart()`.

Here is the call graph for this function:



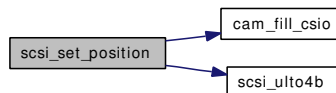
7.36.3.9 `void scsi_set_position (struct ccb_scsiio * csio, u_int32_t retries, void(*)(struct cam_periph *, union ccb *) cbfcnp, u_int8_t tag_action, int hardsoft, u_int32_t blkno, u_int8_t sense_len, u_int32_t timeout)`

Definition at line 3664 of file `scsi_sa.c`.

References `CAM_DIR_NONE`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `LOCATE`, `SA_SPOS_BT`, and `scsi_ulto4b()`.

Referenced by `sasetpos()`.

Here is the call graph for this function:



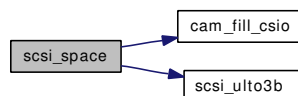
7.36.3.10 `void scsi_space (struct ccb_scsiio * csio, u_int32_t retries, void(*)(struct cam_periph *, union ccb *) cbfcnp, u_int8_t tag_action, scsi_space_code code, u_int32_t count, u_int8_t sense_len, u_int32_t timeout)`

Definition at line 3547 of file `scsi_sa.c`.

References `CAM_DIR_NONE`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `scsi_space::code`, `scsi_space::control`, `scsi_space::count`, `scsi_space::opcode`, `scsi_ulto3b()`, and `SPACE`.

Referenced by `saspace()`.

Here is the call graph for this function:



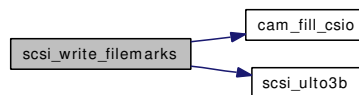
7.36.3.11 void **scsi_write_filemarks** (struct **ccb_scsiio** * *csio*, u_int32_t *retries*, void(*) (struct **cam_periph** *, union **ccb** *) *cbfcnp*, u_int8_t *tag_action*, int *immediate*, int *setmark*, u_int32_t *num_marks*, u_int8_t *sense_len*, u_int32_t *timeout*)

Definition at line 3565 of file *scsi_sa.c*.

References `CAM_DIR_NONE`, `cam_fill_csio()`, `cdb_t::cdb_bytes`, `ccb_scsiio::cdb_io`, `scsi_ulto3b()`, `SWFMRK_IMMED`, `SWFMRK_WSMK`, and `WRITE_FILEMARKS`.

Referenced by `sawwritefilemarks()`.

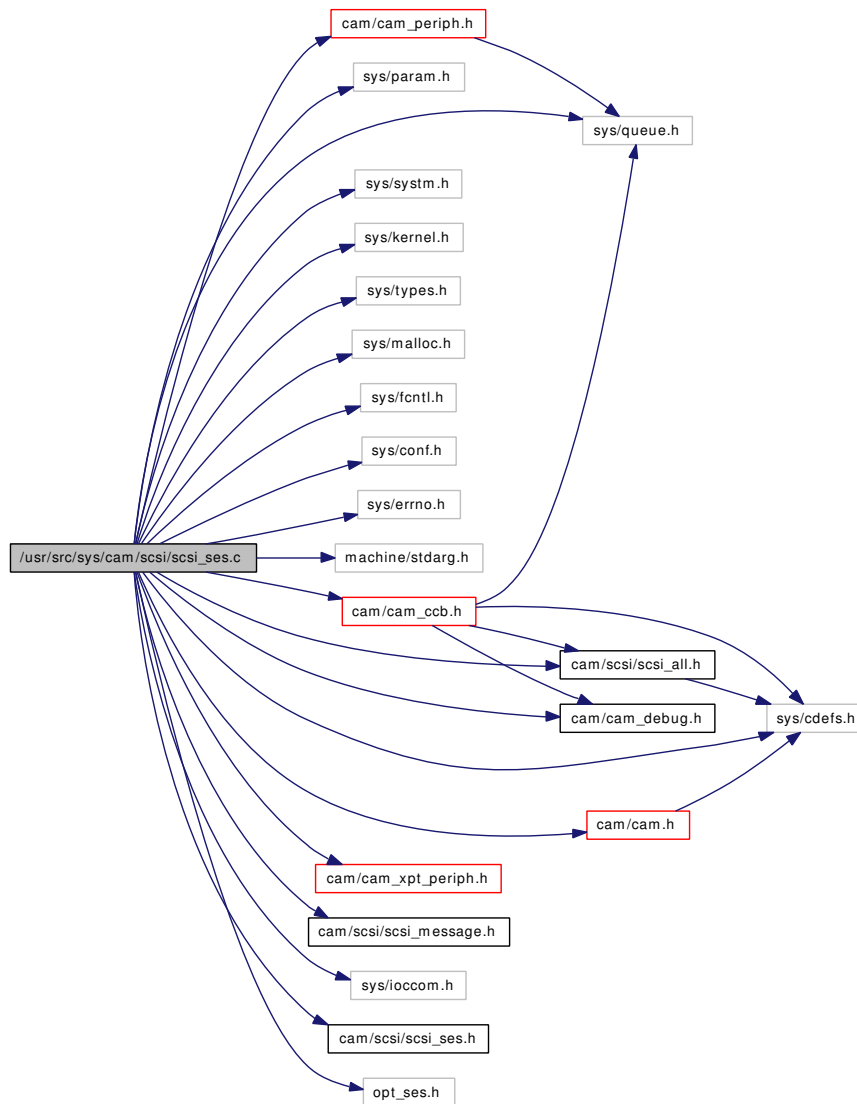
Here is the call graph for this function:



7.37 /usr/src/sys/cam/scsi/scsi_ses.c File Reference

```
#include <sys/cdefs.h>
#include <sys/param.h>
#include <sys/queue.h>
#include <sys/system.h>
#include <sys/kernel.h>
#include <sys/types.h>
#include <sys/malloc.h>
#include <sys/fcntl.h>
#include <sys/conf.h>
#include <sys/errno.h>
#include <machine/stdarg.h>
#include <cam/cam.h>
#include <cam/cam_ccb.h>
#include <cam/cam_periph.h>
#include <cam/cam_xpt_periph.h>
#include <cam/cam_debug.h>
#include <cam/scsi/scsi_all.h>
#include <cam/scsi/scsi_message.h>
#include <sys/ioccom.h>
#include <cam/scsi/scsi_ses.h>
#include <opt_ses.h>
```

Include dependency graph for scsi_ses.c:



Data Structures

- struct [encvec](#)
- struct [encobj](#)
- struct [ses_softc](#)
- struct [SesCfgHdr](#)
- struct [SesEncHdr](#)
- struct [SesEncDesc](#)
- struct [SesThdr](#)
- struct [SesComStat](#)
- struct [typidx](#)
- struct [sscfg](#)
- struct [scfg](#)

Defines

- #define ENCI_SVALID 0x80
- #define SEN_ID "UNISYS SUN_SEN"
- #define SEN_ID_LEN 24
- #define STRNCMP strcmp
- #define PRINTF printf
- #define SES_LOG ses_log
- #define SES_DLOG if (0) ses_log
- #define SES_VLOG if (bootverbose) ses_log
- #define SES_MALLOC(amt) malloc(amt, M_SCSISES, M_NOWAIT)
- #define SES_FREE(ptr, amt) free(ptr, M_SCSISES)
- #define MEMZERO bzero
- #define MEMCPY(dest, src, amt) bcopy(src, dest, amt)
- #define ccb_state ppriv_field0
- #define ccb_bp ppriv_ptr1
- #define SES_FLAG_INVALID 0x01
- #define SES_FLAG_OPEN 0x02
- #define SES_FLAG_INITIALIZED 0x04
- #define SESUNIT(x) (minor((x)))
- #define SES_CFLAGS CAM_RETRY_SELTO
- #define SES_FLAGS SF_NO_PRINT | SF_RETRY_UA
- #define SAFTE_START 44
- #define SAFTE_END 50
- #define SAFTE_LEN SAFTE_END-SAFTE_START
- #define SesStatusPage SesControlPage
- #define SesStringIn SesStringOut
- #define SesThresholdIn SesThresholdOut
- #define SesArrayStatus SesArrayControl
- #define SES_CFGHDR_MINLEN 12
- #define SES_ENCHDR_MINLEN 48
- #define SES_ENCHDR_VMIN 36
- #define sbyte(x, byte) (((uint32_t)(x)) >> (byte * 8)) & 0xff
- #define sbit(x, bit) (((uint32_t)(x)) << bit)
- #define sset8(outp, idx, sval) (((uint8_t *) (outp))[idx++]) = sbyte(sval, 0)
- #define sset16(outp, idx, sval)
- #define sset24(outp, idx, sval)
- #define sset32(outp, idx, sval)
- #define gbyte(x, byte) (((uint32_t)(x)) & 0xff) << (byte * 8)
- #define gbit(lv, in, idx, shft, mask) lv = ((in[idx] >> shft) & mask)
- #define sget8(inp, idx, lval) lval = (((uint8_t *) (inp))[idx++])
- #define gget8(inp, idx, lval) lval = (((uint8_t *) (inp))[idx])
- #define sget16(inp, idx, lval)
- #define gget16(inp, idx, lval)
- #define sget24(inp, idx, lval)
- #define gget24(inp, idx, lval)
- #define sget32(inp, idx, lval)
- #define gget32(inp, idx, lval)
- #define SCSZ 0x2000
- #define CFLEN (256 + SES_ENCHDR_MINLEN)

- #define `ALL_ENC_STAT`
- #define `SAFTE_RD_RDCFG` 0x00
- #define `SAFTE_RD_RDESTS` 0x01
- #define `SAFTE_RD_RDDSTS` 0x04
- #define `SAFTE_WT_DSTAT` 0x10
- #define `SAFTE_WT_SLTOP` 0x12
- #define `SAFTE_WT_FANSPD` 0x13
- #define `SAFTE_WT_ACTPWS` 0x14
- #define `SAFTE_WT_GLOBAL` 0x15
- #define `SAFT_SCRATCH` 64
- #define `NPSEUDO_THERM` 16
- #define `NPSEUDO_ALARM` 1
- #define `SAFT_ALARM_OFFSET`(cc) (cc) → slotoff - 1
- #define `SAFT_FLG1_ALARM` 0x1
- #define `SAFT_FLG1_GLOBFAIL` 0x2
- #define `SAFT_FLG1_GLOBWARN` 0x4
- #define `SAFT_FLG1_ENCPWROFF` 0x8
- #define `SAFT_FLG1_ENCFANFAIL` 0x10
- #define `SAFT_FLG1_ENCPWRFAIL` 0x20
- #define `SAFT_FLG1_ENCDRVFAIL` 0x40
- #define `SAFT_FLG1_ENCDRVWARN` 0x80
- #define `SAFT_FLG2_LOCKDOOR` 0x4
- #define `SAFT_PRIVATE` sizeof (struct `scfg`)
- #define `SAFT_BAIL`(r, x, k, l)

Typedefs

- typedef `ses_softc` `ses_softc_t`

Enumerations

- enum `enctyp` {
`SES_NONE`, `SES_SES_SCSI2`, `SES_SES`, `SES_SES_PASSTHROUGH`,
`SES_SEN`, `SES_SAFT` }
- enum `SesDiagPageCodes` {
`SesConfigPage` = 0x1, `SesControlPage`, `SesHelpTxt`, `SesStringOut`,
`SesThresholdOut`, `SesArrayControl`, `SesElementDescriptor`, `SesShortStatus` }

Functions

- `__FBSDID` ("\$FreeBSD: src/sys/cam/scsi/scsi_ses.c,v 1.33 2006/12/05 07:45:28 mjacob Exp \$")
- `MALLOC_DEFINE` (M_SCSISES, "SCSI SES", "SCSI SES buffers")
- static `enctyp ses_type` (void *, int)
- static int `ses_softc_init` (`ses_softc_t` *, int)
- static int `ses_init_enc` (`ses_softc_t` *)
- static int `ses_get_encstat` (`ses_softc_t` *, int)
- static int `ses_set_encstat` (`ses_softc_t` *, `uint8_t`, int)
- static int `ses_get_objstat` (`ses_softc_t` *, `ses_objstat` *, int)

- static int `ses_set_objstat` (`ses_softc_t *`, `ses_objstat *`, int)
- static int `saftc_softc_init` (`ses_softc_t *`, int)
- static int `saftc_init_enc` (`ses_softc_t *`)
- static int `saftc_get_encstat` (`ses_softc_t *`, int)
- static int `saftc_set_encstat` (`ses_softc_t *`, `uint8_t`, int)
- static int `saftc_get_objstat` (`ses_softc_t *`, `ses_objstat *`, int)
- static int `saftc_set_objstat` (`ses_softc_t *`, `ses_objstat *`, int)
- static int `ses_runcmd` (`struct ses_softc *`, `char *`, int, `char *`, int)
- static void `ses_log` (`struct ses_softc *`, `const char *`,...)
- static void `sesasync` (`void *`, `u_int32_t`, `struct cam_path *`, `void *`)
- static void `sesdone` (`struct cam_periph *`, `union ccb *`)
- static int `seserror` (`union ccb *`, `u_int32_t`, `u_int32_t`)
- `PERIPHDRIIVER_DECLARE` (`ses`, `sesdriver`)
- static void `sesinit` (`void`)
- static void `sesoninvalidate` (`struct cam_periph *`periph)
- static void `sescleanup` (`struct cam_periph *`periph)
- static `cam_status` `sesregister` (`struct cam_periph *`periph, `void *`arg)
- static int `sesopen` (`struct cdev *`dev, int flags, int fmt, `struct thread *`td)
- static int `sesclose` (`struct cdev *`dev, int flag, int fmt, `struct thread *`td)
- static void `sesstart` (`struct cam_periph *`p, `union ccb *`sccb)
- static int `sesioctl` (`struct cdev *`dev, `u_long` cmd, `caddr_t` arg_addr, int flag, `struct thread *`td)
- static int `ses_getconfig` (`ses_softc_t *`)
- static int `ses_getputstat` (`ses_softc_t *`, int, `SesComStat *`, int, int)
- static int `ses_cfghdr` (`uint8_t *`, int, `SesCfgHdr *`)
- static int `ses_enchr` (`uint8_t *`, int, `uint8_t`, `SesEncHdr *`)
- static int `ses_endedesc` (`uint8_t *`, int, `uint8_t`, `SesEncDesc *`)
- static int `ses_getthdr` (`uint8_t *`, int, int, `SesThdr *`)
- static int `ses_decode` (`char *`, int, `uint8_t *`, int, int, `SesComStat *`)
- static int `ses_encode` (`char *`, int, `uint8_t *`, int, int, `SesComStat *`)
- static int `saftc_getconfig` (`ses_softc_t *`)
- static int `saftc_rdstat` (`ses_softc_t *`, int)
- static int `set_objstat_sel` (`ses_softc_t *`, `ses_objstat *`, int)
- static int `wrbuf16` (`ses_softc_t *`, `uint8_t`, `uint8_t`, `uint8_t`, `uint8_t`, int)
- static void `wrslot_stat` (`ses_softc_t *`, int)
- static int `perf_slotop` (`ses_softc_t *`, `uint8_t`, `uint8_t`, int)

Variables

- static `d_open_t` `sesopen`
- static `d_close_t` `sesclose`
- static `d_ioctl_t` `sesioctl`
- static `periph_init_t` `sesinit`
- static `periph_ctor_t` `sesregister`
- static `periph_oninv_t` `sesoninvalidate`
- static `periph_dtor_t` `sescleanup`
- static `periph_start_t` `sesstart`
- static `struct periph_driver` `sesdriver`
- static `struct cdevsw` `ses_cdevsw`
- static `char *` `saftc_2little` = "Too Little Data Returned (%d) at line %d\n"

7.37.1 Define Documentation

7.37.1.1 #define ALL_ENC_STAT

Value:

```
(SES_ENCSTAT_CRITICAL | SES_ENCSTAT_UNRECOV | \
    SES_ENCSTAT_NONCRITICAL | SES_ENCSTAT_INFO)
```

Definition at line 1566 of file scsi_ses.c.

Referenced by safte_rdstat(), and safte_set_encstat().

7.37.1.2 #define ccb_bp ppriv_ptr1

Definition at line 139 of file scsi_ses.c.

7.37.1.3 #define ccb_state ppriv_field0

Definition at line 138 of file scsi_ses.c.

7.37.1.4 #define CFLEN (256 + SES_ENCHDR_MINLEN)

Definition at line 935 of file scsi_ses.c.

Referenced by ses_getconfig().

7.37.1.5 #define ENCL_SVALID 0x80

Definition at line 79 of file scsi_ses.c.

Referenced by safte_get_objstat(), safte_set_encstat(), ses_get_encstat(), and sesioctl().

7.37.1.6 #define gbit(lv, in, idx, shft, mask) lv = ((in[idx] >> shft) & mask)

Definition at line 900 of file scsi_ses.c.

7.37.1.7 #define gbyte(x, byte) (((uint32_t)(x)) & 0xff) << (byte * 8)

Definition at line 899 of file scsi_ses.c.

7.37.1.8 #define gget16(inp, idx, lval)

Value:

```
lval = gbyte(((uint8_t *) (inp))[idx], 1) | \
    (((uint8_t *) (inp))[idx+1])
```

Definition at line 908 of file scsi_ses.c.

7.37.1.9 #define gget24(inp, idx, lval)**Value:**

```
lval = gbyte((((uint8_t *) (inp))[idx]), 2) | \  
          gbyte((((uint8_t *) (inp))[idx+1]), 1) | \  
          (((uint8_t *) (inp))[idx+2])
```

Definition at line 917 of file scsi_ses.c.

7.37.1.10 #define gget32(inp, idx, lval)**Value:**

```
lval = gbyte((((uint8_t *) (inp))[idx]), 3) | \  
          gbyte((((uint8_t *) (inp))[idx+1]), 2) | \  
          gbyte((((uint8_t *) (inp))[idx+2]), 1) | \  
          (((uint8_t *) (inp))[idx+3])
```

Definition at line 928 of file scsi_ses.c.

Referenced by ses_cfghdr().

7.37.1.11 #define gget8(inp, idx, lval) lval = (((uint8_t *) (inp))[idx])

Definition at line 902 of file scsi_ses.c.

Referenced by ses_cfghdr(), ses_decode(), ses_enddesc(), ses_enchr(), and ses_getthdr().

7.37.1.12 #define MEMCPY(dest, src, amt) bcopy(src, dest, amt)

Definition at line 128 of file scsi_ses.c.

Referenced by ses_enddesc().

7.37.1.13 #define MEMZERO bzero

Definition at line 127 of file scsi_ses.c.

Referenced by perf_slotop(), safte_softc_init(), ses_getconfig(), wrbuf16(), and wrslot_stat().

7.37.1.14 #define NPSEUDO_ALARM 1

Definition at line 1591 of file scsi_ses.c.

Referenced by safte_getconfig(), and safte_softc_init().

7.37.1.15 #define NPSEUDO_THERM 16

Definition at line 1590 of file scsi_ses.c.

Referenced by safte_softc_init().

7.37.1.16 #define PRINTF printf

Definition at line 117 of file scsi_ses.c.

Referenced by ses_decode(), and ses_encode().

7.37.1.17 #define SAFT_ALARM_OFFSET(cc) (cc) → slotoff - 1

Definition at line 1613 of file scsi_ses.c.

7.37.1.18 #define SAFT_BAIL(r, x, k, l)**Value:**

```
if ((r) >= (x)) { \
    SES_LOG(ssc, safte_2little, x, __LINE__); \
    SES_FREE((k), (l)); \
    return (EIO); \
}
```

Definition at line 1629 of file scsi_ses.c.

Referenced by safte_rdstat().

7.37.1.19 #define SAFT_FLG1_ALARM 0x1

Definition at line 1616 of file scsi_ses.c.

Referenced by safte_set_encstat(), safte_set_objstat(), and set_objstat_sel().

7.37.1.20 #define SAFT_FLG1_ENCDRVFAIL 0x40

Definition at line 1622 of file scsi_ses.c.

7.37.1.21 #define SAFT_FLG1_ENCDRVWARN 0x80

Definition at line 1623 of file scsi_ses.c.

7.37.1.22 #define SAFT_FLG1_ENCFANFAIL 0x10

Definition at line 1620 of file scsi_ses.c.

Referenced by safte_set_objstat().

7.37.1.23 #define SAFT_FLG1_ENCPWRFAIL 0x20

Definition at line 1621 of file scsi_ses.c.

Referenced by safte_set_objstat().

7.37.1.24 #define SAFT_FLG1_ENCPWROFF 0x8

Definition at line 1619 of file scsi_ses.c.

7.37.1.25 #define SAFT_FLG1_GLOBFAIL 0x2

Definition at line 1617 of file scsi_ses.c.

Referenced by safte_set_encstat().

7.37.1.26 #define SAFT_FLG1_GLOBWARN 0x4

Definition at line 1618 of file scsi_ses.c.

Referenced by safte_set_encstat().

7.37.1.27 #define SAFT_FLG2_LOCKDOOR 0x4

Definition at line 1625 of file scsi_ses.c.

Referenced by safte_set_objstat(), and set_objstat_sel().

7.37.1.28 #define SAFT_PRIVATE sizeof (struct [scfg](#))

Definition at line 1626 of file scsi_ses.c.

Referenced by safte_softc_init().

7.37.1.29 #define SAFT_SCRATCH 64

Definition at line 1589 of file scsi_ses.c.

Referenced by perf_slotop(), and safte_getconfig().

7.37.1.30 #define SAFTE_END 50

Definition at line 725 of file scsi_ses.c.

Referenced by ses_type().

7.37.1.31 #define SAFTE_LEN SAFTE_END-SAFTE_START

Definition at line 726 of file scsi_ses.c.

Referenced by ses_type().

7.37.1.32 #define SAFTE_RD_RD CFG 0x00

Definition at line 1575 of file scsi_ses.c.

Referenced by safte_getconfig().

7.37.1.33 #define SAFTE_RD_RDDSTS 0x04

Definition at line 1577 of file scsi_ses.c.

7.37.1.34 #define SAFTE_RD_RDESTS 0x01

Definition at line 1576 of file scsi_ses.c.

Referenced by safte_rdstat().

7.37.1.35 #define SAFTE_START 44

Definition at line 724 of file scsi_ses.c.

Referenced by ses_type().

7.37.1.36 #define SAFTE_WT_ACTPWS 0x14

Definition at line 1585 of file scsi_ses.c.

Referenced by safte_set_objstat(), and set_objstat_sel().

7.37.1.37 #define SAFTE_WT_DSTAT 0x10

Definition at line 1582 of file scsi_ses.c.

Referenced by wrslot_stat().

7.37.1.38 #define SAFTE_WT_FANSPD 0x13

Definition at line 1584 of file scsi_ses.c.

Referenced by safte_set_objstat(), and set_objstat_sel().

7.37.1.39 #define SAFTE_WT_GLOBAL 0x15

Definition at line 1586 of file scsi_ses.c.

Referenced by safte_init_enc(), safte_set_encstat(), safte_set_objstat(), and set_objstat_sel().

7.37.1.40 #define SAFTE_WT_SLTOP 0x12

Definition at line 1583 of file scsi_ses.c.

Referenced by perf_slotop().

7.37.1.41 #define sbit(x, bit) (((uint32_t)(x)) << bit)

Definition at line 879 of file scsi_ses.c.

7.37.1.42 #define sbyte(x, byte) (((uint32_t)(x)) >> (byte * 8)) & 0xff)

Definition at line 878 of file scsi_ses.c.

7.37.1.43 #define SCSZ 0x2000

Definition at line 934 of file scsi_ses.c.

Referenced by ses_getconfig().

7.37.1.44 #define SEN_ID "UNISYS SUN_SEN"

Definition at line 90 of file scsi_ses.c.

Referenced by ses_type().

7.37.1.45 #define SEN_ID_LEN 24

Definition at line 91 of file scsi_ses.c.

Referenced by ses_type().

7.37.1.46 #define SES_CFGHDR_MINLEN 12

Definition at line 802 of file scsi_ses.c.

Referenced by ses_cfghdr(), and ses_getthdr().

7.37.1.47 #define SES_CFLAGS CAM_RETRY_SELTO

Definition at line 653 of file scsi_ses.c.

Referenced by ses_runcmd().

7.37.1.48 #define SES_DLOG if (0) ses_log

Definition at line 122 of file scsi_ses.c.

Referenced by perf_slotop(), safte_set_objstat(), wrbuf16(), and wrslot_stat().

7.37.1.49 #define SES_ENCHDR_MINLEN 48

Definition at line 808 of file scsi_ses.c.

Referenced by ses_getconfig().

7.37.1.50 #define SES_ENCHDR_VMIN 36

Definition at line 814 of file scsi_ses.c.

7.37.1.51 #define SES_FLAG_INITIALIZED 0x04

Definition at line 155 of file scsi_ses.c.

Referenced by sesioctl(), and sesopen().

7.37.1.52 #define SES_FLAG_INVALID 0x01

Definition at line 153 of file scsi_ses.c.

Referenced by sesoninvalidate(), and sesopen().

7.37.1.53 #define SES_FLAG_OPEN 0x02

Definition at line 154 of file scsi_ses.c.

Referenced by sesclose(), and sesopen().

7.37.1.54 #define SES_FLAGS SF_NO_PRINT | SF_RETRY_UA

Definition at line 654 of file scsi_ses.c.

Referenced by ses_runcmd().

7.37.1.55 #define SES_FREE(ptr, amt) free(ptr, M_SCSES)

Definition at line 126 of file scsi_ses.c.

Referenced by perf_slotop(), safte_getconfig(), safte_rdstat(), safte_softc_init(), ses_getconfig(), ses_getputstat(), ses_softc_init(), wrbuf16(), and wrslot_stat().

7.37.1.56 #define SES_LOG ses_log

Definition at line 118 of file scsi_ses.c.

Referenced by safte_getconfig(), safte_rdstat(), and ses_getconfig().

7.37.1.57 #define SES_MALLOC(amt) malloc(amt, M_SCSES, M_NOWAIT)

Definition at line 125 of file scsi_ses.c.

Referenced by perf_slotop(), safte_getconfig(), safte_rdstat(), safte_softc_init(), ses_getconfig(), ses_getputstat(), ses_softc_init(), wrbuf16(), and wrslot_stat().

7.37.1.58 #define SES_VLOG if (bootverbose) ses_log

Definition at line 124 of file scsi_ses.c.

Referenced by safte_getconfig(), and ses_getconfig().

7.37.1.59 #define SesArrayStatus SesArrayControl**7.37.1.60 #define SesStatusPage SesControlPage**

Referenced by ses_getputstat().

7.37.1.61 #define SesStringIn SesStringOut**7.37.1.62 #define SesThresholdIn SesThresholdOut****7.37.1.63 #define SESUNIT(x) (minor((x)))**

Definition at line 157 of file scsi_ses.c.

7.37.1.64 #define sget16(inp, idx, lval)**Value:**

```
lval = gbyte((((uint8_t *) (inp))[idx]), 1) | \
          (((uint8_t *) (inp))[idx+1]), idx += 2
```

Definition at line 904 of file scsi_ses.c.

7.37.1.65 #define sget24(inp, idx, lval)**Value:**

```
lval = gbyte((((uint8_t *) (inp))[idx]), 2) | \
          gbyte((((uint8_t *) (inp))[idx+1]), 1) | \
          (((uint8_t *) (inp))[idx+2]), idx += 3
```

Definition at line 912 of file scsi_ses.c.

7.37.1.66 #define sget32(inp, idx, lval)**Value:**

```
lval = gbyte((((uint8_t *) (inp))[idx]), 3) | \
          gbyte((((uint8_t *) (inp))[idx+1]), 2) | \
          gbyte((((uint8_t *) (inp))[idx+2]), 1) | \
          (((uint8_t *) (inp))[idx+3]), idx += 4
```

Definition at line 922 of file scsi_ses.c.

7.37.1.67 #define sget8(inp, idx, lval) lval = (((uint8_t *) (inp))[idx++])

Definition at line 901 of file scsi_ses.c.

7.37.1.68 #define sset16(outp, idx, sval)**Value:**

```
((uint8_t *) (outp))[idx++] = sbyte(sval, 1), \  
((uint8_t *) (outp))[idx++] = sbyte(sval, 0)
```

Definition at line 882 of file scsi_ses.c.

7.37.1.69 #define sset24(outp, idx, sval)**Value:**

```
((uint8_t *) (outp))[idx++] = sbyte(sval, 2), \  
((uint8_t *) (outp))[idx++] = sbyte(sval, 1), \  
((uint8_t *) (outp))[idx++] = sbyte(sval, 0)
```

Definition at line 887 of file scsi_ses.c.

7.37.1.70 #define sset32(outp, idx, sval)**Value:**

```
((uint8_t *) (outp))[idx++] = sbyte(sval, 3), \  
((uint8_t *) (outp))[idx++] = sbyte(sval, 2), \  
((uint8_t *) (outp))[idx++] = sbyte(sval, 1), \  
((uint8_t *) (outp))[idx++] = sbyte(sval, 0)
```

Definition at line 893 of file scsi_ses.c.

7.37.1.71 #define sset8(outp, idx, sval) (((uint8_t *) (outp))[idx++]) = sbyte(sval, 0)

Definition at line 880 of file scsi_ses.c.

Referenced by ses_encode().

7.37.1.72 #define STRNCMP strncmp

Definition at line 116 of file scsi_ses.c.

Referenced by ses_type().

7.37.2 Typedef Documentation**7.37.2.1 typedef struct [ses_softc](#) [ses_softc_t](#)**

Definition at line 69 of file scsi_ses.c.

7.37.3 Enumeration Type Documentation

7.37.3.1 enum `enctyp`

Enumerator:

SES_NONE
SES_SES_SCSI2
SES_SES
SES_SES_PASSTHROUGH
SES_SEN
SES_SAFT

Definition at line 59 of file `scsi_ses.c`.

7.37.3.2 enum `SesDiagPageCodes`

Enumerator:

SesConfigPage
SesControlPage
SesHelpTxt
SesStringOut
SesThresholdOut
SesArrayControl
SesElementDescriptor
SesShortStatus

Definition at line 779 of file `scsi_ses.c`.

7.37.4 Function Documentation

7.37.4.1 `__FBSDID` ("FreeBSD: src/sys/cam/scsi/scsi_ses. c, v 1.33 2006/12/05 07:45:28 mjacob Exp \$")

7.37.4.2 `MALLOC_DEFINE` (M_SCSISES, "SCSI SES", "SCSI SES buffers")

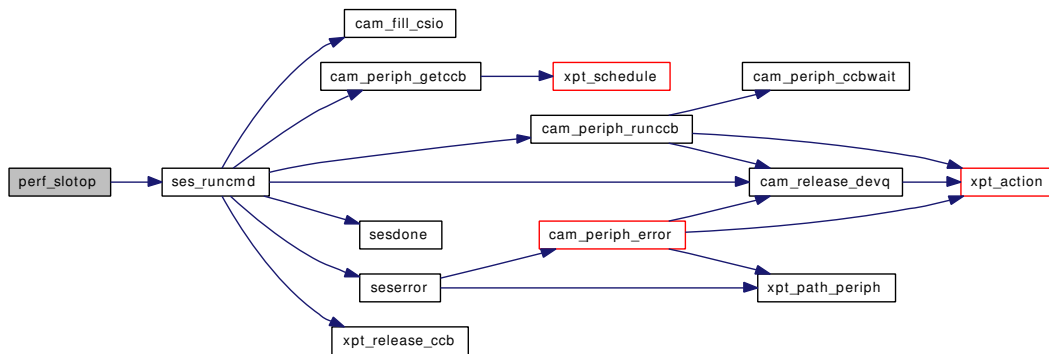
7.37.4.3 `static int perf_slotop` (`ses_softc_t *`, `uint8_t`, `uint8_t`, `int`) [`static`]

Definition at line 2513 of file `scsi_ses.c`.

References `MEMZERO`, `SAFT_SCRATCH`, `SAFTE_WT_SLTOP`, `SES_DLOG`, `SES_FREE`, `SES_MALLOC`, `ses_softc::ses_private`, `ses_runcmd()`, and `WRITE_BUFFER`.

Referenced by `saft_set_objstat()`.

Here is the call graph for this function:



7.37.4.4 PERIPHDRIVER_DECLARE (ses, sesdriver)

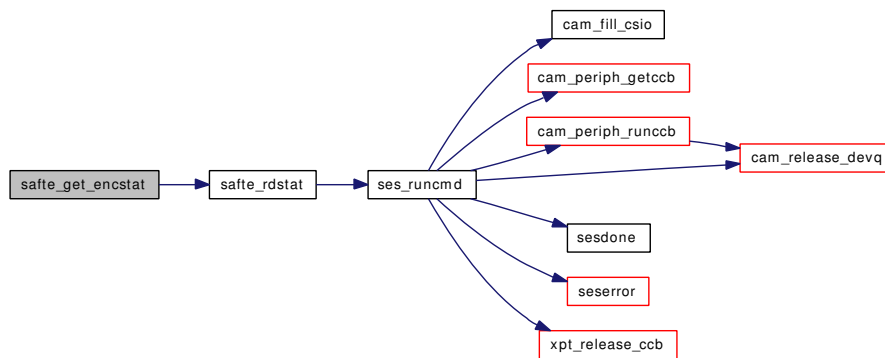
7.37.4.5 static int safte_get_encstat (ses_softc_t *, int) [static]

Definition at line 1730 of file scsi_ses.c.

References safte_rdstat().

Referenced by sesregister().

Here is the call graph for this function:



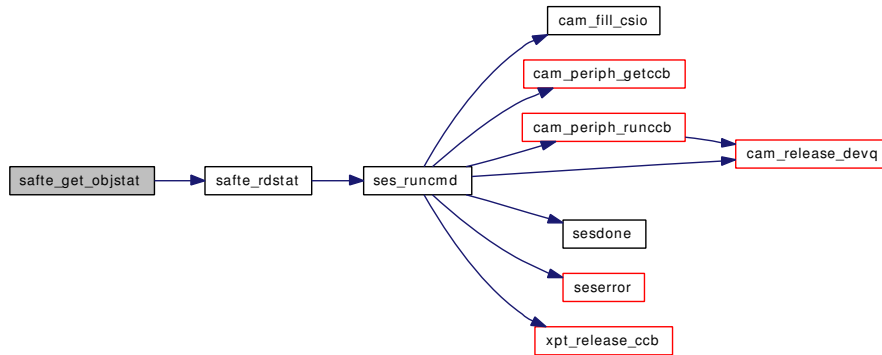
7.37.4.6 static int safte_get_objstat (ses_softc_t *, ses_objstat *, int) [static]

Definition at line 1760 of file scsi_ses.c.

References ses_objstat::cstat, ENCI_SVALID, encobj::encstat, ses_objstat::obj_id, safte_rdstat(), ses_softc::ses_encstat, ses_softc::ses_objmap, and encobj::svalid.

Referenced by sesregister().

Here is the call graph for this function:



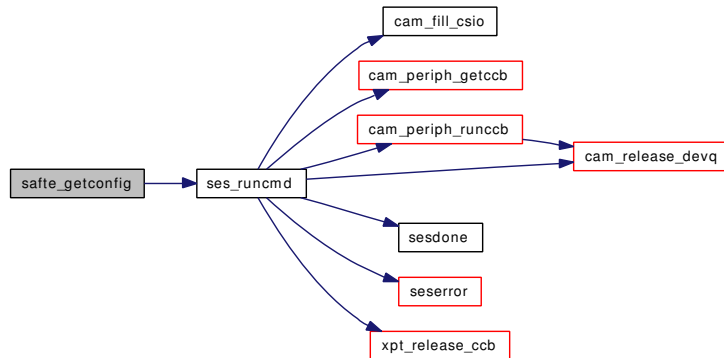
7.37.4.7 static int safte_getconfig (ses_softc_t *) [static]

Definition at line 1924 of file scsi_ses.c.

References scfg::DoorLock, scfg::Nalarm, scfg::Nfans, NPSEUDO_ALARM, scfg::Npwr, scfg::Nslots, scfg::Nspkrs, scfg::Ntherm, READ_BUFFER, SAFT_SCRATCH, SAFTE_RD_RDCFG, SES_FREE, SES_LOG, SES_MALLOC, ses_softc::ses_private, ses_runcmd(), and SES_VLOG.

Referenced by safte_softc_init().

Here is the call graph for this function:



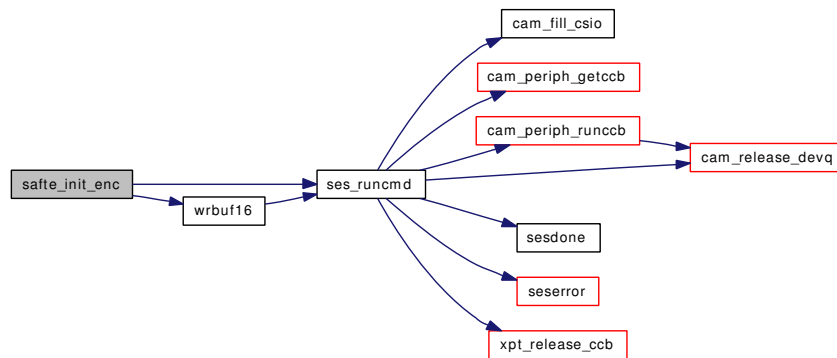
7.37.4.8 static int safte_init_enc (ses_softc_t *) [static]

Definition at line 1715 of file scsi_ses.c.

References SAFTE_WT_GLOBAL, SEND_DIAGNOSTIC, ses_runcmd(), and wrbuf16().

Referenced by sesregister().

Here is the call graph for this function:



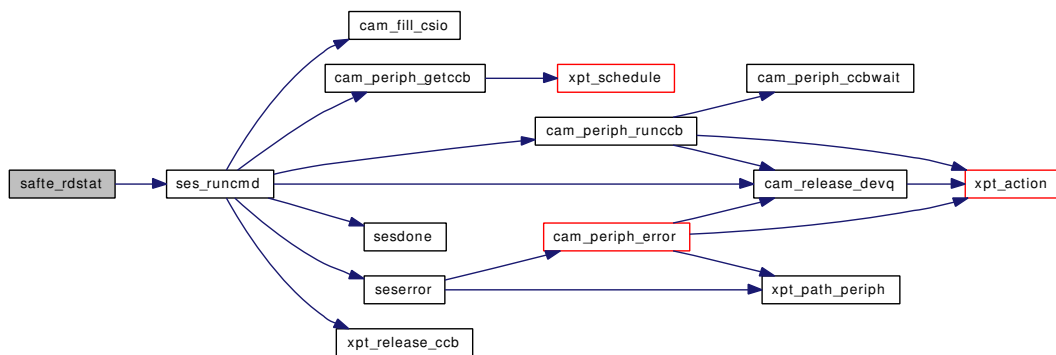
7.37.4.9 static int safte_rdstat (ses_softc_t *, int) [static]

Definition at line 1966 of file scsi_ses.c.

References ALL_ENC_STAT, encobj::encstat, scfg::Nfans, scfg::Nslots, scfg::Ntherm, READ_BUFFER, SAFT_BAIL, SAFTE_RD_RDESTS, ses_softc::ses_encstat, SES_ENCSTAT_CRITICAL, SES_ENCSTAT_INFO, SES_ENCSTAT_NONCRITICAL, SES_FREE, SES_LOG, SES_MALLOC, ses_softc::ses_nobjects, ses_softc::ses_objmap, SES_OBJSTAT_CRIT, SES_OBJSTAT_NOTINSTALLED, SES_OBJSTAT_OK, SES_OBJSTAT_UNKNOWN, SES_OBJSTAT_UNSUPPORTED, ses_softc::ses_private, ses_runcmd(), and encobj::svalid.

Referenced by safte_get_encstat(), and safte_get_objstat().

Here is the call graph for this function:



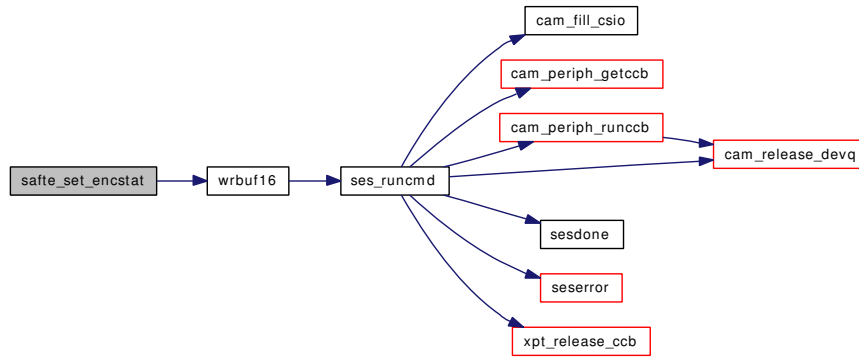
7.37.4.10 static int safte_set_encstat (ses_softc_t *, uint8_t, int) [static]

Definition at line 1736 of file scsi_ses.c.

References ALL_ENC_STAT, ENCL_SVALID, scfg::flag1, scfg::flag2, SAFT_FLG1_ALARM, SAFT_FLG1_GLOBFAIL, SAFT_FLG1_GLOBWARN, SAFTE_WT_GLOBAL, ses_softc::ses_encstat, SES_ENCSTAT_CRITICAL, SES_ENCSTAT_NONCRITICAL, SES_ENCSTAT_UNRECOV, ses_softc::ses_private, and wrbuf16().

Referenced by sesregister().

Here is the call graph for this function:



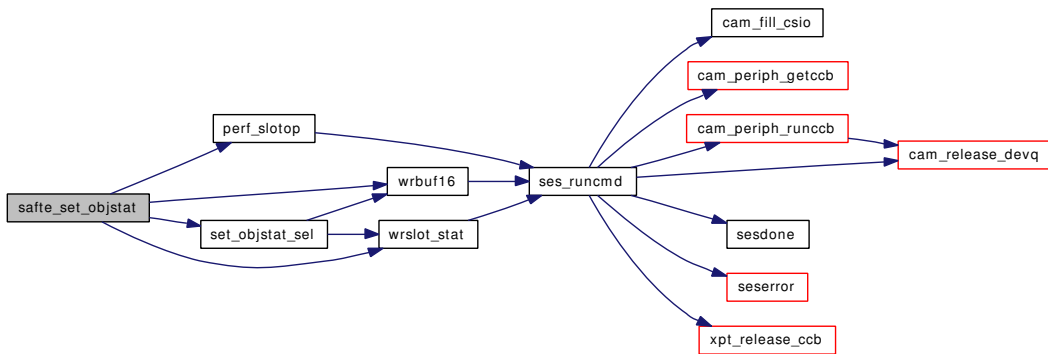
7.37.4.11 `static int safte_set_objstat (ses_softc_t *, ses_objstat *, int)` [static]

Definition at line 1779 of file scsi_ses.c.

References `ses_objstat::cstat`, `encobj::enctype`, `scfg::flag1`, `scfg::flag2`, `ses_objstat::obj_id`, `perf_slotop()`, `encobj::priv`, `scfg::pwoff`, `SAFT_FLG1_ALARM`, `SAFT_FLG1_ENCFANFAIL`, `SAFT_FLG1_ENCPWRFAIL`, `SAFT_FLG2_LOCKDOOR`, `SAFTE_WT_ACTPWS`, `SAFTE_WT_FANSPD`, `SAFTE_WT_GLOBAL`, `SES_DLOG`, `ses_softc::ses_objmap`, `ses_softc::ses_private`, `SESCTL_CSEL`, `SESCTL_RQSFLT`, `SESCTL_RQSID`, `SESCTL_RQSINS`, `SESCTL_RQSRMV`, `SESCTL_RQSTFAIL`, `SESCTL_RQSTON`, `SESTYP_ALARM`, `SESTYP_DEVICE`, `SESTYP_DOORLOCK`, `SESTYP_FAN`, `SESTYP_POWER`, `set_objstat_sel()`, `scfg::slotoff`, `encobj::svalid`, `wrbuf16()`, and `wrslot_stat()`.

Referenced by `sesregister()`.

Here is the call graph for this function:



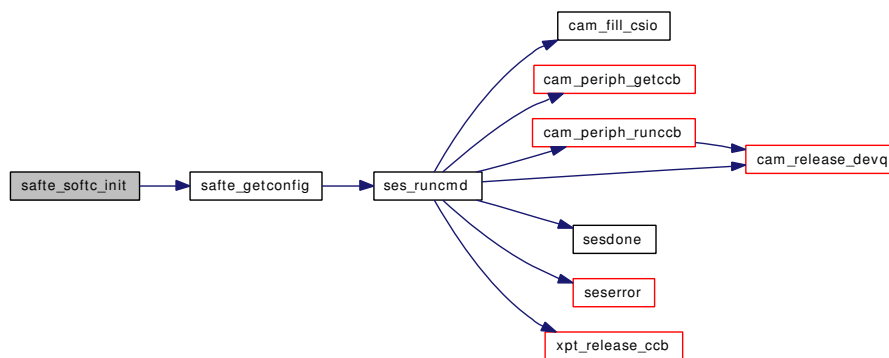
7.37.4.12 `static int safte_softc_init (ses_softc_t *, int)` [static]

Definition at line 1638 of file scsi_ses.c.

References `scfg::DoorLock`, `encobj::enctype`, `MEMZERO`, `scfg::Nfans`, `NPSEUDO_ALARM`, `NPSEUDO_THERM`, `scfg::Npwr`, `scfg::Nslots`, `scfg::Nspkrs`, `scfg::Ntherm`, `scfg::pwoff`, `SAFT_PRIVATE`, `safte_getconfig()`, `ses_softc::ses_encstat`, `SES_FREE`, `SES_MALLOC`, `ses_softc::ses_nobjects`, `ses_softc::ses_objmap`, `ses_softc::ses_private`, `SESTYP_ALARM`, `SESTYP_DEVICE`, `SESTYP_DOORLOCK`, `SESTYP_FAN`, `SESTYP_POWER`, `SESTYP_THERM`, and `scfg::slotoff`.

Referenced by `sesregister()`.

Here is the call graph for this function:



7.37.4.13 `static int ses_cfghdr (uint8_t *, int, SesCfgHdr *)` [static]

Definition at line 1293 of file `scsi_ses.c`.

References `SesCfgHdr::GenCode`, `gget32`, `gget8`, `SesCfgHdr::Nsubenc`, and `SES_CFGHDR_MINLEN`.

Referenced by `ses_getconfig()`.

7.37.4.14 `static int ses_decode (char *, int, uint8_t *, int, int, SesComStat *)` [static]

Definition at line 1404 of file `scsi_ses.c`.

References `SesComStat::comstat`, `SesComStat::comstatus`, `gget8`, and `PRINTF`.

Referenced by `ses_getputstat()`.

7.37.4.15 `static int ses_enddesc (uint8_t *, int, uint8_t, SesEncDesc *)` [static]

Definition at line 1322 of file `scsi_ses.c`.

References `gget8`, and `MEMCPY`.

Referenced by `ses_getconfig()`.

7.37.4.16 `static int ses_enchdr (uint8_t *, int, uint8_t, SesEncHdr *)` [static]

Definition at line 1304 of file `scsi_ses.c`.

References `gget8`, `SesEncHdr::Ntypes`, `SesEncHdr::Subencid`, and `SesEncHdr::VEnclen`.

Referenced by `ses_getconfig()`.

7.37.4.17 `static int ses_encode (char *, int, uint8_t *, int, int, SesComStat *)` [static]

Definition at line 1474 of file `scsi_ses.c`.

References `SesComStat::comstat`, `SesComStat::comstatus`, `PRINTF`, and `sset8`.

Referenced by `ses_getputstat()`.

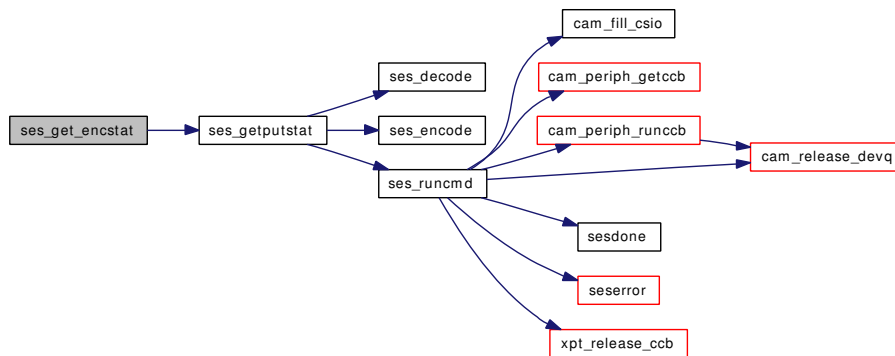
7.37.4.18 `static int ses_get_encstat (ses_softc_t *, int) [static]`

Definition at line 995 of file `scsi_ses.c`.

References `SesComStat::comstatus`, `ENCI_SVALID`, `ses_softc::ses_encstat`, and `ses_getputstat()`.

Referenced by `sesregister()`.

Here is the call graph for this function:



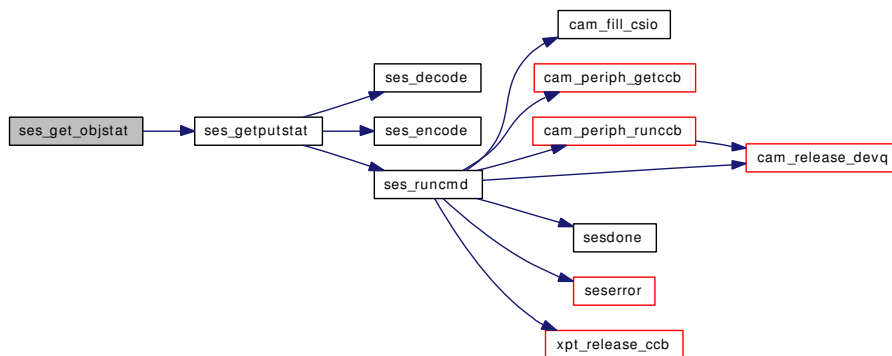
7.37.4.19 `static int ses_get_objstat (ses_softc_t *, ses_objstat *, int) [static]`

Definition at line 1022 of file `scsi_ses.c`.

References `SesComStat::comstat`, `SesComStat::comstatus`, `ses_objstat::cstat`, `encobj::encstat`, `ses_objstat::obj_id`, `ses_getputstat()`, `ses_softc::ses_objmap`, and `encobj::svalid`.

Referenced by `sesregister()`.

Here is the call graph for this function:



7.37.4.20 `static int ses_getconfig (ses_softc_t *) [static]`

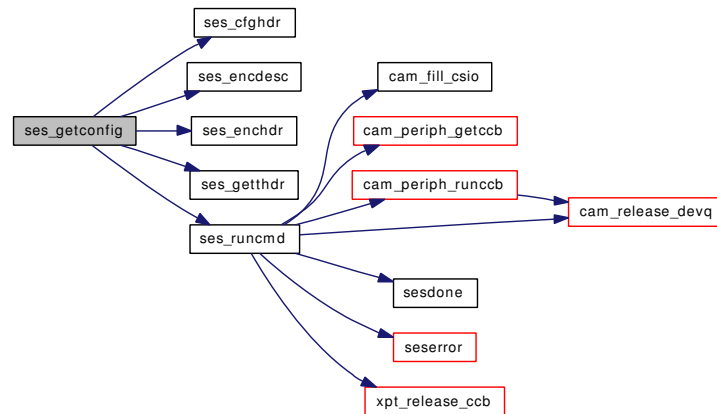
Definition at line 1065 of file `scsi_ses.c`.

References `CFLLEN`, `SesThdr::enc_maxelt`, `SesThdr::enc_subenc`, `SesThdr::enc_tlen`, `SesThdr::enc_type`, `encobj::enctype`, `SesCfgHdr::GenCode`, `MEMZERO`, `SesCfgHdr::Nsubenc`, `SesEncHdr::Ntypes`,

RECEIVE_DIAGNOSTIC, SCSZ, ses_cfghdr(), sscfg::ses_eltmap, ses_endedesc(), ses_enchr(), SES_ENCHDR_MINLEN, SES_FREE, ses_getthdr(), SES_LOG, SES_MALLOC, ses_softc::ses_nobjects, sscfg::ses_ntypes, ses_softc::ses_objmap, typidx::ses_oidx, ses_softc::ses_private, ses_runcmd(), typidx::ses_tid, sscfg::ses_tpid, SES_VLOG, SesConfigPage, SesEncHdr::Subencid, encobj::subenclosure, and SesEncHdr::VEnclen.

Referenced by ses_softc_init().

Here is the call graph for this function:



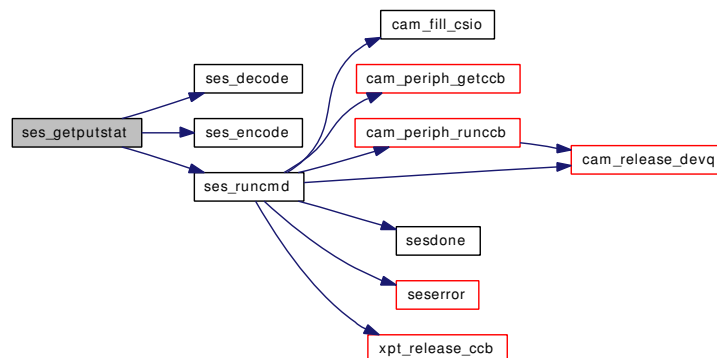
7.37.4.21 static int ses_getputstat (ses_softc_t *, int, SesComStat *, int, int) [static]

Definition at line 1213 of file scsi_ses.c.

References RECEIVE_DIAGNOSTIC, SEND_DIAGNOSTIC, ses_decode(), sscfg::ses_eltmap, ses_encode(), SES_FREE, SES_MALLOC, ses_softc::ses_nobjects, sscfg::ses_ntypes, typidx::ses_oidx, ses_softc::ses_private, ses_runcmd(), typidx::ses_tid, sscfg::ses_tpid, and SesStatusPage.

Referenced by ses_get_encstat(), ses_get_objstat(), ses_set_encstat(), and ses_set_objstat().

Here is the call graph for this function:



7.37.4.22 static int ses_getthdr (uint8_t *, int, int, SesThdr *) [static]

Definition at line 1347 of file scsi_ses.c.

References SesThdr::enc_maxelt, SesThdr::enc_subenc, SesThdr::enc_tlen, SesThdr::enc_type, gget8, and SES_CFGHDR_MINLEN.

Referenced by ses_getconfig().

7.37.4.23 static int ses_init_enc (ses_softc_t *) [static]

Definition at line 989 of file scsi_ses.c.

Referenced by sesregister().

7.37.4.24 static void ses_log (struct ses_softc *, const char *, ...) [static]

Definition at line 700 of file scsi_ses.c.

References ses_softc::periph, cam_periph::periph_name, and cam_periph::unit_number.

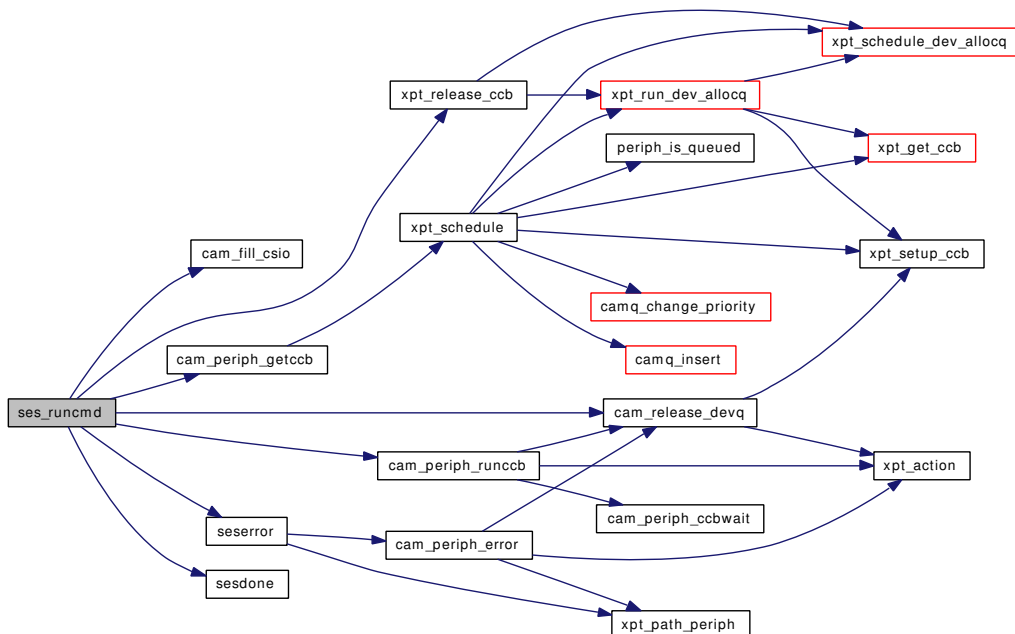
7.37.4.25 static int ses_runcmd (struct ses_softc *, char *, int, char *, int *) [static]

Definition at line 656 of file scsi_ses.c.

References CAM_DEV_QFRZN, CAM_DIR_IN, CAM_DIR_NONE, CAM_DIR_OUT, cam_fill_csio(), cam_periph_getccb(), cam_periph_runccb(), cam_release_devq(), ccb::ccb_h, cdb_t::cdb_bytes, ccb_scsiio::cdb_io, ccb::csio, IOCDBLEN, MSG_SIMPLE_Q_TAG, ccb_hdr::path, ses_softc::periph, ccb_scsiio::resid, SES_CFLAGS, SES_FLAGS, sesdone(), seserror(), ccb_hdr::status, and xpt_release_ccb().

Referenced by perf_slotop(), safte_getconfig(), safte_init_enc(), safte_rdstat(), ses_getconfig(), ses_getputstat(), wrbuf16(), and wrslot_stat().

Here is the call graph for this function:



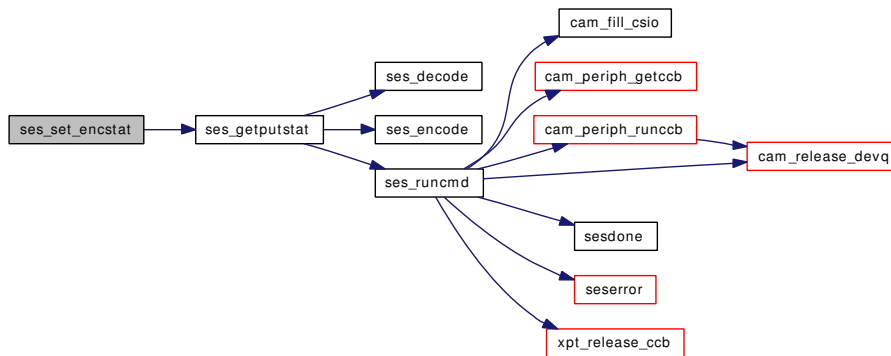
7.37.4.26 static int ses_set_encstat (ses_softc_t *, uint8_t, int) [static]

Definition at line 1008 of file scsi_ses.c.

References SesComStat::comstatus, ses_softc::ses_encstat, and ses_getputstat().

Referenced by sesregister().

Here is the call graph for this function:



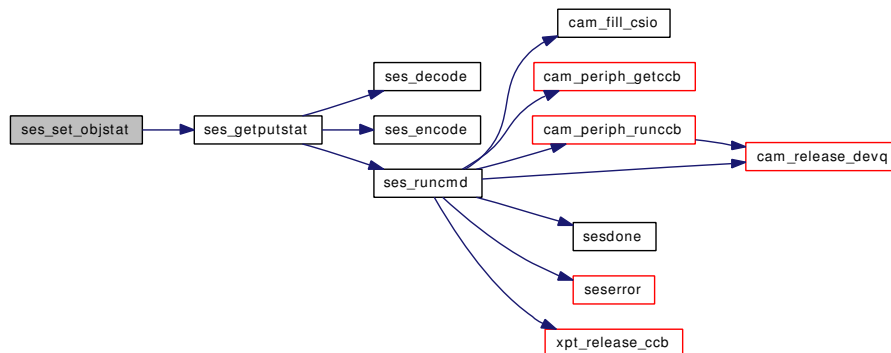
7.37.4.27 static int ses_set_objstat (ses_softc_t *, ses_objstat *, int) [static]

Definition at line 1045 of file scsi_ses.c.

References SesComStat::comstat, SesComStat::comstatus, ses_objstat::cstat, ses_objstat::obj_id, ses_getputstat(), ses_softc::ses_objmap, and SESCTL_CSEL.

Referenced by sesregister().

Here is the call graph for this function:



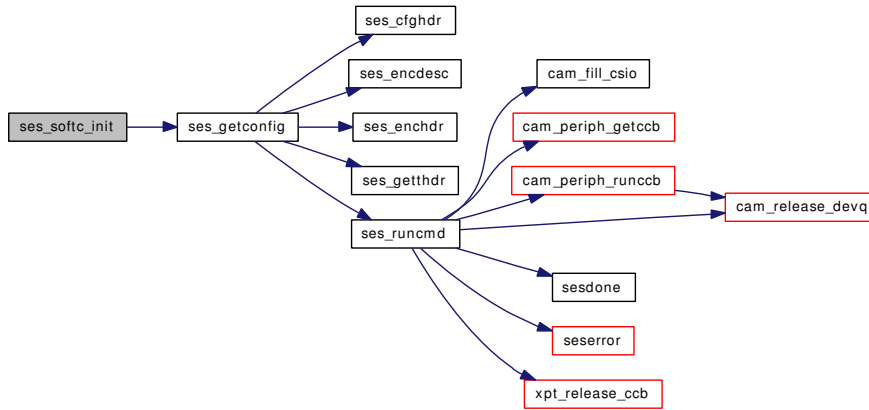
7.37.4.28 static int ses_softc_init (ses_softc_t *, int) [static]

Definition at line 951 of file scsi_ses.c.

References sscfg::ses_eltmap, ses_softc::ses_encstat, SES_FREE, ses_getconfig(), SES_MALLOC, ses_softc::ses_nobjects, sscfg::ses_ntypes, ses_softc::ses_objmap, ses_softc::ses_private, and sscfg::ses_typidx.

Referenced by sesregister().

Here is the call graph for this function:



7.37.4.29 static `enctyp ses_type (void *, int)` [static]

Definition at line 729 of file `scsi_ses.c`.

References `SAFTE_END`, `SAFTE_LEN`, `SAFTE_START`, `SEN_ID`, `SEN_ID_LEN`, `SES_NONE`, `SES_SAFTE`, `SES_SEN`, `SES_SES`, `SES_SES_PASSTHROUGH`, `SES_SES_SCSI2`, `STRNCMP`, and `T_ENCLOSURE`.

Referenced by `sesasync()`, and `sesregister()`.

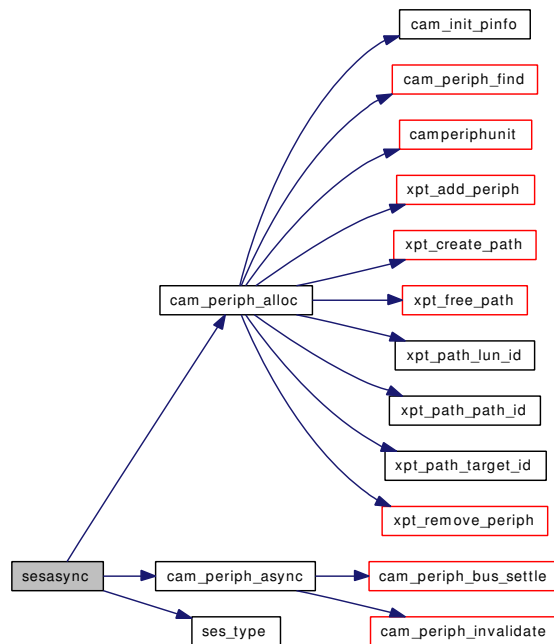
7.37.4.30 static void `sesasync (void *, u_int32_t, struct cam_path *, void *)` [static]

Definition at line 257 of file `scsi_ses.c`.

References `AC_FOUND_DEVICE`, `scsi_inquiry_data::additional_length`, `cam_periph_alloc()`, `cam_periph_async()`, `CAM_PERIPH_BIO`, `CAM_REQ_CMP`, `CAM_REQ_INPROG`, `ccb_getdev::ccb_h`, `ccb_getdev::inq_data`, `ccb_hdr::path`, `SES_SAFTE`, `SES_SEN`, `SES_SES`, `SES_SES_PASSTHROUGH`, `SES_SES_SCSI2`, `ses_type()`, `sescleanup`, `sesoninvalidate`, `sesregister`, and `sesstart`.

Referenced by `sesinit()`, `sesoninvalidate()`, and `sesregister()`.

Here is the call graph for this function:

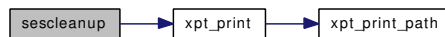


7.37.4.31 static void sescleanup (struct [cam_periph](#) * *periph*) [static]

Definition at line 244 of file `scsi_ses.c`.

References `cam_periph::path`, `ses_softc::periph`, `ses_softc::ses_dev`, `cam_periph::softc`, and `xpt_print()`.

Here is the call graph for this function:

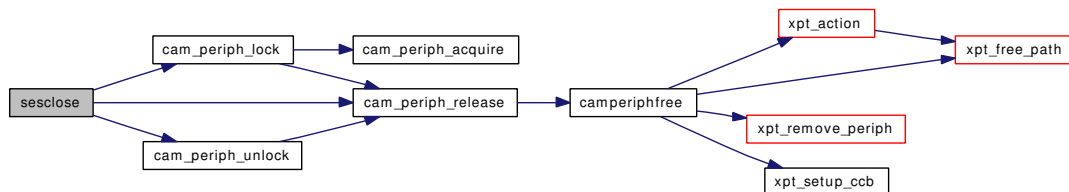


7.37.4.32 static int sesclose (struct `cdev` * *dev*, int *flag*, int *fmt*, struct `thread` * *td*) [static]

Definition at line 464 of file `scsi_ses.c`.

References `cam_periph_lock()`, `cam_periph_release()`, `cam_periph_unlock()`, `ses_softc::periph`, `SES_FLAG_OPEN`, `ses_softc::ses_flags`, and `cam_periph::softc`.

Here is the call graph for this function:



7.37.4.33 `static void sesdone (struct cam_periph *, union ccb *)` [static]

Definition at line 502 of file `scsi_ses.c`.

References `ccb_hdr::cbfcn`, and `ccb::ccb_h`.

Referenced by `ses_runcmd()`.

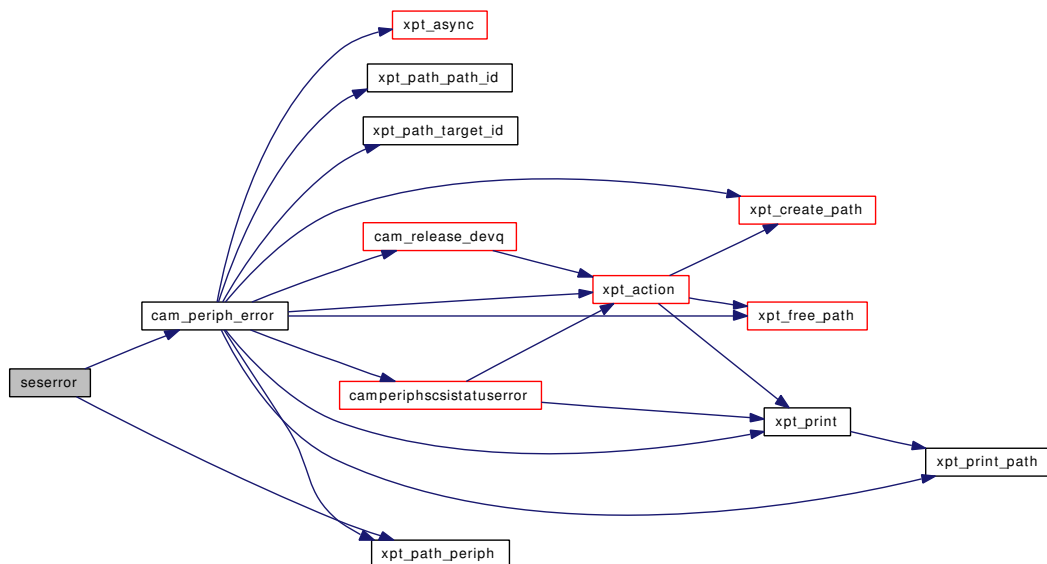
7.37.4.34 `static int seserror (union ccb *, u_int32_t, u_int32_t)` [static]

Definition at line 508 of file `scsi_ses.c`.

References `cam_periph_error()`, `ccb::ccb_h`, `ccb_hdr::path`, `ses_softc::periph`, `ses_softc::ses_saved_ccb`, `cam_periph::softc`, and `xpt_path_periph()`.

Referenced by `ses_runcmd()`, and `sesioctl()`.

Here is the call graph for this function:



7.37.4.35 `static void sesinit (void)` [static]

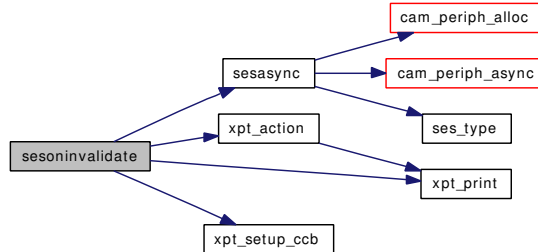
Definition at line 189 of file `scsi_ses.c`.

References `AC_FOUND_DEVICE`, `ccb_setasync::callback`, `ccb_setasync::callback_arg`, `CAM_LUN_WILDCARD`, `CAM_REQ_CMP`, `CAM_TARGET_WILDCARD`, `CAM_XPT_PATH_ID`, `ccb::ccb_h`, `ccb_setasync::ccb_h`, `ccb_setasync::event_enable`, `ccb_hdr::func_code`, `sesasync()`, `ccb_hdr::status`, `xpt_action()`, `xpt_create_path()`, `xpt_free_path()`, `XPT_SASYNC_CB`, and `xpt_setup_ccb()`.

Here is the call graph for this function:

sesasync(), cam_periph::softc, xpt_action(), xpt_print(), XPT_SASYNC_CB, and xpt_setup_ccb().

Here is the call graph for this function:

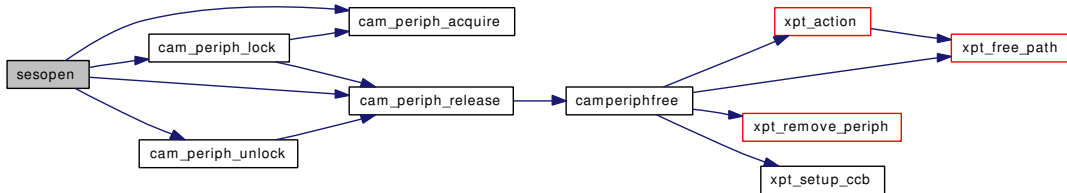


7.37.4.38 static int sesopen (struct cdev * dev, int flags, int fmt, struct thread * td) [static]

Definition at line 408 of file scsi_ses.c.

References cam_periph_acquire(), cam_periph_lock(), cam_periph_release(), cam_periph_unlock(), CAM_REQ_CMP, ses_softc::periph, SES_FLAG_INITIALIZED, SES_FLAG_INVALID, SES_FLAG_OPEN, ses_softc::ses_flags, ses_softc::ses_vec, cam_periph::softc, and encvec::softc_init.

Here is the call graph for this function:

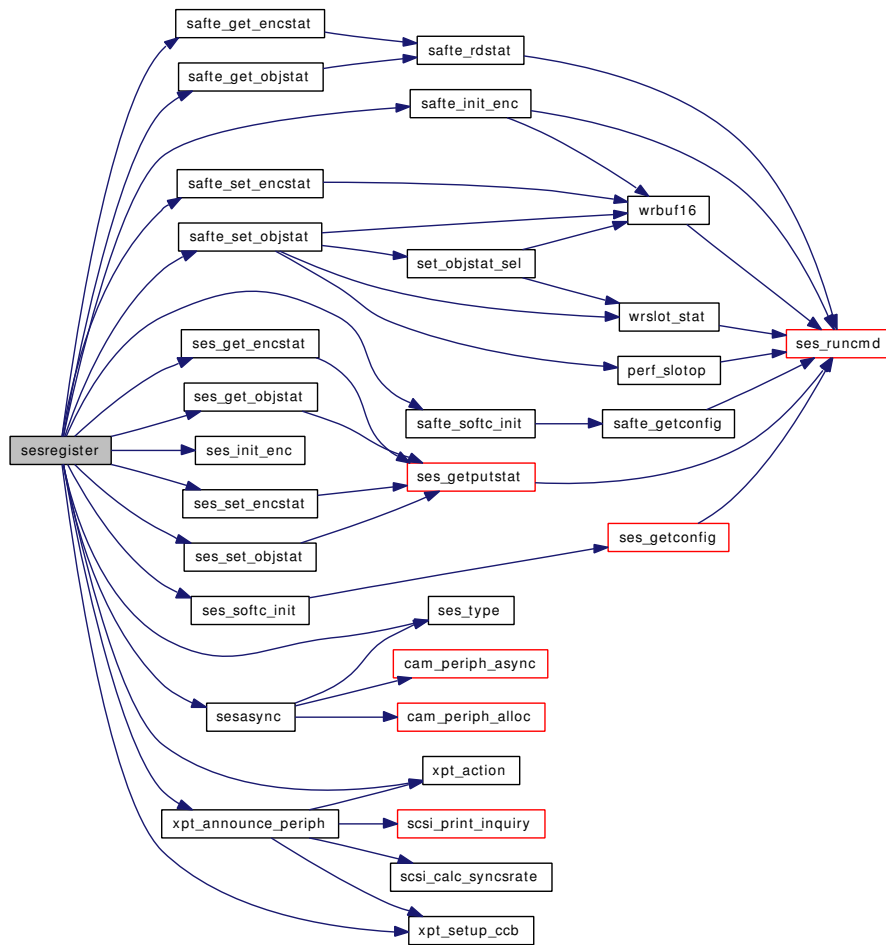


7.37.4.39 static cam_status sesregister (struct cam_periph * periph, void * arg) [static]

Definition at line 309 of file scsi_ses.c.

References AC_LOST_DEVICE, ccb_setasync::callback, ccb_setasync::callback_arg, CAM_REQ_CMP, CAM_REQ_CMP_ERR, ccb_setasync::ccb_h, ccb_setasync::event_enable, ccb_hdr::func_code, encvec::get_encstat, encvec::get_objstat, encvec::init_enc, ccb_getdev::inq_data, cam_periph::path, ses_softc::periph, cam_periph::periph_name, safte_get_encstat(), safte_get_objstat(), safte_init_enc(), safte_set_encstat(), safte_set_objstat(), safte_softc_init(), ses_cdevsw, ses_softc::ses_dev, ses_get_encstat(), ses_get_objstat(), ses_init_enc(), SES_NONE, SES_SAFT, SES_SEN, SES_SES, SES_SES_PASSTHROUGH, SES_SES_SCSI2, ses_set_encstat(), ses_set_objstat(), ses_softc_init(), ses_type(), ses_softc::ses_type, ses_softc::ses_vec, sesasync(), encvec::set_encstat, encvec::set_objstat, cam_periph::softc, encvec::softc_init, cam_periph::unit_number, xpt_action(), xpt_announce_periph(), XPT_SASYNC_CB, and xpt_setup_ccb().

Here is the call graph for this function:



7.37.4.40 static void sesstart (struct cam_periph * p, union ccb * sccb) [static]

Definition at line 490 of file scsi_ses.c.

References CAM_PRIORITY_NONE, ccb::ccb_h, cam_periph::immediate_priority, cam_periph::pinfo, and cam_pinfo::priority.

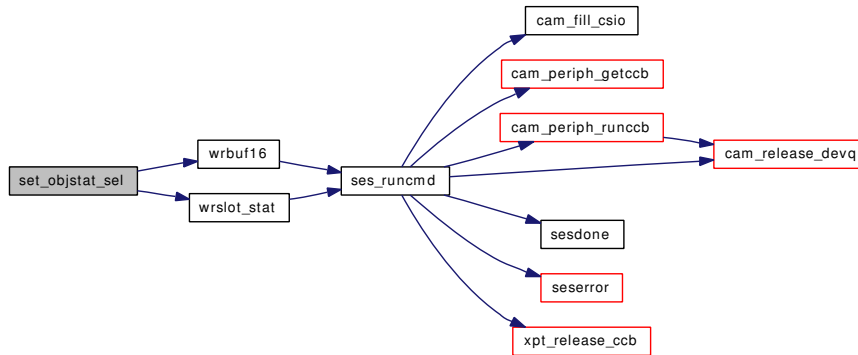
7.37.4.41 static int set_objstat_sel (ses_softc_t *, ses_objstat *, int) [static]

Definition at line 2352 of file scsi_ses.c.

References ses_objstat::cstat, encobj::enctype, scfg::flag1, scfg::flag2, ses_objstat::obj_id, encobj::priv, scfg::pwoff, SAFT_FLG1_ALARM, SAFT_FLG2_LOCKDOOR, SAFTE_WT_ACTPWS, SAFTE_WT_FANSPD, SAFTE_WT_GLOBAL, ses_softc::ses_objmap, ses_softc::ses_private, SESCTL_DISABLE, SESCTL_PRDFAIL, SESTYP_ALARM, SESTYP_DEVICE, SESTYP_DOORLOCK, SESTYP_FAN, SESTYP_POWER, encobj::svalid, wrbuf16(), and wrslot_stat().

Referenced by safte_set_objstat().

Here is the call graph for this function:



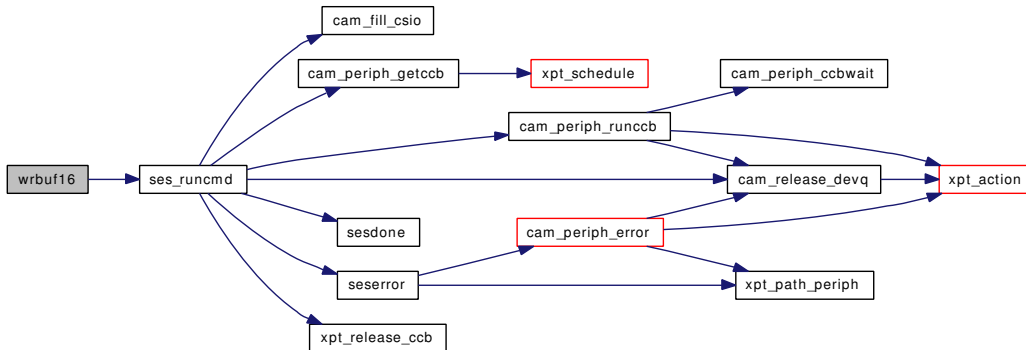
7.37.4.42 static int wrbuf16 (ses_softc_t *, uint8_t, uint8_t, uint8_t, uint8_t, int) [static]

Definition at line 2436 of file scsi_ses.c.

References MEMZERO, SES_DLOG, SES_FREE, SES_MALLOC, ses_softc::ses_private, ses_runcmd(), and WRITE_BUFFER.

Referenced by safte_init_enc(), safte_set_encstat(), safte_set_objstat(), and set_objstat_sel().

Here is the call graph for this function:



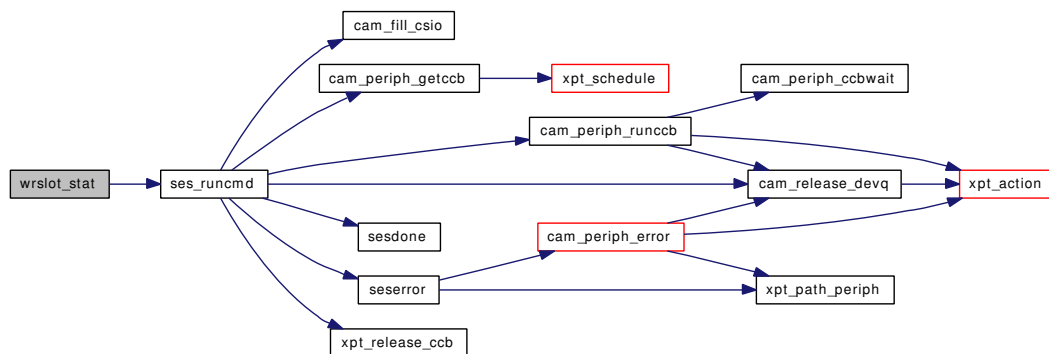
7.37.4.43 static void wrslot_stat (ses_softc_t *, int) [static]

Definition at line 2471 of file scsi_ses.c.

References MEMZERO, scfg::Nslots, encobj::priv, SAFTE_WT_DSTAT, SES_DLOG, SES_FREE, SES_MALLOC, ses_softc::ses_objmap, ses_softc::ses_private, ses_runcmd(), scfg::slotoff, and WRITE_BUFFER.

Referenced by safte_set_objstat(), and set_objstat_sel().

Here is the call graph for this function:



7.37.5 Variable Documentation

7.37.5.1 `char* safte_2little = "Too Little Data Returned (%d) at line %d\n"` [static]

Definition at line 1628 of file `scsi_ses.c`.

7.37.5.2 `struct cdevsw ses_cdevsw` [static]

Initial value:

```

{
    .d_version =    D_VERSION,
    .d_open =      sesopen,
    .d_close =     sesclose,
    .d_ioctl =     sesioctl,
    .d_name =      "ses",
    .d_flags =     D_NEEDGIANT,
}

```

Definition at line 179 of file `scsi_ses.c`.

Referenced by `sesregister()`.

7.37.5.3 `periph_dtor_t sescleanup` [static]

Definition at line 165 of file `scsi_ses.c`.

Referenced by `sesasync()`.

7.37.5.4 `d_close_t sesclose` [static]

Definition at line 160 of file `scsi_ses.c`.

7.37.5.5 `struct periph_driver sesdriver` [static]

Initial value:

```

{
    sesinit, "ses",
}

```

```
        TAILQ_HEAD_INITIALIZER(sesdriver.units), 0  
    }
```

Definition at line 172 of file scsi_ses.c.

7.37.5.6 [periph_init_t sesinit](#) [static]

Definition at line 162 of file scsi_ses.c.

7.37.5.7 [d_ioctl_t sesioctl](#) [static]

Definition at line 161 of file scsi_ses.c.

7.37.5.8 [periph_oninv_t sesoninvalidate](#) [static]

Definition at line 164 of file scsi_ses.c.

Referenced by sesasync().

7.37.5.9 [d_open_t sesopen](#) [static]

Definition at line 159 of file scsi_ses.c.

7.37.5.10 [periph_ctor_t sesregister](#) [static]

Definition at line 163 of file scsi_ses.c.

Referenced by sesasync().

7.37.5.11 [periph_start_t sesstart](#) [static]

Definition at line 166 of file scsi_ses.c.

Referenced by sesasync().

7.38 /usr/src/sys/cam/scsi/scsi_ses.h File Reference

This graph shows which files directly or indirectly include this file:



Data Structures

- struct [ses_object](#)
- struct [ses_objstat](#)
- union [ses_hlptxt](#)

Defines

- #define [SESIOC](#) ('s' - 040)
- #define [SESIOC_GETNOBJ_IO](#)(SESIOC, 1)
- #define [SESIOC_GETOBJMAP_IO](#)(SESIOC, 2)
- #define [SESIOC_GETENCSTAT_IO](#)(SESIOC, 3)
- #define [SESIOC_SETENCSTAT_IO](#)(SESIOC, 4)
- #define [SESIOC_GETOBJSTAT_IO](#)(SESIOC, 5)
- #define [SESIOC_SETOBJSTAT_IO](#)(SESIOC, 6)
- #define [SESIOC_GETTEXT_IO](#)(SESIOC, 7)
- #define [SESIOC_INIT_IO](#)(SESIOC, 8)
- #define [SESTYP_UNSPECIFIED](#) 0x00
- #define [SESTYP_DEVICE](#) 0x01
- #define [SESTYP_POWER](#) 0x02
- #define [SESTYP_FAN](#) 0x03
- #define [SESTYP_THERM](#) 0x04
- #define [SESTYP_DOORLOCK](#) 0x05
- #define [SESTYP_ALARM](#) 0x06
- #define [SESTYP_ESCC](#) 0x07
- #define [SESTYP_SCC](#) 0x08
- #define [SESTYP_NVRAM](#) 0x09
- #define [SESTYP_UPS](#) 0x0b
- #define [SESTYP_DISPLAY](#) 0x0c
- #define [SESTYP_KEYPAD](#) 0x0d
- #define [SESTYP_SCSIXVR](#) 0x0f
- #define [SESTYP_LANGUAGE](#) 0x10
- #define [SESTYP_COMPORT](#) 0x11
- #define [SESTYP_VOM](#) 0x12
- #define [SESTYP_AMMETER](#) 0x13
- #define [SESTYP_SCSI_TGT](#) 0x14
- #define [SESTYP_SCSI_INI](#) 0x15
- #define [SESTYP_SUBENC](#) 0x16
- #define [SES_ENCSTAT_UNRECOV](#) 0x1
- #define [SES_ENCSTAT_CRITICAL](#) 0x2
- #define [SES_ENCSTAT_NONCRITICAL](#) 0x4
- #define [SES_ENCSTAT_INFO](#) 0x8

- #define [SES_OBJSTAT_UNSUPPORTED](#) 0
- #define [SES_OBJSTAT_OK](#) 1
- #define [SES_OBJSTAT_CRIT](#) 2
- #define [SES_OBJSTAT_NONCRIT](#) 3
- #define [SES_OBJSTAT_UNRECOV](#) 4
- #define [SES_OBJSTAT_NOTINSTALLED](#) 5
- #define [SES_OBJSTAT_UNKNOWN](#) 6
- #define [SES_OBJSTAT_NOTAVAIL](#) 7
- #define [SESCTL_CSEL](#) 0x80
- #define [SESCTL_PRDFAIL](#) 0x40
- #define [SESCTL_DISABLE](#) 0x20
- #define [SESCTL_RSTSWAP](#) 0x10
- #define [SESCTL_DRVLCK](#) 0x40
- #define [SESCTL_RQSINS](#) 0x08
- #define [SESCTL_RQSRMV](#) 0x04
- #define [SESCTL_RQSID](#) 0x02
- #define [SESCTL_RQSFLT](#) 0x20
- #define [SESCTL_DEVOFF](#) 0x10
- #define [SESCTL_RQSTFAIL](#) 0x40
- #define [SESCTL_RQSTON](#) 0x20

Typedefs

- typedef unsigned char [ses_encstat](#)

7.38.1 Define Documentation

7.38.1.1 #define [SES_ENCSTAT_CRITICAL](#) 0x2

Definition at line 118 of file [scsi_ses.h](#).

Referenced by [safte_rdstat\(\)](#), and [safte_set_encstat\(\)](#).

7.38.1.2 #define [SES_ENCSTAT_INFO](#) 0x8

Definition at line 120 of file [scsi_ses.h](#).

Referenced by [safte_rdstat\(\)](#).

7.38.1.3 #define [SES_ENCSTAT_NONCRITICAL](#) 0x4

Definition at line 119 of file [scsi_ses.h](#).

Referenced by [safte_rdstat\(\)](#), and [safte_set_encstat\(\)](#).

7.38.1.4 #define [SES_ENCSTAT_UNRECOV](#) 0x1

Definition at line 117 of file [scsi_ses.h](#).

Referenced by [safte_set_encstat\(\)](#).

7.38.1.5 #define SES_OBJSTAT_CRIT 2

Definition at line 133 of file scsi_ses.h.

Referenced by safte_rdstat().

7.38.1.6 #define SES_OBJSTAT_NONCRIT 3

Definition at line 134 of file scsi_ses.h.

7.38.1.7 #define SES_OBJSTAT_NOTAVAIL 7

Definition at line 138 of file scsi_ses.h.

7.38.1.8 #define SES_OBJSTAT_NOTINSTALLED 5

Definition at line 136 of file scsi_ses.h.

Referenced by safte_rdstat().

7.38.1.9 #define SES_OBJSTAT_OK 1

Definition at line 132 of file scsi_ses.h.

Referenced by safte_rdstat().

7.38.1.10 #define SES_OBJSTAT_UNKNOWN 6

Definition at line 137 of file scsi_ses.h.

Referenced by safte_rdstat().

7.38.1.11 #define SES_OBJSTAT_UNRECOV 4

Definition at line 135 of file scsi_ses.h.

7.38.1.12 #define SES_OBJSTAT_UNSUPPORTED 0

Definition at line 131 of file scsi_ses.h.

Referenced by safte_rdstat().

7.38.1.13 #define SESCTL_CSEL 0x80

Definition at line 148 of file scsi_ses.h.

Referenced by safte_set_objstat(), and ses_set_objstat().

7.38.1.14 #define SESCTL_DEVOFF 0x10

Definition at line 161 of file scsi_ses.h.

7.38.1.15 #define SESCTL_DISABLE 0x20

Definition at line 150 of file scsi_ses.h.

Referenced by set_objstat_sel().

7.38.1.16 #define SESCTL_DRVLCK 0x40

Definition at line 155 of file scsi_ses.h.

7.38.1.17 #define SESCTL_PRDFAIL 0x40

Definition at line 149 of file scsi_ses.h.

Referenced by set_objstat_sel().

7.38.1.18 #define SESCTL_RQSFLT 0x20

Definition at line 160 of file scsi_ses.h.

Referenced by safte_set_objstat().

7.38.1.19 #define SESCTL_RQSID 0x02

Definition at line 158 of file scsi_ses.h.

Referenced by safte_set_objstat().

7.38.1.20 #define SESCTL_RQSINS 0x08

Definition at line 156 of file scsi_ses.h.

Referenced by safte_set_objstat().

7.38.1.21 #define SESCTL_RQSRMV 0x04

Definition at line 157 of file scsi_ses.h.

Referenced by safte_set_objstat().

7.38.1.22 #define SESCTL_RQSTFAIL 0x40

Definition at line 164 of file scsi_ses.h.

Referenced by safte_set_objstat().

7.38.1.23 #define SESCTL_RQSTON 0x20

Definition at line 165 of file scsi_ses.h.

Referenced by safte_set_objstat().

7.38.1.24 #define SESCTL_RSTSWAP 0x10

Definition at line 151 of file scsi_ses.h.

7.38.1.25 #define SESIOC ('s' - 040)

Definition at line 32 of file scsi_ses.h.

7.38.1.26 #define SESIOC_GETENCSTAT_IO(SESIOC, 3)

Definition at line 35 of file scsi_ses.h.

Referenced by sesioctl().

7.38.1.27 #define SESIOC_GETNOBJ_IO(SESIOC, 1)

Definition at line 33 of file scsi_ses.h.

Referenced by sesioctl().

7.38.1.28 #define SESIOC_GETOBJMAP_IO(SESIOC, 2)

Definition at line 34 of file scsi_ses.h.

Referenced by sesioctl().

7.38.1.29 #define SESIOC_GETOBJSTAT_IO(SESIOC, 5)

Definition at line 37 of file scsi_ses.h.

Referenced by sesioctl().

7.38.1.30 #define SESIOC_GETTEXT_IO(SESIOC, 7)

Definition at line 39 of file scsi_ses.h.

7.38.1.31 #define SESIOC_INIT_IO(SESIOC, 8)

Definition at line 40 of file scsi_ses.h.

Referenced by sesioctl().

7.38.1.32 #define SESIOC_SETENCSTAT_IO(SESIOC, 4)

Definition at line 36 of file scsi_ses.h.

Referenced by sesioctl().

7.38.1.33 #define SESIOC_SETOBJSTAT_IO(SESIOC, 6)

Definition at line 38 of file scsi_ses.h.

Referenced by sesioctl().

7.38.1.34 #define SESTYP_ALARM 0x06

Definition at line 97 of file scsi_ses.h.

Referenced by safte_set_objstat(), safte_softc_init(), and set_objstat_sel().

7.38.1.35 #define SESTYP_AMMETER 0x13

Definition at line 108 of file scsi_ses.h.

7.38.1.36 #define SESTYP_COMPORT 0x11

Definition at line 106 of file scsi_ses.h.

7.38.1.37 #define SESTYP_DEVICE 0x01

Definition at line 92 of file scsi_ses.h.

Referenced by safte_set_objstat(), safte_softc_init(), and set_objstat_sel().

7.38.1.38 #define SESTYP_DISPLAY 0x0c

Definition at line 102 of file scsi_ses.h.

7.38.1.39 #define SESTYP_DOORLOCK 0x05

Definition at line 96 of file scsi_ses.h.

Referenced by safte_set_objstat(), safte_softc_init(), and set_objstat_sel().

7.38.1.40 #define SESTYP_ESCC 0x07

Definition at line 98 of file scsi_ses.h.

7.38.1.41 #define SESTYP_FAN 0x03

Definition at line 94 of file scsi_ses.h.

Referenced by safte_set_objstat(), safte_softc_init(), and set_objstat_sel().

7.38.1.42 #define SESTYP_KEYPAD 0x0d

Definition at line 103 of file scsi_ses.h.

7.38.1.43 #define SESTYP_LANGUAGE 0x10

Definition at line 105 of file scsi_ses.h.

7.38.1.44 #define SESTYP_NVRAM 0x09

Definition at line 100 of file scsi_ses.h.

7.38.1.45 #define SESTYP_POWER 0x02

Definition at line 93 of file scsi_ses.h.

Referenced by safte_set_objstat(), safte_softc_init(), and set_objstat_sel().

7.38.1.46 #define SESTYP_SCC 0x08

Definition at line 99 of file scsi_ses.h.

7.38.1.47 #define SESTYP_SCSI_INI 0x15

Definition at line 110 of file scsi_ses.h.

7.38.1.48 #define SESTYP_SCSI_TGT 0x14

Definition at line 109 of file scsi_ses.h.

7.38.1.49 #define SESTYP_SCSIXVR 0x0f

Definition at line 104 of file scsi_ses.h.

7.38.1.50 #define SESTYP_SUBENC 0x16

Definition at line 111 of file scsi_ses.h.

7.38.1.51 #define SESTYP_THERM 0x04

Definition at line 95 of file scsi_ses.h.

Referenced by safte_softc_init().

7.38.1.52 #define SESTYP_UNSPECIFIED 0x00

Definition at line 91 of file scsi_ses.h.

7.38.1.53 #define SESTYP_UPS 0x0b

Definition at line 101 of file scsi_ses.h.

7.38.1.54 `#define SESTYP_VOM 0x12`

Definition at line 107 of file `scsi_ses.h`.

7.38.2 Typedef Documentation

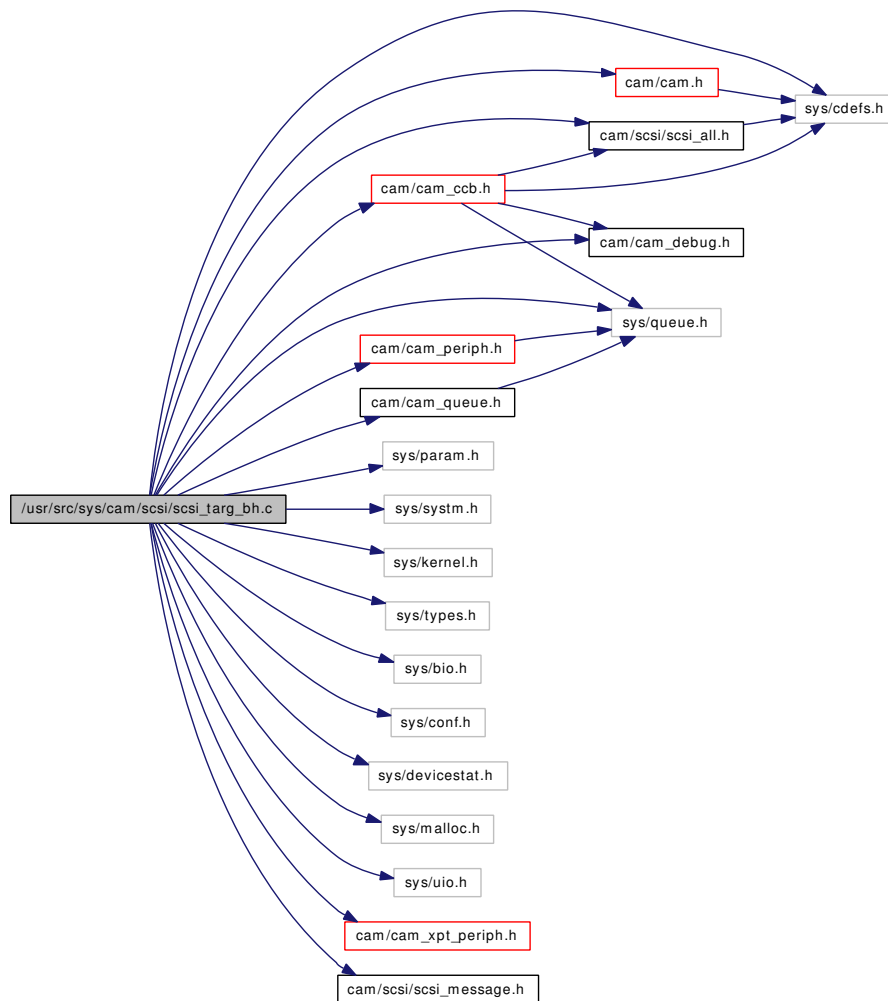
7.38.2.1 `typedef unsigned char ses_encstat`

Definition at line 116 of file `scsi_ses.h`.

7.39 /usr/src/sys/cam/scsi/scsi_targ_bh.c File Reference

```
#include <sys/cdefs.h>
#include <sys/param.h>
#include <sys/queue.h>
#include <sys/system.h>
#include <sys/kernel.h>
#include <sys/types.h>
#include <sys/bio.h>
#include <sys/conf.h>
#include <sys/devicestat.h>
#include <sys/malloc.h>
#include <sys/uio.h>
#include <cam/cam.h>
#include <cam/cam_ccb.h>
#include <cam/cam_periph.h>
#include <cam/cam_queue.h>
#include <cam/cam_xpt_periph.h>
#include <cam/cam_debug.h>
#include <cam/scsi/scsi_all.h>
#include <cam/scsi/scsi_message.h>
```

Include dependency graph for scsi_targ_bh.c:



Data Structures

- struct [targbh_softc](#)
- struct [targbh_cmd_desc](#)

Defines

- #define [MAX_ACCEPT](#) 8
- #define [MAX_IMMEDIATE](#) 16
- #define [MAX_BUF_SIZE](#) 256
- #define [ccb_type](#) ppriv_field0
- #define [ccb_descr](#) ppriv_ptr1
- #define [ccb_atio](#) ppriv_ptr1

Enumerations

- enum `targbh_state` { `TARGBH_STATE_NORMAL`, `TARGBH_STATE_EXCEPTION`, `TARGBH_STATE_TEARDOWN` }
- enum `targbh_flags` { `TARGBH_FLAG_NONE` = 0x00, `TARGBH_FLAG_LUN_ENABLED` = 0x01 }
- enum `targbh_ccb_types` { `TARGBH_CCB_WORKQ`, `TARGBH_CCB_WAITING` }

Functions

- `__FBSDID` ("\$FreeBSD: src/sys/cam/scsi/scsi_targ_bh.c,v 1.23 2006/12/05 07:45:28 mjacob Exp \$")
- `MALLOC_DEFINE` (M_SCSIBH, "SCSI bh", "SCSI blackhole buffers")
- `TAILQ_HEAD` (ccb_queue, ccb_hdr)
- static void `targbhasync` (void *callback_arg, u_int32_t code, struct `cam_path` *path, void *arg)
- static `cam_status` `targbhenlun` (struct `cam_periph` *periph)
- static `cam_status` `targbhdislun` (struct `cam_periph` *periph)
- static void `targbhdone` (struct `cam_periph` *periph, union `ccb` *done_ccb)
- static struct `targbh_cmd_desc` * `targbhallocdescr` (void)
- static void `targbhfreedescr` (struct `targbh_cmd_desc` *buf)
- `PERIPHDRIVER_DECLARE` (targbh, targbhdriver)
- static void `targbhinit` (void)
- static `cam_status` `targbhctor` (struct `cam_periph` *periph, void *arg)
- static void `targbhdtor` (struct `cam_periph` *periph)
- static void `targbhstart` (struct `cam_periph` *periph, union `ccb` *start_ccb)

Variables

- static struct `scsi_inquiry_data` `no_lun_inq_data`
- static struct `scsi_sense_data` `no_lun_sense_data`
- static const int `request_sense_size` = `offsetof(struct scsi_sense_data, fru)`
- static `periph_init_t` `targbhinit`
- static `periph_ctor_t` `targbhctor`
- static `periph_dtor_t` `targbhdtor`
- static `periph_start_t` `targbhstart`
- static struct `periph_driver` `targbhdriver`

7.39.1 Define Documentation

7.39.1.1 #define ccb_atio ppriv_ptr1

Definition at line 80 of file `scsi_targ_bh.c`.

7.39.1.2 #define ccb_descr ppriv_ptr1

Definition at line 77 of file `scsi_targ_bh.c`.

7.39.1.3 **#define ccb_type ppriv_field0**

Definition at line 76 of file scsi_targ_bh.c.

7.39.1.4 **#define MAX_ACCEPT 8**

Definition at line 71 of file scsi_targ_bh.c.

Referenced by targbhenlun().

7.39.1.5 **#define MAX_BUF_SIZE 256**

Definition at line 73 of file scsi_targ_bh.c.

Referenced by targbhallocdescr().

7.39.1.6 **#define MAX_IMMEDIATE 16**

Definition at line 72 of file scsi_targ_bh.c.

7.39.2 Enumeration Type Documentation

7.39.2.1 enum [targbh_ccb_types](#)

Enumerator:

TARGBH_CCB_WORKQ
TARGBH_CCB_WAITING

Definition at line 66 of file scsi_targ_bh.c.

7.39.2.2 enum [targbh_flags](#)

Enumerator:

TARGBH_FLAG_NONE
TARGBH_FLAG_LUN_ENABLED

Definition at line 61 of file scsi_targ_bh.c.

7.39.2.3 enum [targbh_state](#)

Enumerator:

TARGBH_STATE_NORMAL
TARGBH_STATE_EXCEPTION
TARGBH_STATE_TEARDOWN

Definition at line 55 of file scsi_targ_bh.c.

7.39.3 Function Documentation

7.39.3.1 `__FBSDID` ("\$FreeBSD: src/sys/cam/scsi/scsi_targ_bh.c, v 1.23 2006/12/05 07:45:28 mjacob Exp \$")

7.39.3.2 `MALLOC_DEFINE` (M_SCSIBH, "SCSI bh", "SCSI blackhole buffers")

7.39.3.3 `PERIPHDRIVER_DECLARE` (targbh, targbhdriver)

7.39.3.4 `TAILQ_HEAD` (ccb_queue, ccb_hdr)

7.39.3.5 `static struct targbh_cmd_desc * targbhallocdescr` (void) [static]

Definition at line 770 of file scsi_targ_bh.c.

References `MAX_BUF_SIZE`.

Referenced by `targbhenlun`().

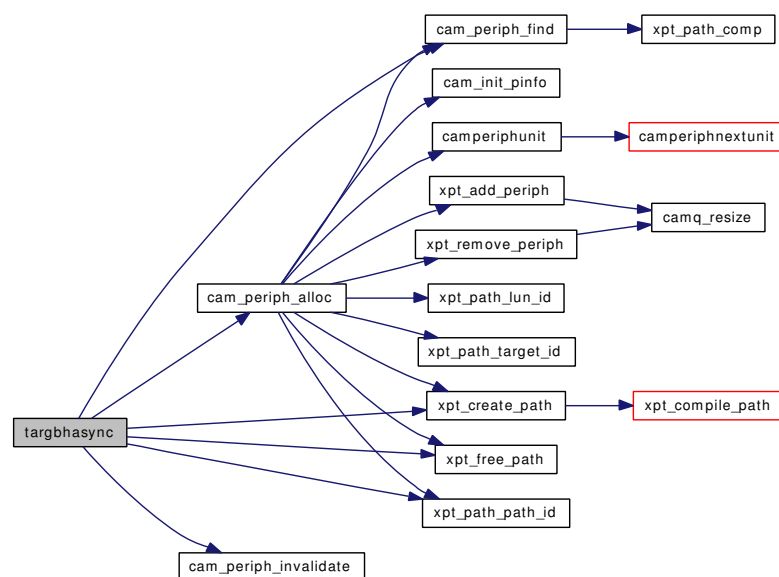
7.39.3.6 `static void targbhasync` (void * *callback_arg*, u_int32_t *code*, struct cam_path * *path*, void * *arg*) [static]

Definition at line 187 of file scsi_targ_bh.c.

References `AC_PATH_DEREGISTERED`, `AC_PATH_REGISTERED`, `CAM_LUN_WILDCARD`, `cam_periph_alloc`(), `CAM_PERIPH_BIO`, `cam_periph_find`(), `cam_periph_invalidate`(), `CAM_REQ_CMP`, `CAM_TARGET_WILDCARD`, `ccb_pathinq::ccb_h`, `ccb_hdr::path_id`, `PIT_PROCESSOR`, `ccb_pathinq::target_sprt`, `xpt_create_path`(), `xpt_free_path`(), and `xpt_path_path_id`().

Referenced by `targbhinit`().

Here is the call graph for this function:

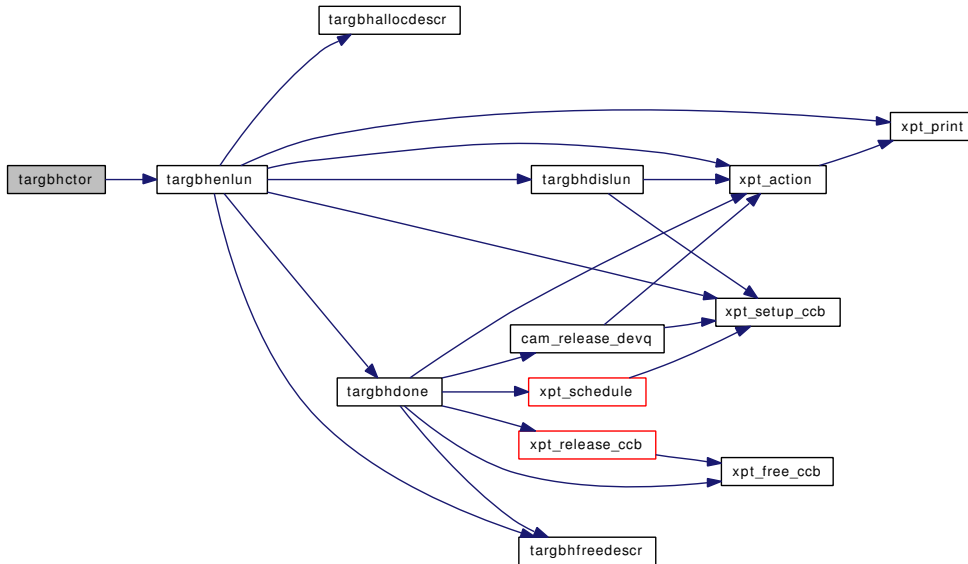


7.39.3.7 static `cam_status` targbhtor (`struct cam_periph * periph`, `void * arg`) [static]

Definition at line 408 of file `scsi_targ_bh.c`.

References `CAM_REQ_CMP_ERR`, `cam_periph::softc`, `TARGBH_STATE_NORMAL`, and `targbhenlun()`.

Here is the call graph for this function:



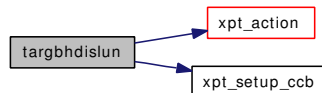
7.39.3.8 static `cam_status` targbhdiscun (`struct cam_periph * periph`) [static]

Definition at line 359 of file `scsi_targ_bh.c`.

References `ccb_abort::abort_ccb`, `targbh_softc::accept_tio_list`, `ccb::atio`, `ccb::cab`, `CAM_REQ_CMP`, `ccb_en_lun::ccb_h`, `ccb::ccb_h`, `ccb_abort::ccb_h`, `ccb_accept_tio::ccb_h`, `ccb::cel`, `ccb_en_lun::enable`, `targbh_softc::flags`, `ccb_hdr::func_code`, `targbh_softc::immed_notify_slist`, `cam_periph::path`, `cam_periph::softc`, `ccb_hdr::status`, `TARGBH_FLAG_LUN_ENABLED`, `XPT_ABORT`, `xpt_action()`, `XPT_EN_LUN`, and `xpt_setup_ccb()`.

Referenced by `targbhtor()`, and `targbhenlun()`.

Here is the call graph for this function:



7.39.3.9 static void targbhdone (`struct cam_periph * periph`, `union ccb * done_ccb`) [static]

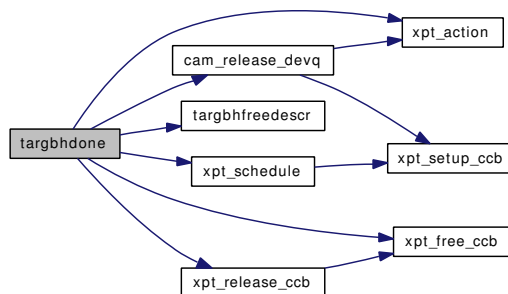
Definition at line 555 of file `scsi_targ_bh.c`.

References `ccb::atio`, `scsi_inquiry::byte2`, `CAM_DEBUG`, `CAM_DEBUG_SUBTRACE`, `CAM_DEV_-QFRZN`, `CAM_DIR_IN`, `CAM_DIR_MASK`, `CAM_DIR_NONE`, `CAM_DIS_DISCONNECT`, `CAM_-`

LUN_WILDCARD, cam_release_devq(), CAM_REQ_ABORTED, CAM_SEND_SENSE, CAM_SENT_SENSE, CAM_TARGET_WILDCARD, ccb_hdr::cbfcnp, ccb_accept_tio::ccb_h, ccb::ccb_h, cdb_t::cdb_bytes, ccb_accept_tio::cdb_io, targbh_cmd_desc::data, targbh_cmd_desc::data_increment, targbh_cmd_desc::data_resid, ccb_hdr::flags, ccb_hdr::func_code, INQUIRY, scsi_request_sense::length, scsi_inquiry::length, no_lun_inq_data, no_lun_sense_data, scsi_inquiry::page_code, ccb_hdr::path, cam_periph::path, targbh_softc::pending_queue, REQUEST_SENSE, SCSI_CDB6_LEN, SCSI_STATUS_CHECK_COND, SCSI_STATUS_OK, ccb_accept_tio::sense_data, ccb_accept_tio::sense_len, SI_EVPD, cam_periph::softc, targbh_softc::state, targbh_cmd_desc::status, ccb_hdr::status, TARGBH_CCB_WAITING, TARGBH_STATE_TEARDOWN, targbhfreedescri(), ccb_hdr::target_id, ccb_hdr::target_lun, targbh_cmd_desc::timeout, targbh_softc::work_queue, XPT_ACCEPT_TARGET_IO, xpt_action(), XPT_CONT_TARGET_IO, xpt_free_ccb(), XPT_IMMED_NOTIFY, xpt_release_ccb(), and xpt_schedule().

Referenced by targbhenlun(), and targbhstart().

Here is the call graph for this function:

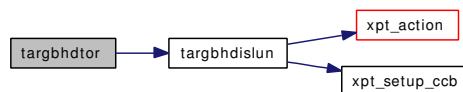


7.39.3.10 static void targbhdctor (struct cam_periph * periph) [static]

Definition at line 433 of file scsi_targ_bh.c.

References targbh_softc::init_level, cam_periph::softc, targbh_softc::state, TARGBH_STATE_TEARDOWN, and targbhdislun().

Here is the call graph for this function:



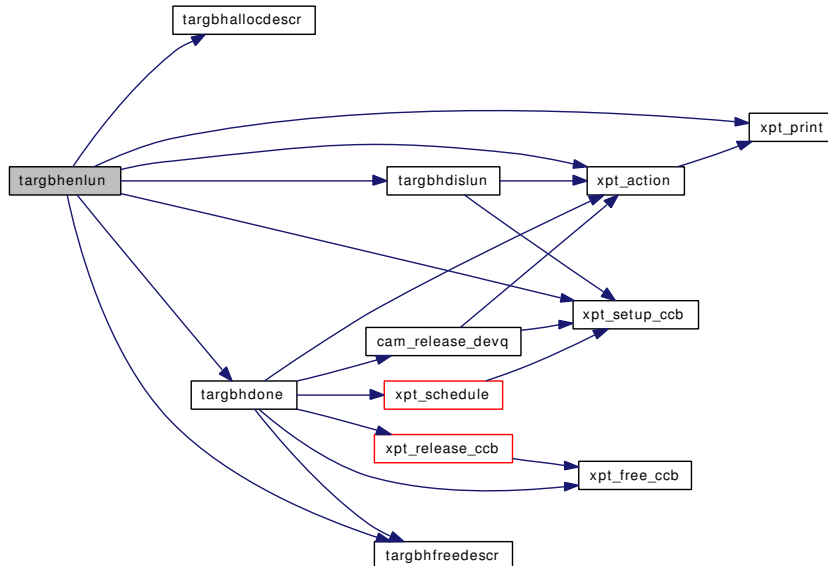
7.39.3.11 static cam_status targbhenlun (struct cam_periph * periph) [static]

Definition at line 244 of file scsi_targ_bh.c.

References targbh_softc::accept_tio_list, CAM_REQ_CMP, CAM_REQ_CMP_ERR, CAM_REQ_INPROG, CAM_RESRC_UNAVAIL, ccb::ccb_h, ccb::cel, ccb_en_lun::enable, targbh_softc::flags, ccb_hdr::func_code, ccb_en_lun::grp6_len, ccb_en_lun::grp7_len, targbh_softc::immed_notify_slist, MAX_ACCEPT, cam_periph::path, cam_periph::softc, ccb_hdr::status, TARGBH_FLAG_LUN_ENABLED, targbhallocdescr(), targbhdislun(), targbdone(), targbhfreedescri(), XPT_ACCEPT_TARGET_IO, xpt_action(), XPT_EN_LUN, XPT_IMMED_NOTIFY, xpt_print(), and xpt_setup_ccb().

Referenced by targbhctor().

Here is the call graph for this function:



7.39.3.12 static void targbhfreedescr (struct targbh_cmd_desc * buf) [static]

Definition at line 793 of file scsi_targ_bh.c.

References targbh_cmd_desc::backing_store.

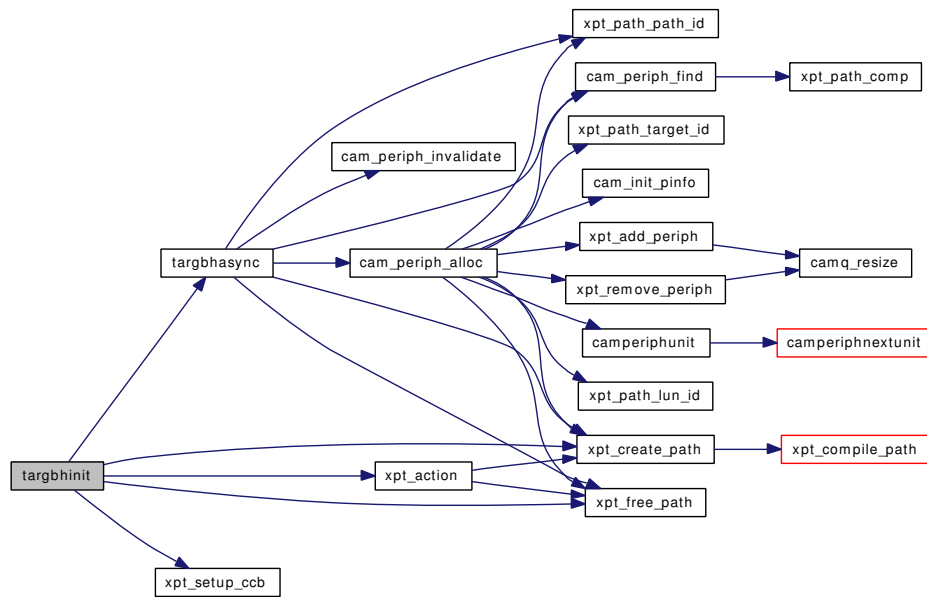
Referenced by targbhdone(), and targbhenlun().

7.39.3.13 static void targbhinit (void) [static]

Definition at line 155 of file scsi_targ_bh.c.

References AC_PATH_DEREGISTERED, AC_PATH_REGISTERED, ccb_setasync::callback, ccb_setasync::callback_arg, CAM_LUN_WILDCARD, CAM_REQ_CMP, CAM_TARGET_WILDCARD, CAM_XPT_PATH_ID, ccb::ccb_h, ccb_setasync::ccb_h, ccb_setasync::event_enable, ccb_hdr::func_code, ccb_hdr::status, targbhasync(), xpt_action(), xpt_create_path(), xpt_free_path(), XPT_SASYNC_CB, and xpt_setup_ccb().

Here is the call graph for this function:

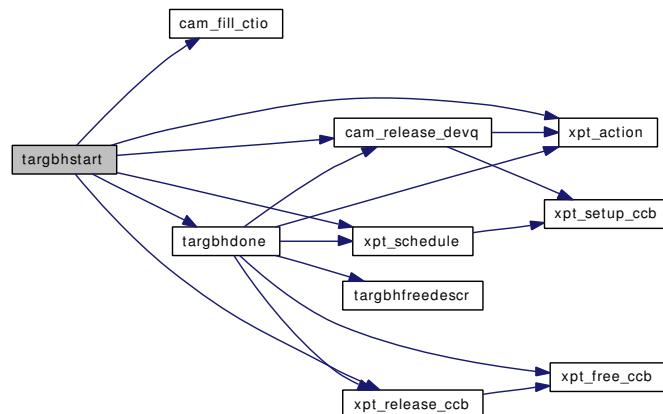


7.39.3.14 static void targbhstart (struct **cam_periph** * *periph*, union **ccb** * *start_ccb*) [static]

Definition at line 457 of file `scsi_targ_bh.c`.

References `CAM_DEBUG`, `CAM_DEBUG_SUBTRACE`, `CAM_DEV_QFRZN`, `CAM_DIR_MASK`, `CAM_DIS_DISCONNECT`, `cam_fill_ctio()`, `CAM_PRIORITY_NONE`, `cam_release_devq()`, `CAM_SEND_SENSE`, `CAM_SEND_STATUS`, `CAM_TAG_ACTION_VALID`, `ccb_accept_tio::ccb_h`, `ccb::ccb_h`, `ccb::csio`, `targbh_cmd_desc::data`, `targbh_cmd_desc::data_increment`, `targbh_cmd_desc::data_resid`, `ccb_hdr::flags`, `cam_periph::immediate_priority`, `ccb_accept_tio::init_id`, `MSG_SIMPLE_Q_TAG`, `cam_periph::path`, `targbh_softc::pending_queue`, `cam_periph::pinfo`, `cam_pinfo::priority`, `ccb_accept_tio::sense_data`, `ccb_scsiio::sense_data`, `ccb_scsiio::sense_len`, `ccb_accept_tio::sense_len`, `cam_periph::softc`, `ccb_hdr::status`, `targbh_cmd_desc::status`, `ccb_accept_tio::tag_id`, `TARGBH_CCB_WAITING`, `TARGBH_CCB_WORKQ`, `targbhdone()`, `ccb_hdr::target_id`, `ccb_hdr::target_lun`, `targbh_cmd_desc::timeout`, `targbh_softc::work_queue`, `xpt_action()`, `xpt_release_ccb()`, and `xpt_schedule()`.

Here is the call graph for this function:



7.39.4 Variable Documentation

7.39.4.1 struct `scsi_inquiry_data no_lun_inq_data` [static]

Initial value:

```
{
    T_NODEVICE | (SID_QUAL_BAD_LU << 5), 0,
    2, 2
}
```

Definition at line 108 of file `scsi_targ_bh.c`.

Referenced by `targbhdone()`.

7.39.4.2 struct `scsi_sense_data no_lun_sense_data` [static]

Initial value:

```
{
    SSD_CURRENT_ERROR|SSD_ERRCODE_VALID,
    0,
    SSD_KEY_NOT_READY,
    { 0, 0, 0, 0 },
    offsetof(struct scsi_sense_data, fru)
        - offsetof(struct scsi_sense_data, extra_len),
    { 0, 0, 0, 0 },
    0x25, 0
}
```

Definition at line 114 of file `scsi_targ_bh.c`.

Referenced by `targbhdone()`.

7.39.4.3 const int `request_sense_size = offsetof(struct scsi_sense_data, fru)` [static]

Definition at line 127 of file `scsi_targ_bh.c`.

7.39.4.4 `periph_ctor_t targbhctor` [static]

Definition at line 134 of file `scsi_targ_bh.c`.

7.39.4.5 struct `periph_driver targbhdriver` [static]

Initial value:

```
{
    targbhinit, "targbh",
    TAILQ_HEAD_INITIALIZER(targbhdriver.units), 0
}
```

Definition at line 146 of file `scsi_targ_bh.c`.

7.39.4.6 `periph_dtor_t targbhdctor` [static]

Definition at line 135 of file scsi_targ_bh.c.

7.39.4.7 `periph_init_t targbhinit` [static]

Definition at line 129 of file scsi_targ_bh.c.

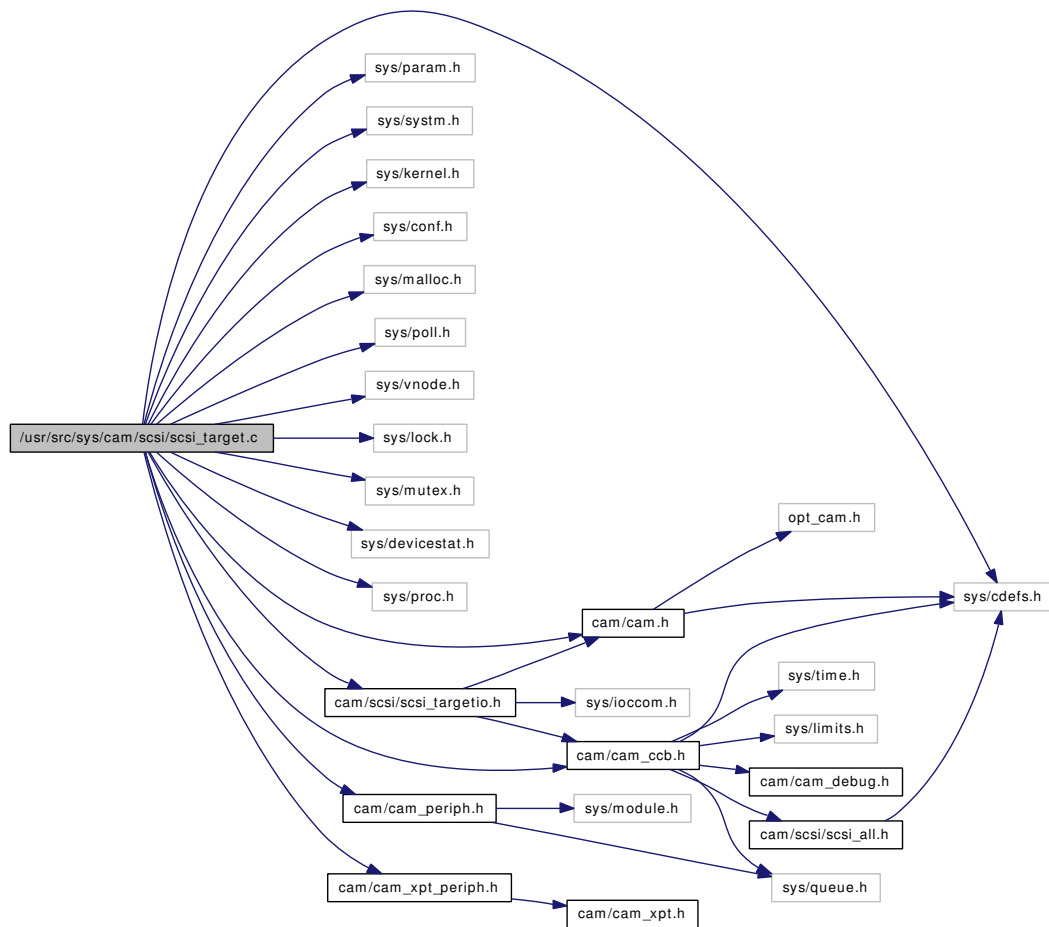
7.39.4.8 `periph_start_t targbhstart` [static]

Definition at line 136 of file scsi_targ_bh.c.

7.40 /usr/src/sys/cam/scsi/scsi_target.c File Reference

```
#include <sys/cdefs.h>
#include <sys/param.h>
#include <sys/system.h>
#include <sys/kernel.h>
#include <sys/conf.h>
#include <sys/malloc.h>
#include <sys/poll.h>
#include <sys/vnode.h>
#include <sys/lock.h>
#include <sys/mutex.h>
#include <sys/devicestat.h>
#include <sys/proc.h>
#include <cam/cam.h>
#include <cam/cam_ccb.h>
#include <cam/cam_periph.h>
#include <cam/cam_xpt_periph.h>
#include <cam/scsi/scsi_targetio.h>
```

Include dependency graph for scsi_target.c:



Data Structures

- struct [targ_cmd_descr](#)
- struct [targ_softc](#)

Defines

- #define [targ_descr](#) periph_priv.entries[1].ptr

Enumerations

- enum [targ_state](#) { [TARG_STATE_RESV](#) = 0x00, [TARG_STATE_OPENED](#) = 0x01, [TARG_STATE_LUN_ENABLED](#) = 0x02 }

Functions

- [__FBSDDID](#) ("\$FreeBSD: src/sys/cam/scsi/scsi_target.c,v 1.71 2006/12/05 07:45:28 mjacob Exp \$")
- [TAILQ_HEAD](#) (descr_queue, [targ_cmd_descr](#))
- static void [targreadfiltdetach](#) (struct knote *kn)

- static int `targreadfilt` (struct `knote *kn`, long `hint`)
- static `cam_status` `targendisln` (struct `cam_path *path`, int `enable`, int `grp6_len`, int `grp7_len`)
- static `cam_status` `targenable` (struct `targ_softc *softc`, struct `cam_path *path`, int `grp6_len`, int `grp7_len`)
- static `cam_status` `targdisable` (struct `targ_softc *softc`)
- static int `targusermerge` (struct `targ_softc *softc`, struct `targ_cmd_descr *descr`, union `ccb *ccb`)
- static int `targsendccb` (struct `targ_softc *softc`, union `ccb *ccb`, struct `targ_cmd_descr *descr`)
- static void `targdone` (struct `cam_periph *periph`, union `ccb *done_ccb`)
- static int `targreturnccb` (struct `targ_softc *softc`, union `ccb *ccb`)
- static union `ccb * targgetccb` (struct `targ_softc *softc`, `xpt_opcode` `type`, int `priority`)
- static void `targfreeccb` (struct `targ_softc *softc`, union `ccb *ccb`)
- static struct `targ_cmd_descr * targgetdescr` (struct `targ_softc *softc`)
- static void `targclone` (void `*arg`, struct `ucred *cred`, char `*name`, int `namelen`, struct `cdev **dev`)
- static void `targasync` (void `*callback_arg`, `u_int32_t` `code`, struct `cam_path *path`, void `*arg`)
- static void `abort_all_pending` (struct `targ_softc *softc`)
- static void `notify_user` (struct `targ_softc *softc`)
- static int `targcamstatus` (`cam_status` `status`)
- static `size_t` `targccblen` (`xpt_opcode` `func_code`)
- `PERIPHDRIIVER_DECLARE` (`targ`, `targdriver`)
- static `MALLOC_DEFINE` (`M_TARG`, "TARG", "TARG data")
- static int `targopen` (struct `cdev *dev`, int `flags`, int `fmt`, struct `thread *td`)
- static int `targclose` (struct `cdev *dev`, int `flag`, int `fmt`, struct `thread *td`)
- static int `targioctl` (struct `cdev *dev`, `u_long` `cmd`, `caddr_t` `addr`, int `flag`, struct `thread *td`)
- static int `targpoll` (struct `cdev *dev`, int `poll_events`, struct `thread *td`)
- static int `targkqfilter` (struct `cdev *dev`, struct `knote *kn`)
- static `cam_status` `targctor` (struct `cam_periph *periph`, void `*arg`)
- static void `targdtor` (struct `cam_periph *periph`)
- static int `targwrite` (struct `cdev *dev`, struct `uio *uio`, int `ioflag`)
- static void `targstart` (struct `cam_periph *periph`, union `ccb *start_ccb`)
- static int `targread` (struct `cdev *dev`, struct `uio *uio`, int `ioflag`)
- static void `targinit` (void)

Variables

- static `d_open_t` `targopen`
- static `d_close_t` `targclose`
- static `d_read_t` `targread`
- static `d_write_t` `targwrite`
- static `d_ioctl_t` `targioctl`
- static `d_poll_t` `targpoll`
- static `d_kqfilter_t` `targkqfilter`
- static struct `filterops` `targread_filtops`
- static struct `cdevsw` `targ_cdevsw`
- static `periph_ctor_t` `targctor`
- static `periph_dtor_t` `targdtor`
- static `periph_start_t` `targstart`
- static `periph_init_t` `targinit`
- static struct `periph_driver` `targdriver`

7.40.1 Define Documentation

7.40.1.1 #define targ_descr_periph_priv.entries[1].ptr

Definition at line 62 of file scsi_target.c.

7.40.2 Enumeration Type Documentation

7.40.2.1 enum targ_state

Enumerator:

TARG_STATE_RESV

TARG_STATE_OPENED

TARG_STATE_LUN_ENABLED

Definition at line 66 of file scsi_target.c.

7.40.3 Function Documentation

7.40.3.1 __FBSDID ("\$FreeBSD: src/sys/cam/scsi/scsi_target.c, v 1.71 2006/12/05 07:45:28 mjakob Exp \$")

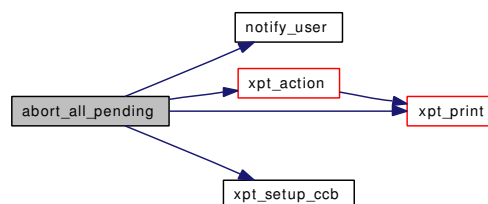
7.40.3.2 static void abort_all_pending (struct targ_softc *softc) [static]

Definition at line 1003 of file scsi_target.c.

References `ccb_abort::abort_ccb`, `targ_softc::abort_queue`, `ccb::cab`, `CAM_DEBUG`, `CAM_DEBUG_PERIPH`, `CAM_REQ_CMP`, `CAM_REQ_CMP_ERR`, `ccb::ccb_h`, `ccb_abort::ccb_h`, `ccb_hdr::func_code`, `notify_user()`, `targ_softc::path`, `targ_softc::pending_ccb_queue`, `ccb_hdr::periph_links`, `ccb_hdr::status`, `targ_softc::user_ccb_queue`, `targ_softc::work_queue`, `XPT_ABORT`, `xpt_action()`, `xpt_print()`, and `xpt_setup_ccb()`.

Referenced by `targdisable()`.

Here is the call graph for this function:



7.40.3.3 static MALLOC_DEFINE (M_TARG, "TARG", "TARG data") [static]

7.40.3.4 static void notify_user (struct targ_softc *softc) [static]

Definition at line 1052 of file scsi_target.c.

References `targ_softc::read_select`, and `targ_softc::user_ccb_queue`.

Referenced by `abort_all_pending()`, `targdone()`, and `targstart()`.

7.40.3.5 PERIPHDRIVER_DECLARE (`targ`, `targdriver`)

7.40.3.6 TAILQ_HEAD (`descr_queue`, `targ_cmd_descr`)

7.40.3.7 `static void targasync (void * callback_arg, u_int32_t code, struct cam_path * path, void * arg)` [static]

Definition at line 995 of file `scsi_target.c`.

Referenced by `targenable()`.

7.40.3.8 `static int targcamstatus (cam_status status)` [static]

Definition at line 1065 of file `scsi_target.c`.

References `CAM_BUSY`, `CAM_CMD_TIMEOUT`, `CAM_DEV_NOT_THERE`, `CAM_FUNC_NOTAVAIL`, `CAM_LUN_ALRDY_ENA`, `CAM_PATH_INVALID`, `CAM_PROVIDE_FAIL`, `CAM_REQ_ABORTED`, `CAM_REQ_CMP`, `CAM_REQ_CMP_ERR`, `CAM_REQ_INPROG`, `CAM_REQ_INVALID`, `CAM_QUEUE_REQ`, `CAM_RESRC_UNAVAIL`, `CAM_STATUS_MASK`, and `CAM_UA_ABORT`.

Referenced by `targioctl()`.

7.40.3.9 `static size_t targccblen (xpt_opcode func_code)` [static]

Definition at line 1102 of file `scsi_target.c`.

References `XPT_ABORT`, `XPT_ACCEPT_TARGET_IO`, `XPT_CONT_TARGET_IO`, `XPT_DEBUG`, `XPT_EN_LUN`, `XPT_IMMED_NOTIFY`, `XPT_PATH_INQ`, `XPT_REL_SIMQ`, and `XPT SCSI IO`.

Referenced by `targgetccb()`, `targreturnccb()`, and `targusermerge()`.

7.40.3.10 `static void targclone (void * arg, struct ucred * cred, char * name, int namelen, struct cdev ** dev)` [static]

Definition at line 979 of file `scsi_target.c`.

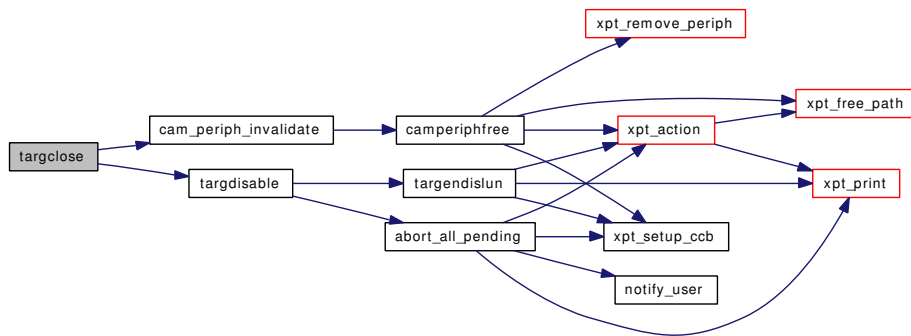
Referenced by `targinit()`.

7.40.3.11 `static int targclose (struct cdev * dev, int flag, int fmt, struct thread * td)` [static]

Definition at line 199 of file `scsi_target.c`.

References `cam_periph_invalidate()`, `CAM_REQ_CMP`, `targ_softc::periph`, and `targdisable()`.

Here is the call graph for this function:



7.40.3.12 static `cam_status` targetor (struct `cam_periph` * *periph*, void * *arg*) [static]

Definition at line 463 of file `scsi_target.c`.

References `CAM_REQ_CMP`, `cam_periph::path`, `targ_softc::path`, `targ_softc::periph`, and `cam_periph::softc`.

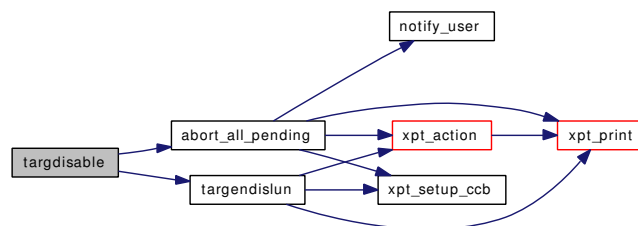
7.40.3.13 static `cam_status` targdisable (struct `targ_softc` * *softc*) [static]

Definition at line 438 of file `scsi_target.c`.

References `abort_all_pending()`, `CAM_DEBUG`, `CAM_DEBUG_PERIPH`, `CAM_REQ_CMP`, `targ_softc::path`, `targ_softc::state`, `TARG_STATE_LUN_ENABLED`, and `targendisln()`.

Referenced by `targetclose()`, and `targioctl()`.

Here is the call graph for this function:



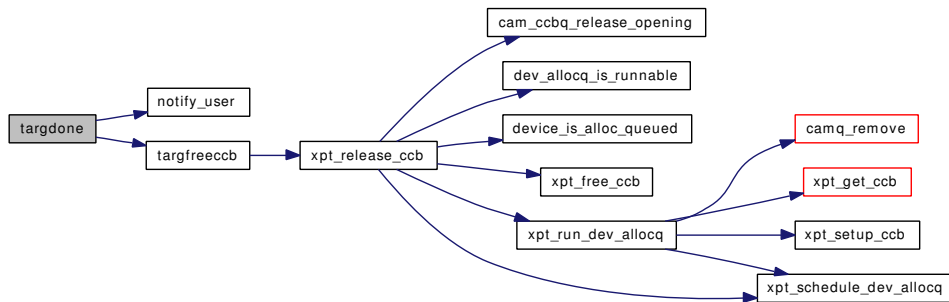
7.40.3.14 static void targdone (struct `cam_periph` * *periph*, union `ccb` * *done_ccb*) [static]

Definition at line 751 of file `scsi_target.c`.

References `CAM_DEBUG`, `CAM_DEBUG_PERIPH`, `CAM_STATUS_MASK`, `ccb::ccb_h`, `ccb_hdr::func_code`, `notify_user()`, `cam_periph::path`, `targ_softc::pending_ccb_queue`, `targ_softc::periph`, `cam_periph::softc`, `targ_softc::state`, `ccb_hdr::status`, `TARG_STATE_LUN_ENABLED`, `targfreeccb()`, `targ_softc::user_ccb_queue`, `XPT_ACCEPT_TARGET_IO`, `XPT_CONT_TARGET_IO`, and `XPT_IMMEDIATE_NOTIFY`.

Referenced by `targetgetccb()`, `targioctl()`, and `targsendccb()`.

Here is the call graph for this function:

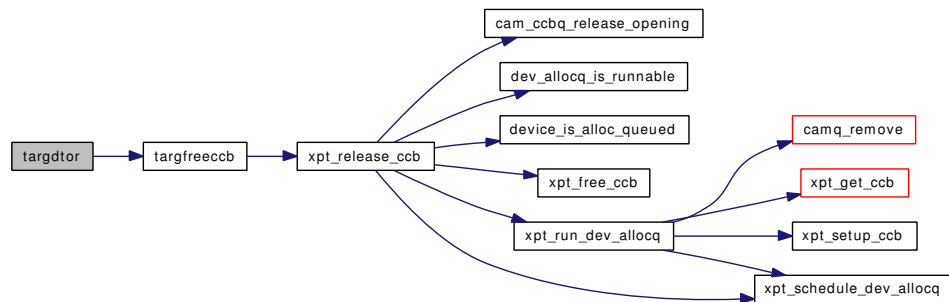


7.40.3.15 static void targdtor (struct cam_periph * periph) [static]

Definition at line 476 of file scsi_target.c.

References targ_softc::abort_queue, targ_softc::path, targ_softc::periph, cam_periph::softc, targfreccb(), and targ_softc::user_ccb_queue.

Here is the call graph for this function:



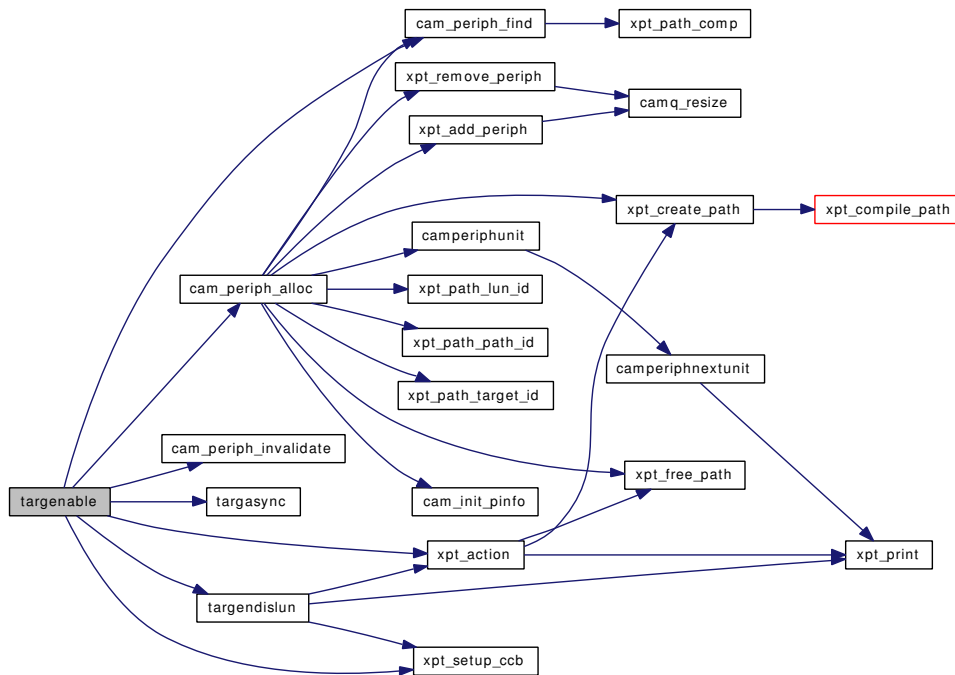
7.40.3.16 static cam_status targenable (struct targ_softc * softc, struct cam_path * path, int grp6_len, int grp7_len) [static]

Definition at line 368 of file scsi_target.c.

References CAM_FUNC_NOTAVAIL, CAM_LUN_ALRDY_ENA, cam_periph_alloc(), CAM_PERIPH_BIO, cam_periph_find(), cam_periph_invalidate(), CAM_REQ_CMP, CAM_STATUS_MASK, ccb::ccb_h, ccb_pathinq::ccb_h, ccb_hdr::func_code, targ_softc::path, targ_softc::periph, PIT_PROCESSOR, cam_periph::softc, targ_softc::state, ccb_hdr::status, TARG_STATE_LUN_ENABLED, targasync(), targctor, targdtor, targendisln(), targstart, cam_periph::unit_number, xpt_action(), XPT_PATH_INQ, and xpt_setup_ccb().

Referenced by targioctl().

Here is the call graph for this function:



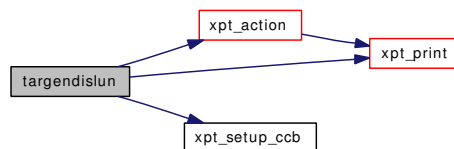
7.40.3.17 static `cam_status` targendisln (`struct cam_path * path`, `int enable`, `int grp6_len`, `int grp7_len`) [`static`]

Definition at line 345 of file `scsi_target.c`.

References `CAM_REQ_CMP`, `CAM_STATUS_MASK`, `ccb::ccb_h`, `ccb_en_lun::ccb_h`, `ccb_en_lun::enable`, `ccb_hdr::func_code`, `ccb_en_lun::grp6_len`, `ccb_en_lun::grp7_len`, `ccb_hdr::status`, `xpt_action()`, `XPT_EN_LUN`, `xpt_print()`, and `xpt_setup_ccb()`.

Referenced by `targdisable()`, and `targenable()`.

Here is the call graph for this function:



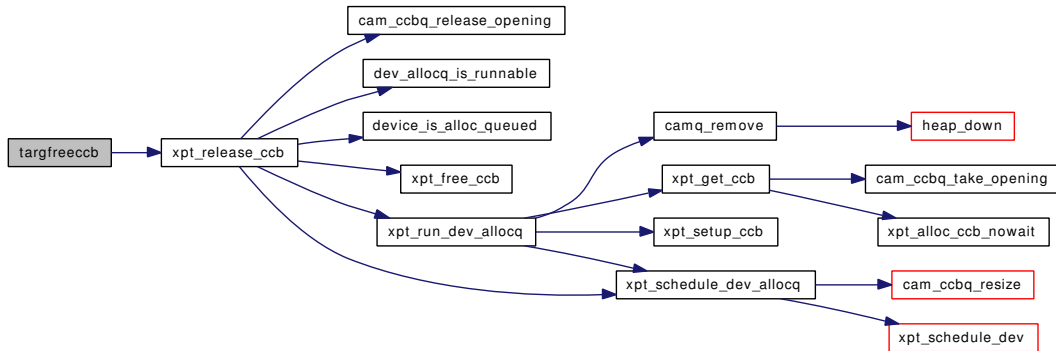
7.40.3.18 static void targfreeccb (`struct targ_softc * softc`, `union ccb * ccb`) [`static`]

Definition at line 934 of file `scsi_target.c`.

References `CAM_DEBUG_PERIPH`, `CAM_DEBUG_PRINT`, `ccb::ccb_h`, `ccb_hdr::func_code`, `XPT_ACCEPT_TARGET_IO`, `XPT_FC_IS_QUEUED`, `XPT_IMMED_NOTIFY`, and `xpt_release_ccb()`.

Referenced by `targdone()`, `targdtor()`, and `targreturnccb()`.

Here is the call graph for this function:



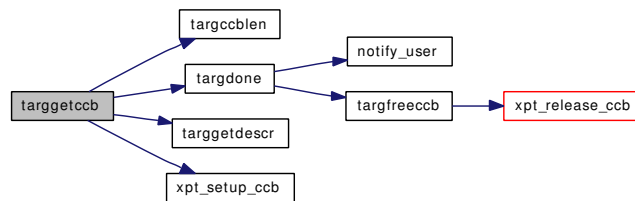
7.40.3.19 static union **ccb** * **targetccb** (struct **targ_softc** * **softc**, **xpt_opcode** type, int **priority**)
[static]

Definition at line 917 of file scsi_target.c.

References CAM_DEBUG, CAM_DEBUG_PERIPH, targ_softc::path, targccblen(), targdone(), target-descr(), and xpt_setup_ccb().

Referenced by targwrite().

Here is the call graph for this function:



7.40.3.20 static struct **targ_cmd_descr** * **targetdescr** (struct **targ_softc** * **softc**) [static]

Definition at line 962 of file scsi_target.c.

Referenced by targetccb(), and targwrite().

7.40.3.21 static void **targinit** (void) [static]

Definition at line 973 of file scsi_target.c.

References targclone().

Here is the call graph for this function:

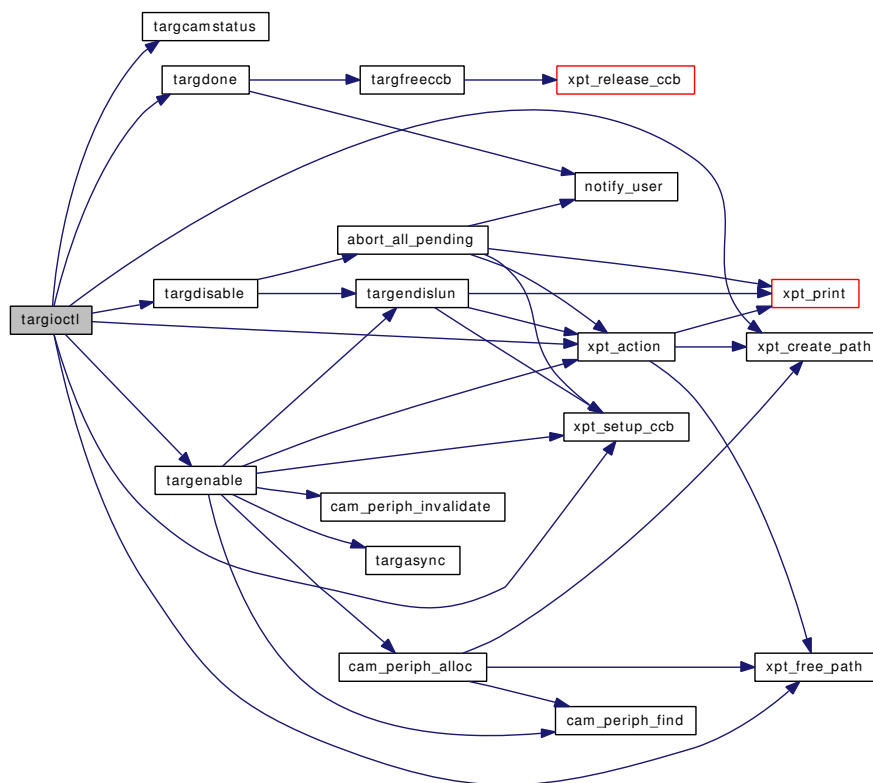


7.40.3.22 static int targioctl (struct cdev * *dev*, u_long *cmd*, caddr_t *addr*, int *flag*, struct thread * *td*) [static]

Definition at line 220 of file scsi_target.c.

References CAM_DEBUG_NONE, CAM_DEBUG_PERIPH, CAM_DEV_NOT_THERE, CAM_FUNC_NOTAVAIL, CAM_PROVIDE_FAIL, CAM_REQ_CMP, CAM_STATUS_MASK, ccb::ccb_h, ioc_enable_lun::grp6_len, ioc_enable_lun::grp7_len, ioc_enable_lun::lun_id, targ_softc::path, ioc_enable_lun::path_id, targ_softc::state, ccb_hdr::status, TARG_STATE_LUN_ENABLED, targcamstatus(), targdisable(), targdone(), targenable(), ioc_enable_lun::target_id, TARGIOCDEBUG, TARGIOCDISABLE, TARGIOCEENABLE, xpt_action(), xpt_create_path(), XPT_DEBUG, xpt_free_path(), and xpt_setup_ccb().

Here is the call graph for this function:



7.40.3.23 static int targkqfilter (struct cdev * *dev*, struct knote * *kn*) [static]

Definition at line 310 of file scsi_target.c.

References targ_softc::read_select.

7.40.3.24 static int targopen (struct cdev * *dev*, int *flags*, int *fmt*, struct thread * *td*) [static]

Definition at line 165 of file scsi_target.c.

References TARG_STATE_OPENED.

7.40.3.25 static int targpoll (struct cdev * dev, int poll_events, struct thread * td) [static]

Definition at line 286 of file scsi_target.c.

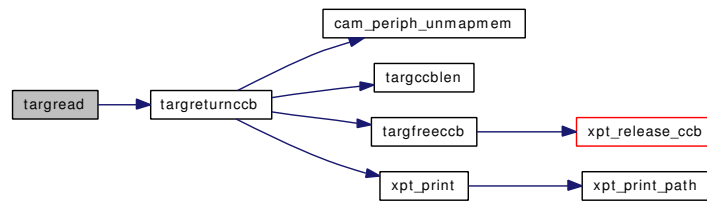
References targ_softc::abort_queue, targ_softc::read_select, and targ_softc::user_ccb_queue.

7.40.3.26 static int targread (struct cdev * dev, struct uio * uio, int ioflag) [static]

Definition at line 789 of file scsi_target.c.

References targ_softc::abort_queue, CAM_DEBUG, CAM_DEBUG_PERIPH, CAM_REQ_ABORTED, targ_softc::path, targreturnccb(), and targ_softc::user_ccb_queue.

Here is the call graph for this function:



7.40.3.27 static int targreadfilt (struct knote * kn, long hint) [static]

Definition at line 332 of file scsi_target.c.

References targ_softc::abort_queue, and targ_softc::user_ccb_queue.

7.40.3.28 static void targreadfiltdetach (struct knote * kn) [static]

Definition at line 322 of file scsi_target.c.

References targ_softc::read_select.

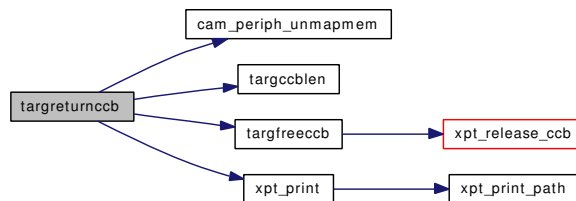
7.40.3.29 static int targreturnccb (struct targ_softc * softc, union ccb * ccb) [static]

Definition at line 885 of file scsi_target.c.

References CAM_DEBUG, CAM_DEBUG_PERIPH, cam_periph_unmapmem(), ccb::ccb_h, ccb_hdr::func_code, targ_cmd_descr::mapinfo, cam_periph_map_info::num_bufs_used, targ_softc::path, ccb_hdr::retry_count, targccbblen(), targfreeccb(), and xpt_print().

Referenced by targread(), and targwrite().

Here is the call graph for this function:



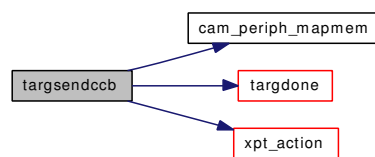
7.40.3.30 static int targsendccb (struct targ_softc * softc, union ccb * ccb, struct targ_cmd_descr * descr) [static]

Definition at line 686 of file scsi_target.c.

References CAM_DATA_PHYS, CAM_DEBUG, CAM_DEBUG_PERIPH, CAM_DIR_MASK, CAM_DIR_NONE, cam_periph_mapmem(), CAM_REQ_CMP_ERR, ccb_hdr::cbfcnp, ccb::ccb_h, ccb_hdr::flags, ccb_hdr::func_code, targ_cmd_descr::mapinfo, cam_periph_map_info::num_bufs_used, targ_softc::path, targ_softc::pending_ccb_queue, ccb_hdr::periph_links, ccb_hdr::status, targdone(), xpt_action(), XPT_CONT_TARGET_IO, XPT_DEV_MATCH, and XPT_FC_IS_QUEUED.

Referenced by targstart(), and targwrite().

Here is the call graph for this function:

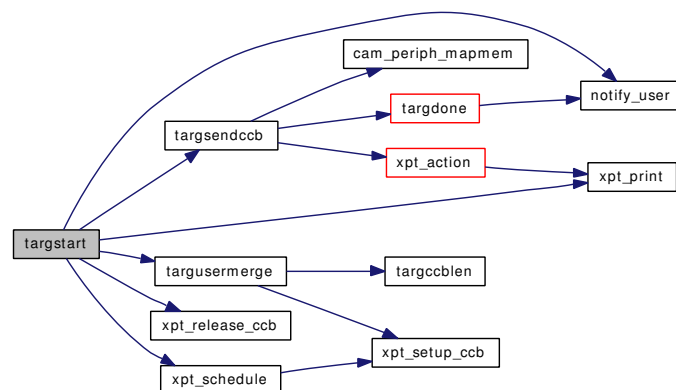


7.40.3.31 static void targstart (struct cam_periph * periph, union ccb * start_ccb) [static]

Definition at line 592 of file scsi_target.c.

References targ_softc::abort_queue, CAM_DEBUG, CAM_DEBUG_PERIPH, CAM_REQ_CMP_ERR, notify_user(), cam_periph::path, targ_softc::path, targ_softc::periph, cam_periph::softc, targsendccb(), targusermerge(), targ_softc::work_queue, xpt_print(), xpt_release_ccb(), and xpt_schedule().

Here is the call graph for this function:



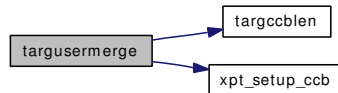
7.40.3.32 static int targusermerge (struct targ_softc * softc, struct targ_cmd_descr * descr, union ccb * ccb) [static]

Definition at line 629 of file scsi_target.c.

References `ccb_abort::abort_ccb`, `CAM_DEBUG`, `CAM_DEBUG_PERIPH`, `CAM_PATH_INVALID`, `CAM_REQ_CMP_ERR`, `ccb_abort::ccb_h`, `ccb::ccb_h`, `ccb_hdr::flags`, `ccb_hdr::func_code`, `targ_softc::path`, `targ_softc::pending_ccb_queue`, `ccb_hdr::retry_count`, `ccb_hdr::status`, `targccblen()`, `ccb_hdr::timeout`, `XPT_ABORT`, and `xpt_setup_ccb()`.

Referenced by `targstart()`, and `targwrite()`.

Here is the call graph for this function:

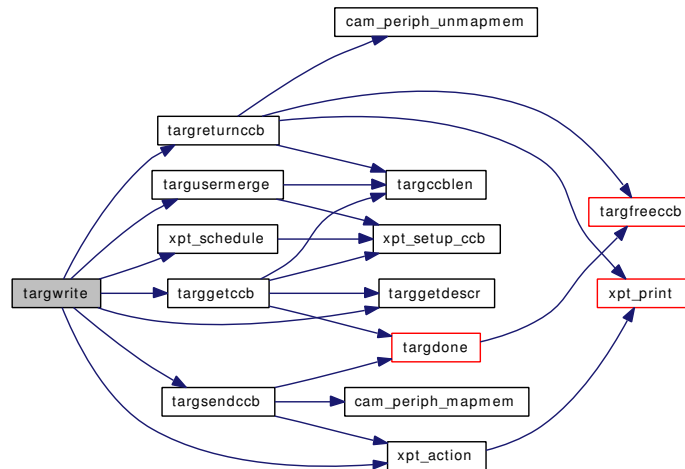


7.40.3.33 `static int targwrite (struct cdev * dev, struct uio * uio, int ioflag)` [static]

Definition at line 505 of file `scsi_target.c`.

References `CAM_DEBUG`, `CAM_DEBUG_PERIPH`, `ccb::ccb_h`, `targ_softc::path`, `targ_softc::pending_ccb_queue`, `targ_softc::periph`, `targgetccb()`, `targgetdescr()`, `targreturnccb()`, `targsendccb()`, `targusermerge()`, `targ_softc::work_queue`, `XPT_ACCEPT_TARGET_IO`, `xpt_action()`, `XPT_FC_QUEUED`, `XPT_IMMED_NOTIFY`, and `xpt_schedule()`.

Here is the call graph for this function:



7.40.4 Variable Documentation

7.40.4.1 `struct cdevsw targ_cdevsw` [static]

Initial value:

```

{
    .d_version =    D_VERSION,
    .d_flags =     D_NEEDGIANT,
    .d_open =      targopen,
    .d_close =     targclose,
    .d_read =      targread,
}
  
```



```
.d_write =      targwrite,  
.d_ioctl =     targioctl,  
.d_poll =      targpoll,  
.d_name =      "targ",  
.d_kqfilter =  targkqfilter  
}
```

Definition at line 108 of file scsi_target.c.

7.40.4.2 `d_close_t targclose` [static]

Definition at line 97 of file scsi_target.c.

7.40.4.3 `periph_ctor_t targetor` [static]

Definition at line 127 of file scsi_target.c.

Referenced by `targenable()`.

7.40.4.4 `struct periph_driver targdriver` [static]

Initial value:

```
{  
    targinit, "targ",  
    TAILQ_HEAD_INITIALIZER(targdriver.units), 0  
}
```

Definition at line 154 of file scsi_target.c.

7.40.4.5 `periph_dtor_t targdtor` [static]

Definition at line 128 of file scsi_target.c.

Referenced by `targenable()`.

7.40.4.6 `periph_init_t targinit` [static]

Definition at line 144 of file scsi_target.c.

7.40.4.7 `d_ioctl_t targioctl` [static]

Definition at line 100 of file scsi_target.c.

7.40.4.8 `d_kqfilter_t targkqfilter` [static]

Definition at line 102 of file scsi_target.c.

7.40.4.9 `d_open_t targopen` [static]

Definition at line 96 of file scsi_target.c.

7.40.4.10 d_poll_t targpoll [static]

Definition at line 101 of file scsi_target.c.

7.40.4.11 d_read_t targread [static]

Definition at line 98 of file scsi_target.c.

7.40.4.12 struct filterops targread_filtops [static]**Initial value:**

```
{ 1, NULL, targreadfiltdetach, targreadfilt }
```

Definition at line 105 of file scsi_target.c.

7.40.4.13 periph_start_t targstart [static]

Definition at line 129 of file scsi_target.c.

Referenced by targenable().

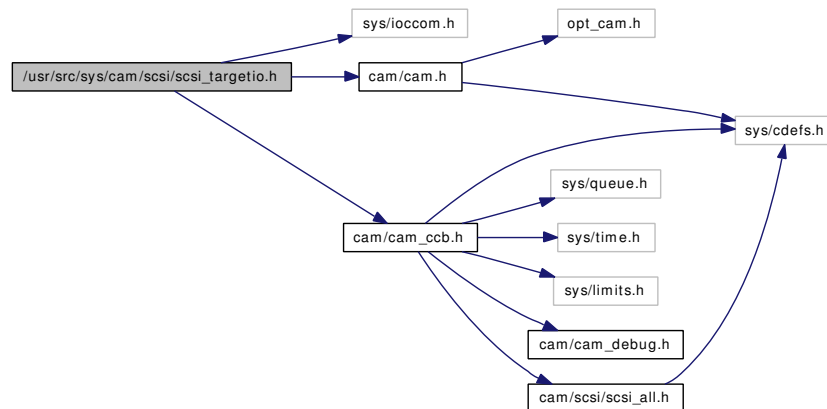
7.40.4.14 d_write_t targwrite [static]

Definition at line 99 of file scsi_target.c.

7.41 /usr/src/sys/cam/scsi/scsi_targetio.h File Reference

```
#include <sys/ioccom.h>
#include <cam/cam.h>
#include <cam/cam_ccb.h>
```

Include dependency graph for scsi_targetio.h:



This graph shows which files directly or indirectly include this file:



Data Structures

- struct [ioc_enable_lun](#)

Defines

- #define [TARGIOCENABLE_IOW](#)('C', 5, struct [ioc_enable_lun](#))
- #define [TARGIOCDISABLE_IO](#)('C', 6)
- #define [TARGIOCDEBUG_IOW](#)('C', 7, int)

Functions

- [TAILQ_HEAD](#) (ccb_queue, ccb_hdr)

7.41.1 Define Documentation

7.41.1.1 #define TARGIOCDEBUG_IOW('C', 7, int)

Definition at line 73 of file scsi_targetio.h.

Referenced by [targioctl\(\)](#).

7.41.1.2 #define TARGIOCDISABLE_IO('C', 6)

Definition at line 68 of file scsi_targetio.h.

Referenced by targioctl().

7.41.1.3 #define TARGIOCENABLE_IOW('C', 5, struct [ioc_enable_lun](#))

Definition at line 67 of file scsi_targetio.h.

Referenced by targioctl().

7.41.2 Function Documentation**7.41.2.1 TAILQ_HEAD (ccb_queue, [ccb_hdr](#))**

Index

- [/usr/ Directory Reference, 18](#)
- [/usr/src/ Directory Reference, 16](#)
- [/usr/src/sys/ Directory Reference, 17](#)
- [/usr/src/sys/cam/ Directory Reference, 13](#)
- [/usr/src/sys/cam/cam.c, 430](#)
- [/usr/src/sys/cam/cam.h, 434](#)
- [/usr/src/sys/cam/cam_ccb.h, 443](#)
- [/usr/src/sys/cam/cam_debug.h, 463](#)
- [/usr/src/sys/cam/cam_periph.c, 465](#)
- [/usr/src/sys/cam/cam_periph.h, 480](#)
- [/usr/src/sys/cam/cam_queue.c, 493](#)
- [/usr/src/sys/cam/cam_queue.h, 500](#)
- [/usr/src/sys/cam/cam_sim.c, 508](#)
- [/usr/src/sys/cam/cam_sim.h, 511](#)
- [/usr/src/sys/cam/cam_xpt.c, 515](#)
- [/usr/src/sys/cam/cam_xpt.h, 572](#)
- [/usr/src/sys/cam/cam_xpt_periph.h, 579](#)
- [/usr/src/sys/cam/cam_xpt_sim.h, 584](#)
- [/usr/src/sys/cam/scsi/ Directory Reference, 15](#)
- [/usr/src/sys/cam/scsi/scsi_all.c, 588](#)
- [/usr/src/sys/cam/scsi/scsi_all.h, 608](#)
- [/usr/src/sys/cam/scsi/scsi_cd.c, 658](#)
- [/usr/src/sys/cam/scsi/scsi_cd.h, 695](#)
- [/usr/src/sys/cam/scsi/scsi_ch.c, 718](#)
- [/usr/src/sys/cam/scsi/scsi_ch.h, 741](#)
- [/usr/src/sys/cam/scsi/scsi_da.c, 752](#)
- [/usr/src/sys/cam/scsi/scsi_da.h, 773](#)
- [/usr/src/sys/cam/scsi/scsi_dvcfg.h, 784](#)
- [/usr/src/sys/cam/scsi/scsi_iu.h, 786](#)
- [/usr/src/sys/cam/scsi/scsi_low.c, 789](#)
- [/usr/src/sys/cam/scsi/scsi_low.h, 830](#)
- [/usr/src/sys/cam/scsi/scsi_low_pisa.c, 867](#)
- [/usr/src/sys/cam/scsi/scsi_low_pisa.h, 869](#)
- [/usr/src/sys/cam/scsi/scsi_message.h, 870](#)
- [/usr/src/sys/cam/scsi/scsi_pass.c, 877](#)
- [/usr/src/sys/cam/scsi/scsi_pass.h, 888](#)
- [/usr/src/sys/cam/scsi/scsi_pt.c, 889](#)
- [/usr/src/sys/cam/scsi/scsi_pt.h, 901](#)
- [/usr/src/sys/cam/scsi/scsi_sa.c, 902](#)
- [/usr/src/sys/cam/scsi/scsi_sa.h, 935](#)
- [/usr/src/sys/cam/scsi/scsi_ses.c, 952](#)
- [/usr/src/sys/cam/scsi/scsi_ses.h, 985](#)
- [/usr/src/sys/cam/scsi/scsi_targ_bh.c, 993](#)
- [/usr/src/sys/cam/scsi/scsi_target.c, 1004](#)
- [/usr/src/sys/cam/scsi/scsi_targetio.h, 1019](#)
- [_CAM_CAM_CCB_H
cam_ccb.h, 449](#)
- [_CAM_CAM_DEBUG_H
cam_debug.h, 463](#)
- [_CAM_CAM_H
cam.h, 436](#)
- [_CAM_CAM_PERIPH_H
cam_periph.h, 482](#)
- [_CAM_CAM_QUEUE_H
cam_queue.h, 501](#)
- [_CAM_CAM_SIM_H
cam_sim.h, 511](#)
- [_CAM_CAM_XPT_H
cam_xpt.h, 572](#)
- [_CAM_CAM_XPT_PERIPH_H
cam_xpt_periph.h, 580](#)
- [_CAM_CAM_XPT_SIM_H
cam_xpt_sim.h, 584](#)
- [_SCSI_PASS_H
scsi_pass.h, 888](#)
- [_SCSI_SCSI_ALL_H
scsi_all.h, 617](#)
- [_SCSI_SCSI_CD_H
scsi_cd.h, 700](#)
- [_SCSI_SCSI_CH_H
scsi_ch.h, 743](#)
- [_SCSI_SCSI_DA_H
scsi_da.h, 775](#)
- [_SCSI_SCSI_IU_H
scsi_iu.h, 786](#)
- [_SCSI_SCSI_PT_H
scsi_pt.h, 901](#)
- [_SCSI_SCSI_SA_H
scsi_sa.h, 938](#)
- [_FBSDID
cam.c, 431
cam_periph.c, 468
cam_queue.c, 494
cam_sim.c, 509
cam_xpt.c, 526
scsi_all.c, 592
scsi_cd.c, 664
scsi_ch.c, 722
scsi_da.c, 757
scsi_low.c, 800](#)

- scsi_low_pisa.c, 868
- scsi_pass.c, 880
- scsi_pt.c, 892
- scsi_sa.c, 913
- scsi_ses.c, 966
- scsi_targ_bh.c, 997
- scsi_target.c, 1007
- __pad0__
 - sa_softc, 200
- _last_ctl_cdb
 - sa_softc, 200
- _last_ctl_resid
 - sa_softc, 200
- _last_ctl_sense
 - sa_softc, 200
- _last_io_cdb
 - sa_softc, 200
- _last_io_resid
 - sa_softc, 200
- _last_io_sense
 - sa_softc, 200
- A
 - scsi_all.c, 591
- abort_all_pending
 - scsi_target.c, 1007
- abort_ccb
 - ccb_abort, 52
- abort_queue
 - targ_softc, 417
- ABORTIO
 - scsi_low.h, 836
- AC_BUS_RESET
 - cam_ccb.h, 455
- ac_callback_t
 - cam_ccb.h, 455
- ac_code
 - cam_ccb.h, 455
- AC_FOUND_DEVICE
 - cam_ccb.h, 455
- AC_GETDEV_CHANGED
 - cam_ccb.h, 455
- AC_INQ_CHANGED
 - cam_ccb.h, 455
- AC_LOST_DEVICE
 - cam_ccb.h, 455
- AC_PATH_DEREGISTERED
 - cam_ccb.h, 455
- AC_PATH_REGISTERED
 - cam_ccb.h, 455
- AC_SCSI_AEN
 - cam_ccb.h, 455
- AC_SENT_BDR
 - cam_ccb.h, 455
- AC_TRANSFER_NEG
 - cam_ccb.h, 455
- AC_UNSOL_RESEL
 - cam_ccb.h, 455
- accept_tio_list
 - targbh_softc, 421
- action
 - asc_table_entry, 20
 - sense_key_table_entry, 393
- active_ccbs
 - cam_ccbq, 24
- active_dev
 - cam_devq, 26
- active_partition
 - scsi_dev_conf_page, 220
- add_sense_code
 - scsi_sense_data, 372
- add_sense_code_qual
 - scsi_sense_data, 372
- additional_length
 - scsi_inquiry_data, 230
- addr
 - scsi_read_capacity, 289
 - scsi_read_capacity_16, 290
 - scsi_read_capacity_data, 291
 - scsi_read_capacity_data_long, 292
 - scsi_rw_10, 356
 - scsi_rw_12, 358
 - scsi_rw_16, 360
 - scsi_rw_6, 362
 - scsi_verify, 388
 - scsi_write_and_verify, 390
- addr_0
 - scsi_read_cd_cap_data, 293
 - scsi_read_cd_capacity, 295
- addr_1
 - scsi_read_cd_cap_data, 293
 - scsi_read_cd_capacity, 295
- addr_2
 - scsi_read_cd_cap_data, 293
 - scsi_read_cd_capacity, 295
- addr_3
 - scsi_read_cd_cap_data, 293
 - scsi_read_cd_capacity, 295
- address
 - scsi_defect_desc_block, 217
 - scsi_read_dvd_structure, 324
- aen_holdoff_period
 - scsi_control_page, 211
- agid
 - scsi_read_dvd_structure, 324
 - scsi_report_key_data_agid, 338
- agid_keyformat
 - scsi_report_key, 336

- scsi_send_key, 365
- ALL
 - scsi_all.c, 591
- ALL_ENC_STAT
 - scsi_ses.c, 957
- alloc_active
 - cam_devq, 26
- alloc_len
 - cd_mode_params, 122
 - scsi_read_capacity_16, 290
 - scsi_read_dvd_structure, 324
 - scsi_report_key, 336
- alloc_length
 - scsi_read_defect_data_10, 297
 - scsi_read_defect_data_12, 298
 - scsi_read_format_capacities, 328
- alloc_openings
 - cam_devq, 26
- alloc_queue
 - cam_devq, 26
- allocated_supl
 - scsi_read_dvd_struct_data_spare_area, 323
- alt_sec_0
 - disk_pages::format_device_page, 150
- alt_sec_1
 - disk_pages::format_device_page, 150
- alt_trk_v_0
 - disk_pages::format_device_page, 150
- alt_trk_v_1
 - disk_pages::format_device_page, 150
- alt_trk_z_0
 - disk_pages::format_device_page, 151
- alt_trk_z_1
 - disk_pages::format_device_page, 151
- app_code
 - scsi_read_dvd_struct_data_lead_in, 314
- array_size
 - camq, 43
- asc
 - asc_key, 19
 - asc_table_entry, 20
- asc_info
 - scsi_sense_quirk_entry, 374
- asc_key, 19
 - asc, 19
 - ascq, 19
- asc_table
 - scsi_all.c, 604
- asc_table_entry, 20
 - action, 20
 - asc, 20
 - ascq, 20
 - desc, 20
- asc_table_size
 - scsi_all.c, 604
- ascentrycomp
 - scsi_all.c, 592
- ascq
 - asc_key, 19
 - asc_table_entry, 20
- async_flags
 - ccb_pathinq, 84
- async_node, 21
- atio
 - ccb, 47
- atio_link
 - targbh_cmd_desc, 419
- audio
 - cd_pages, 124
- AUDIO_PAGE
 - scsi_cd.h, 700
- avoltag
 - read_element_status_descriptor, 190
- backing_store
 - targbh_cmd_desc, 419
- base_transfer_speed
 - ccb_pathinq, 84
- bca
 - scsi_read_dvd_struct_data_layer_desc, 312
- bca_info
 - scsi_read_dvd_struct_data_bca, 301
- begin_lba
 - scsi_sync_cache, 381
- bio_queue
 - cd_softc, 127
 - da_softc, 138
 - pt_softc, 186
 - sa_softc, 200
- bitrate
 - ccb_pathinq_settings_fc, 88
 - ccb_pathinq_settings_sas, 89
 - ccb_trans_settings_fc, 111
 - ccb_trans_settings_sas, 112
- blk_addr
 - scsi_play_10, 276
 - scsi_play_12, 277
 - scsi_play_rel_12, 280
 - scsi_read_header, 329
- blk_desc
 - cd_mode_data, 119
 - cd_mode_data_10, 120
 - scsi_mode_sense_data, 270
- blk_desc_len
 - scsi_mode_header_10, 262
 - scsi_mode_header_6, 263
- blk_gran
 - sa_softc, 200

- blk_mask
 - sa_softc, 201
- blk_shift
 - sa_softc, 201
- blkaddr
 - scsi_tape_locate, 382
- blklen
 - scsi_mode_blk_desc, 258
- blkno
 - sa_softc, 201
- blksize
 - cd_params, 125
- block_descr_len
 - scsi_mode_hdr_10, 260
 - scsi_mode_hdr_6, 261
- block_len
 - scsi_mode_block_descr, 259
- block_length
 - format_capacity_descriptor, 159
- block_size
 - ccb_calc_geometry, 55
- book_type_version
 - scsi_read_dvd_struct_data_layer_desc, 312
- bp
 - cam_periph_map_info, 37
 - slccb, 406
- buffer_id
 - scsi_read_buffer, 288
 - scsi_write_buffer, 391
- buffer_mode
 - sa_softc, 201
- bus
 - cam_path, 31
 - ccb_dm_cookie, 61
- bus_id
 - bus_match_pattern, 22
 - bus_match_result, 23
 - cam_sim, 39
 - ccb_pathinq, 84
- BUS_MATCH_ANY
 - cam_ccb.h, 455
- BUS_MATCH_BUS_ID
 - cam_ccb.h, 455
- BUS_MATCH_NAME
 - cam_ccb.h, 455
- BUS_MATCH_NONE
 - cam_ccb.h, 455
- BUS_MATCH_PATH
 - cam_ccb.h, 455
- bus_match_pattern, 22
 - bus_id, 22
 - dev_name, 22
 - flags, 22
 - path_id, 22
 - unit_number, 22
- bus_match_result, 23
 - bus_id, 23
 - dev_name, 23
 - path_id, 23
 - unit_number, 23
- BUS_MATCH_UNIT
 - cam_ccb.h, 455
- bus_pattern
 - match_pattern, 172
- bus_pattern_flags
 - cam_ccb.h, 455
- bus_result
 - match_result, 173
- bus_width
 - ccb_trans_settings_spi, 114
- busses_to_config
 - cam_xpt.c, 567
- busses_to_reset
 - cam_xpt.c, 567
- BUSYERR
 - scsi_low.h, 836
- byte0
 - scsi_report_key_data_title, 345
- byte1
 - format_ipat_descriptor, 162
 - scsi_read_subchannel, 330
 - scsi_tape_locate, 382
 - scsi_tape_read_position, 386
- byte10
 - scsi_dev_conf_page, 220
- byte2
 - format_defect_list_header, 161
 - scsi_changedef, 209
 - scsi_dev_conf_page, 220
 - scsi_exchange_medium, 223
 - scsi_format_unit, 225
 - scsi_initialize_element_status, 227
 - scsi_inquiry, 228
 - scsi_log_select, 238
 - scsi_log_sense, 239
 - scsi_mode_select_10, 265
 - scsi_mode_select_6, 266
 - scsi_mode_sense_10, 267
 - scsi_mode_sense_6, 268
 - scsi_move_medium, 272
 - scsi_pause, 275
 - scsi_play_10, 276
 - scsi_play_12, 277
 - scsi_play_msf, 278
 - scsi_play_rel_12, 280
 - scsi_play_track, 281
 - scsi_position_to_element, 283
 - scsi_prevent, 285

- scsi_read_block_limits, 286
- scsi_read_buffer, 288
- scsi_read_capacity, 289
- scsi_read_cd_capacity, 295
- scsi_read_defect_data_10, 297
- scsi_read_defect_data_12, 298
- scsi_read_element_status, 326
- scsi_read_format_capacities, 328
- scsi_read_header, 329
- scsi_read_subchannel, 330
- scsi_read_toc, 332
- scsi_reassign_blocks, 333
- scsi_release, 335
- scsi_request_sense, 349
- scsi_request_volume_element_address, 350
- scsi_reserve, 352
- scsi_rezero_unit, 355
- scsi_rw_10, 356
- scsi_rw_12, 358
- scsi_rw_16, 360
- scsi_send_diag, 364
- scsi_send_receive, 367
- scsi_send_volume_tag, 368
- scsi_sense, 371
- scsi_set_speed, 376
- scsi_start_stop_unit, 378
- scsi_sync_cache, 381
- scsi_test_unit_ready, 387
- scsi_verify, 388
- scsi_write_and_verify, 390
- scsi_write_buffer, 391
- scsi_write_filemarks, 392
- byte3
 - scsi_da_rw_recovery_page, 213
- byte4
 - format_capacity_descriptor, 159
 - scsi_read_dvd_struct_data_copy_manage, 302
 - scsi_read_dvd_struct_data_medium_status, 318
 - scsi_report_key_data_rpc, 343
- byte8
 - scsi_dev_conf_page, 220
- bytes
 - ccb_priv_area, 92
 - ccb_priv_entry, 93
 - ccb_spriv_area, 106
 - scsi_generic, 226
- bytes_from_index
 - scsi_defect_desc_bytes_from_index, 218
- bytes_s_0
 - disk_pages::flexible_disk_page, 146
 - disk_pages::format_device_page, 151
- bytes_s_1
 - disk_pages::flexible_disk_page, 146
 - disk_pages::format_device_page, 151
- C
 - scsi_all.c, 591
- c
 - scsi_low.h, 865
- c_handle
 - cam_sim, 39
- cab
 - ccb, 47
- callback
 - ccb_setasync, 103
- callback_arg
 - ccb_setasync, 103
- CAM
 - scsi_low.h, 836
- cam.c
 - __FBSDDID, 431
 - cam_calc_geometry, 431
 - cam_error_print, 431
 - cam_error_string, 431
 - cam_fetch_status_entry, 432
 - cam_quirkmatch, 432
 - cam_status_table, 433
 - cam_strmatch, 432
 - cam_strvis, 432
 - camstatusentrycomp, 432
 - num_cam_status_entries, 433
 - SYSCTL_NODE, 433
- cam.h
 - _CAM_CAM_H, 436
 - CAM_ACTIVE_INDEX, 436
 - CAM_AUTOSENSE_FAIL, 439
 - CAM_AUTOSNS_VALID, 440
 - CAM_BDR_SENT, 439
 - CAM_BUS_WILDCARD, 436
 - CAM_BUSY, 439
 - CAM_CCB_LEN_ERR, 439
 - CAM_CDB_RECVD, 440
 - CAM_CMD_TIMEOUT, 439
 - CAM_DATA_RUN_ERR, 439
 - CAM_DEV_NOT_THERE, 439
 - CAM_DEV_QFRZN, 440
 - CAM_DONEQ_INDEX, 436
 - CAM_EPF_ALL, 438
 - CAM_EPF_LEVEL_MASK, 438
 - CAM_EPF_MINIMAL, 438
 - CAM_EPF_NONE, 438
 - CAM_EPF_NORMAL, 438
 - cam_error_print, 440
 - cam_error_proto_flags, 438
 - cam_error_scsi_flags, 438
 - cam_error_string, 440
 - cam_error_string_flags, 438

- CAM_ESF_ALL, 439
- CAM_ESF_CAM_STATUS, 438
- CAM_ESF_COMMAND, 438
- CAM_ESF_NONE, 438
- CAM_ESF_PRINT_NONE, 438
- CAM_ESF_PRINT_SENSE, 438
- CAM_ESF_PRINT_STATUS, 438
- CAM_ESF_PROTO_STATUS, 439
- CAM_EXPECT_INQ_CHANGE, 439
- cam_fetch_status_entry, 441
- CAM_FLAG_NONE, 439
- cam_flags, 439
- CAM_FUNC_NOTAVAIL, 440
- CAM_IDE, 440
- CAM_IID_INVALID, 440
- cam_init_pinfo, 441
- CAM_INVALID_CDB, 440
- CAM_LUN_ALRDY_ENA, 440
- CAM_LUN_INVALID, 440
- CAM_LUN_WILDCARD, 437
- CAM_MAX_CDBLEN, 437
- CAM_MESSAGE_RECV, 440
- CAM_MSG_REJECT_REC, 439
- CAM_NO_HBA, 439
- CAM_NO_NEXUS, 440
- CAM_PATH_INVALID, 439
- CAM_PRIORITY_NONE, 437
- CAM_PROVIDE_FAIL, 439
- cam_quirkmatch, 441
- cam_quirkmatch_t, 442
- CAM_RELEASE_SIMQ, 440
- CAM_REQ_ABORTED, 439
- CAM_REQ_CMP, 439
- CAM_REQ_CMP_ERR, 439
- CAM_REQ_INPROG, 439
- CAM_REQ_INVALID, 439
- CAM_REQ_TERMIO, 439
- CAM_REQ_TOO_BIG, 440
- CAM_REQUEUE_REQ, 440
- CAM_RESRC_UNAVAIL, 440
- CAM_RETRY_SELTO, 439
- CAM_SCSI_BUS_RESET, 439
- CAM_SCSI_BUSY, 440
- CAM_SCSI_STATUS_ERROR, 439
- CAM_SEL_TIMEOUT, 439
- CAM_SENT_SENSE, 440
- CAM_SEQUENCE_FAIL, 439
- CAM_SIM_QUEUED, 440
- cam_status, 439
- CAM_STATUS_MASK, 440
- cam_status_table, 442
- cam_strmatch, 442
- cam_strvis, 442
- CAM_TARGET_WILDCARD, 437
- CAM_TID_INVALID, 440
- CAM_UA_ABORT, 439
- CAM_UA_TERMIO, 439
- CAM_UNACKED_EVENT, 440
- CAM_UNCOR_PARITY, 439
- CAM_UNEXP_BUSFREE, 439
- CAM_UNQUEUED_INDEX, 437
- CAM_UNREC_HBA_ERROR, 440
- CAM_XPT_PATH_ID, 437
- GENERATIONCMP, 437
- lun_id_t, 438
- num_cam_status_entries, 442
- path_id_t, 438
- target_id_t, 438
- CAM_ACTIVE_INDEX
 - cam.h, 436
- CAM_AUTOSENSE_FAIL
 - cam.h, 439
- CAM_AUTOSNS_VALID
 - cam.h, 440
- CAM_BDR_SENT
 - cam.h, 439
- CAM_BUS_GENERATION
 - cam_ccb.h, 449
- CAM_BUS_WILDCARD
 - cam.h, 436
- CAM_BUSY
 - cam.h, 439
- cam_calc_geometry
 - cam.c, 431
 - cam_ccb.h, 461
- cam_ccb.h
 - AC_BUS_RESET, 455
 - AC_FOUND_DEVICE, 455
 - AC_GETDEV_CHANGED, 455
 - AC_INQ_CHANGED, 455
 - AC_LOST_DEVICE, 455
 - AC_PATH_DEREGISTERED, 455
 - AC_PATH_REGISTERED, 455
 - AC_SCSI_AEN, 455
 - AC_SENT_BDR, 455
 - AC_TRANSFER_NEG, 455
 - AC_UNSOL_RESEL, 455
 - BUS_MATCH_ANY, 455
 - BUS_MATCH_BUS_ID, 455
 - BUS_MATCH_NAME, 455
 - BUS_MATCH_NONE, 455
 - BUS_MATCH_PATH, 455
 - BUS_MATCH_UNIT, 455
 - CAM_CDB_LINKED, 456
 - CAM_CDB_PHYS, 457
 - CAM_CDB_POINTER, 456
 - CAM_DATA_PHYS, 457
 - CAM_DATAB_VALID, 457

- CAM_DEV_MATCH_ERROR, 456
- CAM_DEV_MATCH_LAST, 456
- CAM_DEV_MATCH_LIST_CHANGED, 456
- CAM_DEV_MATCH_MORE, 456
- CAM_DEV_MATCH_SIZE_ERROR, 456
- CAM_DEV_POS_BUS, 458
- CAM_DEV_POS_DEVICE, 458
- CAM_DEV_POS_EDT, 458
- CAM_DEV_POS_NONE, 458
- CAM_DEV_POS_PDPTR, 458
- CAM_DEV_POS_PDRV, 458
- CAM_DEV_POS_PERIPH, 458
- CAM_DEV_POS_TARGET, 458
- CAM_DEV_POS_TYEMASK, 458
- CAM_DEV_QFREEZE, 456
- CAM_DEV_QFRZDIS, 456
- CAM_DIR_IN, 456
- CAM_DIR_MASK, 456
- CAM_DIR_NONE, 456
- CAM_DIR_OUT, 456
- CAM_DIR_RESV, 456
- CAM_DIS_AUTODISC, 457
- CAM_DIS_AUTOSENSE, 456
- CAM_DIS_AUTOSRP, 457
- CAM_DIS_DISCONNECT, 457
- CAM_DISCONNECT, 457
- CAM_ENG_SGLIST, 457
- CAM_ENG_SYNC, 456
- CAM_GDEVLIST_ERROR, 457
- CAM_GDEVLIST_LAST_DEVICE, 457
- CAM_GDEVLIST_LIST_CHANGED, 457
- CAM_GDEVLIST_MORE_DEVS, 457
- CAM_HIGH_POWER, 457
- CAM_MSG_BUF_PHYS, 457
- CAM_MSGB_VALID, 457
- CAM_NEGOTIATE, 456
- CAM_PASS_ERR_RECOVER, 457
- CAM_QUEUE_ENABLE, 456
- CAM_SCATTER_VALID, 456
- CAM_SEND_SENSE, 457
- CAM_SEND_STATUS, 457
- CAM_SENSE_PHYS, 457
- CAM_SENSE_PTR, 457
- CAM_SG_LIST_PHYS, 457
- CAM_SNS_BUF_PHYS, 457
- CAM_SOFT_RST_OP, 456
- CAM_STATUS_VALID, 457
- CAM_TAG_ACTION_VALID, 457
- CAM_TERM_IO, 457
- CAM_TGT_CCB_AVAIL, 457
- CAM_TGT_PHASE_MODE, 457
- CTS_TYPE_CURRENT_SETTINGS, 457
- CTS_TYPE_USER_SETTINGS, 457
- DEV_MATCH_ANY, 458
- DEV_MATCH_BUS, 458
- DEV_MATCH_DEVICE, 458
- DEV_MATCH_INQUIRY, 458
- DEV_MATCH_LUN, 458
- DEV_MATCH_NONE, 458
- DEV_MATCH_PATH, 458
- DEV_MATCH_PERIPH, 458
- DEV_MATCH_TARGET, 458
- DEV_RESULT_NOFLAG, 458
- DEV_RESULT_UNCONFIGURED, 458
- EAD_LZ1V1, 459
- EAD_LZ2V1, 459
- EAD_LZ2V2, 459
- EAD_VUNIQUE, 459
- EIT_BUFFER, 459
- EIT_ENCRYPT, 459
- EIT_LOSSLESS, 459
- EIT_LOSSY, 459
- PERIPH_MATCH_ANY, 459
- PERIPH_MATCH_LUN, 459
- PERIPH_MATCH_NAME, 459
- PERIPH_MATCH_NONE, 459
- PERIPH_MATCH_PATH, 459
- PERIPH_MATCH_TARGET, 459
- PERIPH_MATCH_UNIT, 459
- PI_LINKED_CDB, 459
- PI_MDP_ABLE, 459
- PI_SDTR_ABLE, 459
- PI_SOFT_RST, 459
- PI_TAG_ABLE, 459
- PI_WIDE_16, 459
- PI_WIDE_32, 459
- PIM_NO_6_BYTE, 460
- PIM_NOBUSRESET, 460
- PIM_NOINITIATOR, 460
- PIM_NOREMOVE, 460
- PIM_SCANHILO, 460
- PIM_SEQSCAN, 460
- PIT_DISCONNECT, 460
- PIT_GRP_6, 460
- PIT_GRP_7, 460
- PIT_PHASE, 460
- PIT_PROCESSOR, 460
- PIT_TERM_IO, 460
- PROTO_ATA, 455
- PROTO_ATAPI, 455
- PROTO_SCSI, 455
- PROTO_UNKNOWN, 455
- PROTO_UNSPECIFIED, 455
- XPORT_ATA, 456
- XPORT_FC, 456
- XPORT_PPB, 456
- XPORT_SAS, 456

- XPORT_SPI, 456
- XPORT_SSA, 456
- XPORT_UNKNOWN, 456
- XPORT_UNSPECIFIED, 456
- XPORT_USB, 456
- XPT_ABORT, 461
- XPT_ACCEPT_TARGET_IO, 461
- XPT_CALC_GEOMETRY, 461
- XPT_CONT_TARGET_IO, 461
- XPT_DEBUG, 460
- XPT_DEV_MATCH, 460
- XPT_EN_LUN, 461
- XPT_ENG_EXEC, 461
- XPT_ENG_INQ, 461
- XPT_FC_DEV_QUEUED, 460
- XPT_FC_QUEUED, 460
- XPT_FC_USER_CCB, 460
- XPT_FC_XPT_ONLY, 460
- XPT_GDEV_STATS, 460
- XPT_GDEV_TYPE, 460
- XPT_GDEVLIST, 460
- XPT_GET_TRAN_SETTINGS, 461
- XPT_IMMED_NOTIFY, 461
- XPT_NOOP, 460
- XPT_NOTIFY_ACK, 461
- XPT_PATH_INQ, 460
- XPT_PATH_STATS, 460
- XPT_REL_SIMQ, 460
- XPT_RESET_BUS, 461
- XPT_RESET_DEV, 461
- XPT_SASync_CB, 460
- XPT_SCAN_BUS, 460
- XPT_SCAN_LUN, 461
- XPT_SCSI_IO, 460
- XPT_SDEV_TYPE, 460
- XPT_SET_TRAN_SETTINGS, 461
- XPT_TARGET_IO, 461
- XPT_TERM_IO, 461
- XPT_VUNIQUE, 461
- cam_ccb.h
 - _CAM_CAM_CCB_H, 449
 - ac_callback_t, 455
 - ac_code, 455
 - bus_pattern_flags, 455
 - CAM_BUS_GENERATION, 449
 - cam_calc_geometry, 461
 - CAM_DEV_GENERATION, 449
 - CAM_FAILURE, 449
 - CAM_FALSE, 449
 - cam_fill_csio, 461
 - cam_fill_ctio, 461
 - CAM_PERIPH_GENERATION, 449
 - cam_proto, 455
 - CAM_SUCCESS, 450
 - CAM_TAG_ACTION_NONE, 450
 - CAM_TARGET_GENERATION, 450
 - CAM_TIME_DEFAULT, 450
 - CAM_TIME_INFINITY, 450
 - CAM_TRUE, 450
 - CAM_VERSION, 450
 - cam_xport, 455
 - ccb_dev_match_status, 456
 - ccb_flags, 456
 - ccb_getdevlist_status_e, 457
 - CCB_PERIPH_PRIV_SIZE, 450
 - CCB_SIM_PRIV_SIZE, 450
 - CTS_FC_VALID_PORT, 450
 - CTS_FC_VALID_SPEED, 451
 - CTS_FC_VALID_WWNN, 451
 - CTS_FC_VALID_WWPN, 451
 - CTS_SAS_VALID_SPEED, 451
 - CTS_SCSI_FLAGS_TAG_ENB, 451
 - CTS_SCSI_VALID_TQ, 451
 - CTS_SPI_FLAGS_DISC_ENB, 451
 - CTS_SPI_VALID_BUS_WIDTH, 451
 - CTS_SPI_VALID_DISC, 451
 - CTS_SPI_VALID_PPR_OPTIONS, 452
 - CTS_SPI_VALID_SYNC_OFFSET, 452
 - CTS_SPI_VALID_SYNC_RATE, 452
 - cts_type, 457
 - DEV_IDLEN, 452
 - dev_match_type, 457
 - dev_pattern_flags, 458
 - dev_pos_type, 458
 - dev_result_flags, 458
 - ei_algo, 458
 - ei_type, 459
 - HBA_IDLEN, 452
 - IOCDBLEN, 452
 - PATHINQ_SETTINGS_SIZE, 452
 - periph_pattern_flags, 459
 - pi_inqflag, 459
 - pi_miscflag, 459
 - pi_tmflag, 460
 - PROTO_VERSION_UNKNOWN, 452
 - PROTO_VERSION_UNSPECIFIED, 452
 - RELSIM_ADJUST_OPENINGS, 453
 - RELSIM_RELEASE_AFTER_CMDCMPLT, 453
 - RELSIM_RELEASE_AFTER_QEMPTY, 453
 - RELSIM_RELEASE_AFTER_TIMEOUT, 453
 - SIM_IDLEN, 453
 - VUHBALEN, 453
 - XPORT_VERSION_UNKNOWN, 453
 - XPORT_VERSION_UNSPECIFIED, 453
 - XPT_CCB_INVALID, 453

- XPT_FC_GROUP, 454
- XPT_FC_GROUP_COMMON, 454
- XPT_FC_GROUP_HBA_ENGINE, 454
- XPT_FC_GROUP_MASK, 454
- XPT_FC_GROUP_SCSI_CONTROL, 454
- XPT_FC_GROUP_TMODE, 454
- XPT_FC_GROUP_VENDOR_UNIQUE, 454
- XPT_FC_IS_DEV_QUEUED, 454
- XPT_FC_IS_QUEUED, 454
- xpt_opcode, 460
- CAM_CCB_LEN_ERR
 - cam.h, 439
- cam_ccbq, 24
 - active_ccbs, 24
 - dev_active, 24
 - dev_openings, 24
 - devq_openings, 24
 - held, 25
 - queue, 25
- cam_ccbq_alloc
 - cam_queue.c, 494
 - cam_queue.h, 501
- cam_ccbq_ccb_done
 - cam_queue.h, 501
- cam_ccbq_fini
 - cam_queue.h, 502
- cam_ccbq_free
 - cam_queue.c, 494
 - cam_queue.h, 502
- cam_ccbq_init
 - cam_queue.c, 494
 - cam_queue.h, 502
- cam_ccbq_insert_ccb
 - cam_queue.h, 502
- cam_ccbq_peek_ccb
 - cam_queue.h, 502
- cam_ccbq_pending_ccb_count
 - cam_queue.h, 503
- cam_ccbq_release_opening
 - cam_queue.h, 503
- cam_ccbq_remove_ccb
 - cam_queue.h, 503
- cam_ccbq_resize
 - cam_queue.c, 495
 - cam_queue.h, 503
- cam_ccbq_send_ccb
 - cam_queue.h, 503
- cam_ccbq_take_opening
 - cam_queue.h, 504
- CAM_CDB_LINKED
 - cam_ccb.h, 456
- CAM_CDB_PHYS
 - cam_ccb.h, 457
- CAM_CDB_POINTER
 - cam_ccb.h, 456
- CAM_CDB_RECVD
 - cam.h, 440
- CAM_CMD_TIMEOUT
 - cam.h, 439
- CAM_DATA_PHYS
 - cam_ccb.h, 457
- CAM_DATA_RUN_ERR
 - cam.h, 439
- CAM_DATAB_VALID
 - cam_ccb.h, 457
- cam_dead_sim
 - cam_xpt.c, 567
- CAM_DEBUG
 - cam_debug.h, 463
- cam_debug.h
 - CAM_DEBUG_CDB, 464
 - CAM_DEBUG_INFO, 464
 - CAM_DEBUG_NONE, 464
 - CAM_DEBUG_PERIPH, 464
 - CAM_DEBUG_SUBTRACE, 464
 - CAM_DEBUG_TRACE, 464
 - CAM_DEBUG_XPT, 464
- cam_debug.h
 - _CAM_CAM_DEBUG_H, 463
 - CAM_DEBUG, 463
 - cam_debug_flags, 464
 - CAM_DEBUG_PRINT, 464
 - CAM_DEBUGGED, 464
- CAM_DEBUG_CDB
 - cam_debug.h, 464
- cam_debug_flags
 - cam_debug.h, 464
- CAM_DEBUG_INFO
 - cam_debug.h, 464
- CAM_DEBUG_NONE
 - cam_debug.h, 464
- CAM_DEBUG_PERIPH
 - cam_debug.h, 464
- CAM_DEBUG_PRINT
 - cam_debug.h, 464
- CAM_DEBUG_SUBTRACE
 - cam_debug.h, 464
- CAM_DEBUG_TRACE
 - cam_debug.h, 464
- CAM_DEBUG_XPT
 - cam_debug.h, 464
- CAM_DEBUGGED
 - cam_debug.h, 464
- CAM_DEV_DV_HIT_BOTTOM
 - cam_xpt.c, 522
- CAM_DEV_GENERATION
 - cam_ccb.h, 449
- CAM_DEV_IN_DV

- cam_xpt.c, 522
- CAM_DEV_INQUIRY_DATA_VALID
 - cam_xpt.c, 522
- CAM_DEV_MATCH_ERROR
 - cam_ccb.h, 456
- CAM_DEV_MATCH_LAST
 - cam_ccb.h, 456
- CAM_DEV_MATCH_LIST_CHANGED
 - cam_ccb.h, 456
- CAM_DEV_MATCH_MORE
 - cam_ccb.h, 456
- CAM_DEV_MATCH_SIZE_ERROR
 - cam_ccb.h, 456
- CAM_DEV_NOT_THERE
 - cam.h, 439
- CAM_DEV_POS_BUS
 - cam_ccb.h, 458
- CAM_DEV_POS_DEVICE
 - cam_ccb.h, 458
- CAM_DEV_POS_EDT
 - cam_ccb.h, 458
- CAM_DEV_POS_NONE
 - cam_ccb.h, 458
- CAM_DEV_POS_PDPTR
 - cam_ccb.h, 458
- CAM_DEV_POS_PDRV
 - cam_ccb.h, 458
- CAM_DEV_POS_PERIPH
 - cam_ccb.h, 458
- CAM_DEV_POS_TARGET
 - cam_ccb.h, 458
- CAM_DEV_POS_TYPEMASK
 - cam_ccb.h, 458
- CAM_DEV_QFREEZE
 - cam_ccb.h, 456
- CAM_DEV_QFRZDIS
 - cam_ccb.h, 456
- CAM_DEV_QFRZN
 - cam.h, 440
- CAM_DEV_REL_ON_COMPLETE
 - cam_xpt.c, 522
- CAM_DEV_REL_ON_QUEUE_EMPTY
 - cam_xpt.c, 522
- CAM_DEV_REL_TIMEOUT_PENDING
 - cam_xpt.c, 522
- CAM_DEV_RESIZE_QUEUE_NEEDED
 - cam_xpt.c, 522
- CAM_DEV_TAG_AFTER_COUNT
 - cam_xpt.c, 522
- CAM_DEV_UNCONFIGURED
 - cam_xpt.c, 522
- cam_devq, 26
 - active_dev, 26
 - alloc_active, 26
 - alloc_openings, 26
 - alloc_queue, 26
 - send_active, 27
 - send_openings, 27
 - send_queue, 27
- cam_devq_alloc
 - cam_queue.c, 495
 - cam_queue.h, 504
- cam_devq_free
 - cam_queue.c, 495
 - cam_queue.h, 504
- cam_devq_init
 - cam_queue.c, 495
 - cam_queue.h, 504
- cam_devq_resize
 - cam_queue.c, 496
 - cam_queue.h, 505
- CAM_DIR_IN
 - cam_ccb.h, 456
- CAM_DIR_MASK
 - cam_ccb.h, 456
- CAM_DIR_NONE
 - cam_ccb.h, 456
- CAM_DIR_OUT
 - cam_ccb.h, 456
- CAM_DIR_RESV
 - cam_ccb.h, 456
- CAM_DIS_AUTODISC
 - cam_ccb.h, 457
- CAM_DIS_AUTOSENSE
 - cam_ccb.h, 456
- CAM_DIS_AUTOSRP
 - cam_ccb.h, 457
- CAM_DIS_DISCONNECT
 - cam_ccb.h, 457
- CAM_DISCONNECT
 - cam_ccb.h, 457
- CAM_DONEQ_INDEX
 - cam.h, 436
- cam_eb, 28
- CAM_EB_RUNQ_SCHEDULED
 - cam_xpt.c, 522
- cam_ed, 29
- CAM_ENG_SGLIST
 - cam_ccb.h, 457
- CAM_ENG_SYNC
 - cam_ccb.h, 456
- CAM_EPF_ALL
 - cam.h, 438
- CAM_EPF_LEVEL_MASK
 - cam.h, 438
- CAM_EPF_MINIMAL
 - cam.h, 438
- CAM_EPF_NONE

- cam.h, 438
- CAM_EPF_NORMAL
 - cam.h, 438
- cam_error_print
 - cam.c, 431
 - cam.h, 440
- cam_error_proto_flags
 - cam.h, 438
- cam_error_scsi_flags
 - cam.h, 438
- cam_error_string
 - cam.c, 431
 - cam.h, 440
- cam_error_string_flags
 - cam.h, 438
- CAM_ESF_ALL
 - cam.h, 439
- CAM_ESF_CAM_STATUS
 - cam.h, 438
- CAM_ESF_COMMAND
 - cam.h, 438
- CAM_ESF_NONE
 - cam.h, 438
- CAM_ESF_PRINT_NONE
 - cam.h, 438
- CAM_ESF_PRINT_SENSE
 - cam.h, 438
- CAM_ESF_PRINT_STATUS
 - cam.h, 438
- CAM_ESF_PROTO_STATUS
 - cam.h, 439
- cam_et, 30
- CAM_EXPECT_INQ_CHANGE
 - cam.h, 439
- CAM_FAILURE
 - cam_ccb.h, 449
- CAM_FALSE
 - cam_ccb.h, 449
- cam_fetch_status_entry
 - cam.c, 432
 - cam.h, 441
- cam_fill_csio
 - cam_ccb.h, 461
- cam_fill_ctio
 - cam_ccb.h, 461
- CAM_FLAG_NONE
 - cam.h, 439
- cam_flags
 - cam.h, 439
- cam_freeze_devq
 - cam_periph.c, 468
 - cam_periph.h, 484
- CAM_FUNC_NOTAVAIL
 - cam.h, 440
- CAM_GDEVLIST_ERROR
 - cam_ccb.h, 457
- CAM_GDEVLIST_LAST_DEVICE
 - cam_ccb.h, 457
- CAM_GDEVLIST_LIST_CHANGED
 - cam_ccb.h, 457
- CAM_GDEVLIST_MORE_DEVS
 - cam_ccb.h, 457
- CAM_HIGH_POWER
 - cam_ccb.h, 457
- CAM_IDE
 - cam.h, 440
- CAM_IID_INVALID
 - cam.h, 440
- cam_init_pinfo
 - cam.h, 441
- CAM_INVALID_CDB
 - cam.h, 440
- CAM_LUN_ALRDY_ENA
 - cam.h, 440
- CAM_LUN_INVALID
 - cam.h, 440
- CAM_LUN_WILDCARD
 - cam.h, 437
- CAM_MAX_CDBLEN
 - cam.h, 437
- CAM_MAX_HIGHPOWER
 - cam_xpt.c, 523
- CAM_MESSAGE_RECV
 - cam.h, 440
- cam_module_event_handler
 - cam_xpt.c, 526
- CAM_MSG_BUF_PHYS
 - cam_ccb.h, 457
- CAM_MSG_REJECT_REC
 - cam.h, 439
- CAM_MSGB_VALID
 - cam_ccb.h, 457
- CAM_NEGOTIATE
 - cam_ccb.h, 456
- CAM_NO_HBA
 - cam.h, 439
- CAM_NO_NEXUS
 - cam.h, 440
- CAM_PASS_ERR_RECOVER
 - cam_ccb.h, 457
- cam_path, 31
 - bus, 31
 - device, 31
 - periph, 31
 - target, 32
- CAM_PATH_ANY
 - cam_sim.c, 509
- CAM_PATH_INVALID

- cam.h, 439
- cam_periph, 33
 - deferred_ac, 34
 - deferred_callback, 34
 - flags, 34
 - immediate_priority, 34
 - path, 34
 - periph_dtor, 34
 - periph_name, 35
 - periph_oninval, 35
 - periph_start, 35
 - pinfo, 35
 - refcount, 35
 - SLIST_ENTRY, 34
 - SLIST_HEAD, 34
 - softc, 35
 - TAILQ_ENTRY, 34
 - type, 36
 - unit_number, 36
- cam_periph.c
 - __FBSDID, 468
 - cam_freeze_devq, 468
 - cam_periph_acquire, 468
 - cam_periph_alloc, 468
 - cam_periph_async, 469
 - cam_periph_bus_settle, 469
 - cam_periph_ccbwait, 470
 - cam_periph_error, 470
 - cam_periph_find, 471
 - cam_periph_freeze_after_event, 471
 - cam_periph_getccb, 471
 - cam_periph_invalidate, 472
 - cam_periph_ioctl, 472
 - cam_periph_lock, 473
 - cam_periph_mapmem, 473
 - cam_periph_release, 473
 - cam_periph_runccb, 474
 - cam_periph_unlock, 474
 - cam_periph_unmapmem, 475
 - cam_release_devq, 475
 - camperiphdone, 475
 - camperiphfree, 476
 - camperiphnextunit, 476
 - camperiphscsisenseerror, 477
 - camperiphscsisstatuserror, 477
 - camperiphunit, 478
 - MALLOC_DEFINE, 478
 - nperiph_drivers, 479
 - periph_busy_delay, 479
 - periph_drivers, 479
 - periph_noresrc_delay, 479
 - periph_selto_delay, 479
 - periphdriver_register, 478
 - saved_ccb_ptr, 468
 - TUNABLE_INT, 478, 479
- cam_periph.h
 - CAM_PERIPH_BIO, 484
- cam_periph.h
 - _CAM_CAM_PERIPH_H, 482
 - cam_freeze_devq, 484
 - cam_periph_acquire, 484
 - cam_periph_alloc, 484
 - cam_periph_async, 485
 - cam_periph_bus_settle, 486
 - cam_periph_ccbwait, 486
 - cam_periph_error, 486
 - cam_periph_find, 487
 - cam_periph_freeze_after_event, 487
 - cam_periph_getccb, 487
 - CAM_PERIPH_INVALID, 482
 - cam_periph_invalidate, 488
 - cam_periph_ioctl, 488
 - cam_periph_lock, 489
 - CAM_PERIPH_LOCK_WANTED, 482
 - CAM_PERIPH_LOCKED, 482
 - cam_periph_mapmem, 489
 - CAM_PERIPH_MAXMAPS, 482
 - CAM_PERIPH_NEW_DEV_FOUND, 482
 - CAM_PERIPH_RECOVERY_INPROG, 482
 - cam_periph_release, 489
 - cam_periph_runccb, 490
 - CAM_PERIPH_RUNNING, 482
 - cam_periph_type, 484
 - cam_periph_unlock, 490
 - cam_periph_unmapmem, 491
 - cam_release_devq, 491
 - periph_ctor_t, 483
 - periph_drivers, 491
 - periph_dtor_t, 483
 - periph_init_func_t, 483
 - periph_init_t, 484
 - periph_oninv_t, 484
 - periph_start_t, 484
 - PERIPHDRIVER_DECLARE, 482
 - periphdriver_register, 491
 - ppriv_field0, 483
 - ppriv_field1, 483
 - ppriv_ptr0, 483
 - ppriv_ptr1, 483
 - xpt_periph, 492
- cam_periph_acquire
 - cam_periph.c, 468
 - cam_periph.h, 484
- cam_periph_alloc
 - cam_periph.c, 468
 - cam_periph.h, 484
- cam_periph_async
 - cam_periph.c, 469

- cam_periph.h, 485
- CAM_PERIPH_BIO
 - cam_periph.h, 484
- cam_periph_bus_settle
 - cam_periph.c, 469
 - cam_periph.h, 486
- cam_periph_ccbwait
 - cam_periph.c, 470
 - cam_periph.h, 486
- cam_periph_error
 - cam_periph.c, 470
 - cam_periph.h, 486
- cam_periph_find
 - cam_periph.c, 471
 - cam_periph.h, 487
- cam_periph_freeze_after_event
 - cam_periph.c, 471
 - cam_periph.h, 487
- CAM_PERIPH_GENERATION
 - cam_ccb.h, 449
- cam_periph_getccb
 - cam_periph.c, 471
 - cam_periph.h, 487
- CAM_PERIPH_INVALID
 - cam_periph.h, 482
- cam_periph_invalidate
 - cam_periph.c, 472
 - cam_periph.h, 488
- cam_periph_ioctl
 - cam_periph.c, 472
 - cam_periph.h, 488
- cam_periph_lock
 - cam_periph.c, 473
 - cam_periph.h, 489
- CAM_PERIPH_LOCK_WANTED
 - cam_periph.h, 482
- CAM_PERIPH_LOCKED
 - cam_periph.h, 482
- cam_periph_map_info, 37
 - bp, 37
 - num_bufs_used, 37
- cam_periph_mapmem
 - cam_periph.c, 473
 - cam_periph.h, 489
- CAM_PERIPH_MAXMAPS
 - cam_periph.h, 482
- CAM_PERIPH_NEW_DEV_FOUND
 - cam_periph.h, 482
- CAM_PERIPH_RECOVERY_INPROG
 - cam_periph.h, 482
- cam_periph_release
 - cam_periph.c, 473
 - cam_periph.h, 489
- cam_periph_runccb
 - cam_periph.c, 474
 - cam_periph.h, 490
- CAM_PERIPH_RUNNING
 - cam_periph.h, 482
- cam_periph_type
 - cam_periph.h, 484
- cam_periph_unlock
 - cam_periph.c, 474
 - cam_periph.h, 490
- cam_periph_unmapmem
 - cam_periph.c, 475
 - cam_periph.h, 491
- cam_pinfo, 38
 - generation, 38
 - index, 38
 - priority, 38
- CAM_PRIORITY_NONE
 - cam.h, 437
- cam_proto
 - cam_ccb.h, 455
- CAM_PROVIDE_FAIL
 - cam.h, 439
- cam_queue.c
 - __FBSDID, 494
 - cam_ccbq_alloc, 494
 - cam_ccbq_free, 494
 - cam_ccbq_init, 494
 - cam_ccbq_resize, 495
 - cam_devq_alloc, 495
 - cam_devq_free, 495
 - cam_devq_init, 495
 - cam_devq_resize, 496
 - camq_alloc, 496
 - camq_change_priority, 496
 - camq_fini, 496
 - camq_free, 497
 - camq_init, 497
 - camq_insert, 497
 - camq_remove, 497
 - camq_resize, 498
 - heap_down, 498
 - heap_up, 498
 - MALLOC_DEFINE, 498, 499
 - queue_cmp, 499
 - swap, 499
- cam_queue.h
 - _CAM_CAM_QUEUE_H, 501
 - cam_ccbq_alloc, 501
 - cam_ccbq_ccb_done, 501
 - cam_ccbq_fini, 502
 - cam_ccbq_free, 502
 - cam_ccbq_init, 502
 - cam_ccbq_insert_ccb, 502
 - cam_ccbq_peek_ccb, 502

- cam_ccbq_pending_ccb_count, 503
- cam_ccbq_release_opening, 503
- cam_ccbq_remove_ccb, 503
- cam_ccbq_resize, 503
- cam_ccbq_send_ccb, 503
- cam_ccbq_take_opening, 504
- cam_devq_alloc, 504
- cam_devq_free, 504
- cam_devq_init, 504
- cam_devq_resize, 505
- camq_alloc, 505
- camq_change_priority, 505
- camq_fini, 505
- camq_free, 505
- CAMQ_GET_HEAD, 501
- CAMQ_HEAD, 501
- camq_init, 506
- camq_insert, 506
- camq_remove, 506
- camq_resize, 506
- LIST_HEAD, 507
- SLIST_HEAD, 507
- TAILQ_HEAD, 507
- CAM_QUEUE_ENABLE
 - cam_ccb.h, 456
- CAM_QUIRK_HILUNS
 - cam_xpt.c, 523
- CAM_QUIRK_NOHILUNS
 - cam_xpt.c, 523
- CAM_QUIRK_NOLUNS
 - cam_xpt.c, 523
- CAM_QUIRK_NO SERIAL
 - cam_xpt.c, 523
- cam_quirkmatch
 - cam.c, 432
 - cam.h, 441
- cam_quirkmatch_t
 - cam.h, 442
- cam_release_devq
 - cam_periph.c, 475
 - cam_periph.h, 491
- CAM_RELEASE_SIMQ
 - cam.h, 440
- CAM_REQ_ABORTED
 - cam.h, 439
- CAM_REQ_CMP
 - cam.h, 439
- CAM_REQ_CMP_ERR
 - cam.h, 439
- CAM_REQ_INPROG
 - cam.h, 439
- CAM_REQ_INVALID
 - cam.h, 439
- CAM_REQ_TERMIO
 - cam.h, 439
- CAM_REQ_TOO_BIG
 - cam.h, 440
- CAM_REQUEUE_REQ
 - cam.h, 440
- CAM_RESRC_UNAVAIL
 - cam.h, 440
- CAM_RETRY_SELTO
 - cam.h, 439
- CAM_SCATTER_VALID
 - cam_ccb.h, 456
- CAM SCSI2_MAXLUN
 - cam_xpt.c, 523
- CAM SCSI_BUS_RESET
 - cam.h, 439
- CAM SCSI_BUSY
 - cam.h, 440
- CAM SCSI_STATUS_ERROR
 - cam.h, 439
- CAM_SEL_TIMEOUT
 - cam.h, 439
- CAM_SEND_SENSE
 - cam_ccb.h, 457
- CAM_SEND_STATUS
 - cam_ccb.h, 457
- CAM_SENSE_PHYS
 - cam_ccb.h, 457
- CAM_SENSE_PTR
 - cam_ccb.h, 457
- CAM_SENT_SENSE
 - cam.h, 440
- CAM_SEQUENCE_FAIL
 - cam.h, 439
- CAM_SG_LIST_PHYS
 - cam_ccb.h, 457
- cam_sim, 39
 - bus_id, 39
 - c_handle, 39
 - devq, 39
 - flags, 40
 - max_dev_openings, 40
 - max_tagged_dev_openings, 40
 - path_id, 40
 - sim_action, 40
 - sim_name, 40
 - sim_poll, 40
 - softc, 40
 - unit_number, 41
- cam_sim.c
 - __FBSDID, 509
 - CAM_PATH_ANY, 509
 - cam_sim_alloc, 509
 - cam_sim_free, 509
 - cam_sim_set_path, 509

- cam_simq_alloc, [509](#)
- cam_simq_free, [510](#)
- MALLOC_DEFINE, [510](#)
- cam_sim.h
 - _CAM_CAM_SIM_H, [511](#)
 - cam_sim_alloc, [512](#)
 - cam_sim_bus, [512](#)
 - cam_sim_free, [513](#)
 - cam_sim_name, [513](#)
 - cam_sim_path, [513](#)
 - CAM_SIM_REL_TIMEOUT_PENDING, [511](#)
 - cam_sim_set_path, [513](#)
 - cam_sim_softc, [513](#)
 - cam_sim_unit, [513](#)
 - cam_simq_alloc, [513](#)
 - cam_simq_free, [514](#)
 - sim_action_func, [512](#)
 - sim_poll_func, [512](#)
 - spriv_field0, [512](#)
 - spriv_field1, [512](#)
 - spriv_ptr0, [512](#)
 - spriv_ptr1, [512](#)
- cam_sim_alloc
 - cam_sim.c, [509](#)
 - cam_sim.h, [512](#)
- cam_sim_bus
 - cam_sim.h, [512](#)
- cam_sim_free
 - cam_sim.c, [509](#)
 - cam_sim.h, [513](#)
- cam_sim_name
 - cam_sim.h, [513](#)
- cam_sim_path
 - cam_sim.h, [513](#)
- CAM_SIM_QUEUED
 - cam.h, [440](#)
- CAM_SIM_REL_TIMEOUT_PENDING
 - cam_sim.h, [511](#)
- cam_sim_set_path
 - cam_sim.c, [509](#)
 - cam_sim.h, [513](#)
- cam_sim_softc
 - cam_sim.h, [513](#)
- cam_sim_unit
 - cam_sim.h, [513](#)
- cam_simq_alloc
 - cam_sim.c, [509](#)
 - cam_sim.h, [513](#)
- cam_simq_free
 - cam_sim.c, [510](#)
 - cam_sim.h, [514](#)
- CAM_SNS_BUF_PHYS
 - cam_ccb.h, [457](#)
- CAM_SOFT_RST_OP
 - cam_ccb.h, [456](#)
- cam_srch_hi
 - cam_xpt.c, [567](#)
- cam_status
 - cam.h, [439](#)
- cam_status_entry, [42](#)
 - status_code, [42](#)
 - status_text, [42](#)
- CAM_STATUS_MASK
 - cam.h, [440](#)
- cam_status_table
 - cam.c, [433](#)
 - cam.h, [442](#)
- CAM_STATUS_VALID
 - cam_ccb.h, [457](#)
- cam_strmatch
 - cam.c, [432](#)
 - cam.h, [442](#)
- cam_strvis
 - cam.c, [432](#)
 - cam.h, [442](#)
- CAM_SUCCESS
 - cam_ccb.h, [450](#)
- CAM_TAG_ACTION_NONE
 - cam_ccb.h, [450](#)
- CAM_TAG_ACTION_VALID
 - cam_ccb.h, [457](#)
- CAM_TAG_DELAY_COUNT
 - cam_xpt.c, [523](#)
- CAM_TARGET_GENERATION
 - cam_ccb.h, [450](#)
- CAM_TARGET_WILDCARD
 - cam.h, [437](#)
- CAM_TERM_IO
 - cam_ccb.h, [457](#)
- CAM_TGT_CCB_AVAIL
 - cam_ccb.h, [457](#)
- CAM_TGT_PHASE_MODE
 - cam_ccb.h, [457](#)
- CAM_TID_INVALID
 - cam.h, [440](#)
- CAM_TIME_DEFAULT
 - cam_ccb.h, [450](#)
- CAM_TIME_INFINITY
 - cam_ccb.h, [450](#)
- CAM_TRUE
 - cam_ccb.h, [450](#)
- CAM_UA_ABORT
 - cam.h, [439](#)
- CAM_UA_TERMIO
 - cam.h, [439](#)
- CAM_UNACKED_EVENT
 - cam.h, [440](#)

- CAM_UNCOR_PARITY
 - cam.h, 439
- CAM_UNEXP_BUSFREE
 - cam.h, 439
- CAM_UNQUEUED_INDEX
 - cam.h, 437
- CAM_UNREC_HBA_ERROR
 - cam.h, 440
- CAM_VERSION
 - cam_ccb.h, 450
- cam_xport
 - cam_ccb.h, 455
- cam_xpt.c
 - DM_RET_ACTION_MASK, 525
 - DM_RET_COPY, 524
 - DM_RET_DESCEND, 525
 - DM_RET_ERROR, 525
 - DM_RET_FLAG_MASK, 524
 - DM_RET_NONE, 524
 - DM_RET_STOP, 524
 - PROBE_DV_EXIT, 525
 - PROBE_FULL_INQUIRY, 525
 - PROBE_INQUIRY, 525
 - PROBE_INQUIRY_BASIC_DV1, 525
 - PROBE_INQUIRY_BASIC_DV2, 525
 - PROBE_INQUIRY_CKSUM, 525
 - PROBE_MODE_SENSE, 525
 - PROBE_NO_ANNOUNCE, 525
 - PROBE_SERIAL_CKSUM, 525
 - PROBE_SERIAL_NUM, 525
 - PROBE_TUR, 525
 - PROBE_TUR_FOR_NEGOTIATION, 525
 - XPT_DEPTH_BUS, 525
 - XPT_DEPTH_DEVICE, 525
 - XPT_DEPTH_PERIPH, 525
 - XPT_DEPTH_TARGET, 525
 - XPT_FLAG_OPEN, 525
- cam_xpt.c
 - __FBSDID, 526
 - busses_to_config, 567
 - busses_to_reset, 567
 - cam_dead_sim, 567
 - CAM_DEV_DV_HIT_BOTTOM, 522
 - CAM_DEV_IN_DV, 522
 - CAM_DEV_INQUIRY_DATA_VALID, 522
 - CAM_DEV_REL_ON_COMPLETE, 522
 - CAM_DEV_REL_ON_QUEUE_EMPTY, 522
 - CAM_DEV_REL_TIMEOUT_PENDING, 522
 - CAM_DEV_RESIZE_QUEUE_NEEDED, 522
 - CAM_DEV_TAG_AFTER_COUNT, 522
 - CAM_DEV_UNCONFIGURED, 522
 - CAM_EB_RUNQ_SCHEDULED, 522
 - CAM_MAX_HIGHPOWER, 523
 - cam_module_event_handler, 526
 - CAM_QUIRK_HILUNS, 523
 - CAM_QUIRK_NOHILUNS, 523
 - CAM_QUIRK_NOLUNS, 523
 - CAM_QUIRK_NOSERIAL, 523
 - CAM_SCSI2_MAXLUN, 523
 - cam_srch_hi, 567
 - CAM_TAG_DELAY_COUNT, 523
 - camisr, 526
 - CAN_SRCH_HI_DENSE, 523
 - CAN_SRCH_HI_SPARSE, 523
 - dead_sim_action, 527
 - dead_sim_poll, 527
 - DECLARE_MODULE, 527
 - dev_allocq_is_runnable, 527
 - dev_match_ret, 524
 - device_is_alloc_queued, 527
 - device_is_send_queued, 527
 - MALLOC_DEFINE, 528
 - microp, 567
 - MODULE_VERSION, 528
 - periph_is_queued, 528
 - PERIPHDRIVER_DECLARE, 528
 - probe_action, 525
 - probe_driver, 567
 - probe_flags, 525
 - probe_periph_init, 528
 - probecleanup, 528
 - probedone, 528
 - proberegister, 529
 - proberequestbackoff, 529
 - proberequestdefaultnegotiation, 530
 - probeschedule, 530
 - probestart, 531
 - quantum, 568
 - samsung, 568
 - seagate, 568
 - SIM_DEAD, 524
 - SLIST_HEAD, 531
 - sony, 568
 - STAILQ_HEAD, 531
 - sysctl_cam_search_luns, 531
 - SYSCTL_PROC, 531
 - TAILQ_HEAD, 532
 - TUNABLE_INT, 532
 - west_digital, 568
 - xpt_action, 532
 - xpt_add_periph, 534
 - xpt_alloc_ccb, 534
 - xpt_alloc_ccb_nowait, 535
 - xpt_alloc_device, 535
 - xpt_alloc_target, 535

- xpt_announce_periph, 536
- xpt_async, 536
- xpt_async_bcast, 537
- xpt_bus_deregister, 537
- xpt_bus_register, 537
- xpt_busfunc_t, 524
- xpt_cdevsw, 568
- xpt_compile_path, 538
- xpt_config, 538
- xpt_config_hook, 568
- xpt_create_path, 539
- xpt_dev_async, 539
- xpt_dev_ccbq_resize, 540
- xpt_devicefunc_t, 524
- xpt_devise_transport, 540
- xpt_done, 541
- xpt_find_bus, 541
- xpt_find_device, 541
- xpt_find_quirk, 541
- xpt_find_target, 541
- xpt_finishconfig, 541
- xpt_flags, 525
- xpt_for_all_busses, 542
- xpt_for_all_devices, 542
- xpt_free_ccb, 542
- xpt_free_path, 543
- xpt_freeze_devq, 543
- xpt_freeze_simq, 543
- xpt_get_ccb, 543
- xpt_init, 543
- xpt_merge_ccb, 544
- xpt_path_comp, 544
- xpt_path_lun_id, 544
- xpt_path_path_id, 545
- xpt_path_periph, 545
- xpt_path_sim, 545
- xpt_path_string, 545
- xpt_path_target_id, 545
- xpt_pdrvfunc_t, 524
- xpt_periph_init, 545
- xpt_periphfunc_t, 524
- xpt_polled_action, 545
- xpt_print, 546
- xpt_print_path, 546
- xpt_quirk_table, 568
- xpt_quirk_table_size, 569
- xpt_release_bus, 546
- xpt_release_ccb, 546
- xpt_release_device, 547
- xpt_release_devq, 547
- xpt_release_devq_device, 548
- xpt_release_devq_timeout, 548, 569
- xpt_release_path, 548
- xpt_release_simq, 549
- xpt_release_simq_timeout, 549, 569
- xpt_release_target, 550
- xpt_remove_periph, 550
- xpt_rescan, 550
- xpt_run_dev_allocq, 550
- xpt_run_dev_sendq, 551
- xpt_scan_bus, 551
- xpt_scan_lun, 552
- xpt_schedule, 553
- xpt_schedule_dev, 553
- xpt_schedule_dev_allocq, 554
- xpt_schedule_dev_sendq, 554
- xpt_set_transfer_settings, 554
- xpt_setup_ccb, 555
- xpt_start_tags, 556
- xpt_targetfunc_t, 524
- xpt_toggle_tags, 556
- xpt_traverse_depth, 525
- xptaction, 557
- xptbusmatch, 557
- xptbustraverse, 557
- xptclose, 557, 569
- xptconfigbuscountfunc, 558, 569
- xptconfigfunc, 558, 569
- xptdefbusfunc, 558, 569
- xptdefdevicefunc, 559, 569
- xptdefperiphfunc, 559, 569
- xptdeftargetfunc, 559, 569
- xptdevicematch, 559
- xptdevicetraverse, 560
- xptdone, 560
- xptedtbusfunc, 560, 569
- xptedtdevicefunc, 560, 570
- xptedtmatch, 561
- xptedtperiphfunc, 561, 570
- xptedttargetfunc, 562, 570
- xptioctl, 562, 570
- xptnextfreepathid, 563
- xptopen, 563, 570
- xptpassannouncefunc, 563, 570
- xptpathid, 563
- xptpdperiphtraverse, 564
- xptpdrvtraverse, 564
- xptperiphlistmatch, 564
- xptperiphmatch, 564
- xptperiphtraverse, 565
- xptplistpdrvfunc, 565, 570
- xptplistperiphfunc, 565, 570
- xptpoll, 565
- xptregister, 565
- xptscandone, 566
- xptsetasyncbusfunc, 566, 570
- xptsetasynfunc, 566, 570
- xpttargettraverse, 567

- xsoftc, 570
- cam_xpt.h
 - _CAM_CAM_XPT_H, 572
 - xpt_action, 572
 - xpt_async, 574
 - xpt_create_path, 575
 - xpt_free_path, 575
 - xpt_merge_ccb, 576
 - xpt_path_comp, 576
 - xpt_path_lun_id, 576
 - xpt_path_path_id, 576
 - xpt_path_periph, 576
 - xpt_path_sim, 576
 - xpt_path_string, 577
 - xpt_path_target_id, 577
 - xpt_print, 577
 - xpt_print_path, 577
 - xpt_rescan, 577
 - xpt_setup_ccb, 578
- CAM_XPT_PATH_ID
 - cam.h, 437
- cam_xpt_periph.h
 - _CAM_CAM_XPT_PERIPH_H, 580
 - xpt_add_periph, 580
 - xpt_alloc_ccb, 580
 - xpt_alloc_ccb_nowait, 580
 - xpt_announce_periph, 580
 - xpt_free_ccb, 581
 - xpt_polled_action, 581
 - xpt_release_ccb, 581
 - xpt_remove_periph, 582
 - xpt_schedule, 582
- cam_xpt_sim.h
 - _CAM_CAM_XPT_SIM_H, 584
 - xpt_bus_deregister, 584
 - xpt_bus_register, 585
 - xpt_done, 585
 - xpt_freeze_devq, 586
 - xpt_freeze_simq, 586
 - xpt_release_devq, 586
 - xpt_release_simq, 586
- CAMGETPASSTHRU
 - scsi_pass.h, 888
- CAMIOCOMMAND
 - scsi_pass.h, 888
- camisr
 - cam_xpt.c, 526
- camperiphdone
 - cam_periph.c, 475
- camperiphfree
 - cam_periph.c, 476
- camperiphnextunit
 - cam_periph.c, 476
- camperiphscsisenseerror
 - cam_periph.c, 477
- camperiphscsistatuserror
 - cam_periph.c, 477
- camperiphunit
 - cam_periph.c, 478
- camq, 43
 - array_size, 43
 - entries, 43
 - generation, 43
 - qfrozen_cnt, 43
 - queue_array, 44
- camq_alloc
 - cam_queue.c, 496
 - cam_queue.h, 505
- camq_change_priority
 - cam_queue.c, 496
 - cam_queue.h, 505
- camq_entry, 45
 - LIST_ENTRY, 45
 - SLIST_ENTRY, 45
 - STAILQ_ENTRY, 45
 - TAILQ_ENTRY, 45
- camq_fini
 - cam_queue.c, 496
 - cam_queue.h, 505
- camq_free
 - cam_queue.c, 497
 - cam_queue.h, 505
- CAMQ_GET_HEAD
 - cam_queue.h, 501
- CAMQ_HEAD
 - cam_queue.h, 501
- camq_init
 - cam_queue.c, 497
 - cam_queue.h, 506
- camq_insert
 - cam_queue.c, 497
 - cam_queue.h, 506
- camq_remove
 - cam_queue.c, 497
 - cam_queue.h, 506
- camq_resize
 - cam_queue.c, 498
 - cam_queue.h, 506
- camstatusentrycomp
 - cam.c, 432
- CAN_ROTATE
 - scsi_ch.h, 743
- CAN_SRCH_HI_DENSE
 - cam_xpt.c, 523
- CAN_SRCH_HI_SPARSE
 - cam_xpt.c, 523
- cap
 - scsi_mode_sense_data, 270

- capacity_list_length
 - format_capacity_list_header, 160
- cbfcnp
 - ccb_hdr, 76
- ccb, 46
 - atio, 47
 - cab, 47
 - ccb_h, 47
 - ccg, 48
 - cdbg, 48
 - cdm, 48
 - cee, 48
 - cei, 48
 - cel, 48
 - cgd, 49
 - cgdl, 49
 - cgds, 49
 - cin, 49
 - cna, 49
 - cpi, 49
 - cpis, 49
 - crb, 49
 - crcn, 49
 - crd, 49
 - crs, 50
 - csa, 50
 - csd, 50
 - csio, 50
 - ctio, 50
 - cts, 50
 - tio, 50
- ccb_abort, 52
 - abort_ccb, 52
 - ccb_h, 52
- ccb_accept_tio, 53
 - ccb_h, 53
 - cdb_io, 53
 - cdb_len, 53
 - init_id, 54
 - sense_data, 54
 - sense_len, 54
 - tag_action, 54
 - tag_id, 54
- ccb_atio
 - scsi_targ_bh.c, 995
- CCB_AUTOSENSE
 - scsi_low.h, 836
- ccb_bp
 - scsi_cd.c, 662
 - scsi_ch.c, 721
 - scsi_da.c, 755
 - scsi_pass.c, 879
 - scsi_pt.c, 891
 - scsi_sa.c, 906
 - scsi_ses.c, 957
- ccb_calc_geometry, 55
 - block_size, 55
 - ccb_h, 55
 - cylinders, 55
 - heads, 56
 - secs_per_track, 56
 - volume_size, 56
- CCB_CLEARQ
 - scsi_low.h, 836
- ccb_dataalen
 - slccb, 406
- ccb_debug, 57
 - ccb_h, 57
 - flags, 57
- ccb_descr
 - scsi_targ_bh.c, 995
- ccb_dev_match, 58
 - ccb_h, 58
 - match_buf_len, 58
 - matches, 58
 - num_matches, 58
 - num_patterns, 59
 - pattern_buf_len, 59
 - patterns, 59
 - pos, 59
 - status, 59
- ccb_dev_match_status
 - cam_ccb.h, 456
- ccb_dev_position, 60
 - cookie, 60
 - generations, 60
 - position_type, 60
- CCB_DISCQ
 - scsi_low.h, 836
- ccb_dm_cookie, 61
 - bus, 61
 - device, 61
 - pdrv, 61
 - periph, 61
 - target, 61
- ccb_en_lun, 63
 - ccb_h, 63
 - enable, 63
 - grp6_len, 63
 - grp7_len, 64
- ccb_eng_exec, 65
 - ccb_h, 65
 - data_ptr, 65
 - dest_len, 66
 - dmax_len, 66
 - dxfer_len, 66
 - eng_num, 66
 - engdata_ptr, 66

- pdrv_ptr, 66
 - req_map, 66
 - sglist_cnt, 66
 - src_resid, 66
 - timeout, 66
 - vu_flags, 66
- ccb_eng_inq, 68
 - ccb_h, 68
 - eng_algo, 68
 - eng_memeory, 68
 - eng_num, 68
 - eng_type, 69
- ccb_error
 - slccb, 406
- ccb_flags
 - cam_ccb.h, 456
 - slccb, 406
- ccb_getdev, 70
 - ccb_h, 70
 - inq_data, 70
 - reserved, 70
 - serial_num, 71
 - serial_num_len, 71
- ccb_getdevlist, 72
 - ccb_h, 72
 - generation, 72
 - index, 72
 - periph_name, 73
 - status, 73
 - unit_number, 73
- ccb_getdevlist_status_e
 - cam_ccb.h, 457
- ccb_getdevstats, 74
 - ccb_h, 74
 - dev_active, 74
 - dev_openings, 74
 - devq_openings, 74
 - devq_queued, 75
 - held, 75
 - last_reset, 75
 - maxtags, 75
 - mintags, 75
- ccb_h
 - ccb, 47
 - ccb_abort, 52
 - ccb_accept_tio, 53
 - ccb_calc_geometry, 55
 - ccb_debug, 57
 - ccb_dev_match, 58
 - ccb_en_lun, 63
 - ccb_eng_exec, 65
 - ccb_eng_inq, 68
 - ccb_getdev, 70
 - ccb_getdevlist, 72
 - ccb_getdevstats, 74
 - ccb_immed_notify, 80
 - ccb_notify_ack, 82
 - ccb_pathinq, 84
 - ccb_pathstats, 91
 - ccb_relsim, 94
 - ccb_rescan, 96
 - ccb_resetbus, 97
 - ccb_resetdev, 98
 - ccb_scsiio, 99
 - ccb_setasync, 104
 - ccb_setdev, 105
 - ccb_termio, 107
 - ccb_trans_settings, 109
- ccb_hdr, 76
 - cbfcnp, 76
 - flags, 76
 - func_code, 77
 - path, 77
 - path_id, 77
 - periph_links, 77
 - periph_priv, 77
 - pinfo, 78
 - retry_count, 78
 - sim_links, 78
 - sim_priv, 78
 - status, 78
 - target_id, 78
 - target_lun, 78
 - timeout, 79
 - timeout_ch, 79
 - xpt_links, 79
- ccb_immed_notify, 80
 - ccb_h, 80
 - initiator_id, 80
 - message_args, 80
 - sense_data, 80
 - sense_len, 81
- CCB_INTERNAL
 - scsi_low.h, 836
- ccb_msgoutflag
 - slccb, 406
- CCB_NORETRY
 - scsi_low.h, 837
- CCB_NOSDONE
 - scsi_low.h, 837
- ccb_notify_ack, 82
 - ccb_h, 82
 - event, 82
 - seq_id, 82
- ccb_omsgoutflag
 - slccb, 406
- ccb_otag
 - slccb, 406

- ccb_pathinq, 83
 - async_flags, 84
 - base_transfer_speed, 84
 - bus_id, 84
 - ccb_h, 84
 - ccb_pathinq_settings_opaque, 84
 - dev_name, 84
 - fc, 84
 - hba_eng_cnt, 84
 - hba_inquiry, 84
 - hba_misc, 85
 - hba_vid, 85
 - hpath_id, 85
 - initiator_id, 85
 - max_lun, 85
 - max_target, 85
 - protocol, 85
 - protocol_version, 85
 - sas, 85
 - sim_vid, 86
 - spi, 86
 - target_sprt, 86
 - transport, 86
 - transport_version, 86
 - unit_number, 86
 - version_num, 86
 - vuhba_flags, 86
 - xport_specific, 86
- ccb_pathinq_settings_fc, 88
 - bitrate, 88
 - port, 88
 - wwnn, 88
 - wwpn, 88
- ccb_pathinq_settings_opaque
 - ccb_pathinq, 84
- ccb_pathinq_settings_sas, 89
 - bitrate, 89
- ccb_pathinq_settings_spi, 90
 - ppr_options, 90
- ccb_pathstats, 91
 - ccb_h, 91
 - last_reset, 91
- CCB_PERIPH_PRIV_SIZE
 - cam_ccb.h, 450
- ccb_pflags
 - scsi_sa.c, 906
- CCB_POLLED
 - scsi_low.h, 837
- ccb_priv_area, 92
 - bytes, 92
 - entries, 92
- ccb_priv_entry, 93
 - bytes, 93
 - field, 93
 - ptr, 93
- ccb_rcnt
 - slccb, 407
- ccb_relsim, 94
 - ccb_h, 94
 - openings, 94
 - qfrozen_cnt, 94
 - release_flags, 95
 - release_timeout, 95
- ccb_rescan, 96
 - ccb_h, 96
 - flags, 96
- ccb_resetbus, 97
 - ccb_h, 97
- ccb_resetdev, 98
 - ccb_h, 98
- ccb_scp
 - slccb, 407
- ccb_scsi_cmd
 - slccb, 407
- CCB_SCSIIO
 - scsi_low.h, 837
- ccb_scsiio, 99
 - ccb_h, 99
 - cdb_io, 99
 - cdb_len, 100
 - data_ptr, 100
 - dxfer_len, 100
 - init_id, 100
 - msg_len, 100
 - msg_ptr, 100
 - next_ccb, 100
 - req_map, 100
 - resid, 101
 - scsi_status, 101
 - sense_data, 101
 - sense_len, 101
 - sense_resid, 101
 - sglist_cnt, 101
 - tag_action, 101
 - tag_id, 101
- ccb_selrcnt
 - slccb, 407
- CCB_SENSE
 - scsi_low.h, 837
- ccb_sense
 - slccb, 407
- ccb_setasync, 103
 - callback, 103
 - callback_arg, 103
 - ccb_h, 104
 - event_enable, 104
- ccb_setdev, 105
 - ccb_h, 105

- dev_type, 105
- CCB_SILENT
 - scsi_low.h, 837
- CCB_SIM_PRIV_SIZE
 - cam_ccb.h, 450
- ccb_spriv_area, 106
 - bytes, 106
 - entries, 106
- ccb_sscp
 - slccb, 407
- CCB_STARTQ
 - scsi_low.h, 837
- ccb_state
 - scsi_cd.c, 662
 - scsi_ch.c, 721
 - scsi_da.c, 755
 - scsi_pt.c, 891
 - scsi_ses.c, 957
- ccb_tag
 - slccb, 407
- ccb_tc
 - slccb, 407
- ccb_tcmx
 - slccb, 408
- ccb_termio, 107
 - ccb_h, 107
 - termio_ccb, 107
- ccb_trans_settings, 108
 - ccb_h, 109
 - fc, 109
 - proto_specific, 109
 - protocol, 109
 - protocol_version, 109
 - sas, 109
 - scsi, 109
 - spi, 109
 - transport, 109
 - transport_version, 109
 - type, 110
 - valid, 110
 - xport_specific, 110
- ccb_trans_settings_fc, 111
 - bitrate, 111
 - port, 111
 - valid, 111
 - wwnn, 111
 - wwpn, 111
- ccb_trans_settings_sas, 112
 - bitrate, 112
 - valid, 112
- ccb_trans_settings_scsi, 113
 - flags, 113
 - valid, 113
- ccb_trans_settings_spi, 114
 - bus_width, 114
 - flags, 114
 - ppr_options, 114
 - sync_offset, 114
 - sync_period, 114
 - valid, 114
- CCB_Type
 - scsi_sa.c, 907
- ccb_type
 - scsi_pass.c, 879
 - scsi_targ_bh.c, 995
- CCB_URGENT
 - scsi_low.h, 837
- cag
 - ccb, 48
- cd6byteworkaround
 - scsi_cd.c, 664
- cd_audio_page, 116
 - flags, 116
 - format_lba, 116
 - lb_per_sec, 116
 - page_code, 116
 - param_len, 117
 - port, 117
 - unused, 117
- cd_audio_page::port_control, 118
 - channels, 118
 - volume, 118
- CD_CCB_BUFFER_IO
 - scsi_cd.c, 662
- CD_CCB_PROBE
 - scsi_cd.c, 662
- CD_CCB_RETRY_UA
 - scsi_cd.c, 663
- cd_ccb_state
 - scsi_cd.c, 662
- CD_CCB_TYPE_MASK
 - scsi_cd.c, 663
- CD_CCB_WAITING
 - scsi_cd.c, 663
- cd_changer_flags
 - scsi_cd.c, 663
- CD_FLAG_ACTIVE
 - scsi_cd.c, 663
- CD_FLAG_CHANGER
 - scsi_cd.c, 663
- CD_FLAG_DISC_LOCKED
 - scsi_cd.c, 663
- CD_FLAG_DISC_REMOVABLE
 - scsi_cd.c, 663
- CD_FLAG_INVALID
 - scsi_cd.c, 663
- CD_FLAG_NEW_DISC
 - scsi_cd.c, 663

- CD_FLAG_RETRY_UA
 - scsi_cd.c, 663
- CD_FLAG_SCHED_ON_COMP
 - scsi_cd.c, 663
- CD_FLAG_SCTX_INIT
 - scsi_cd.c, 663
- CD_FLAG_TAGGED_QUEUING
 - scsi_cd.c, 663
- CD_FLAG_VALID_MEDIA
 - scsi_cd.c, 663
- CD_FLAG_VALID_TOC
 - scsi_cd.c, 663
- cd_flags
 - scsi_cd.c, 663
- cd_mode_data, 119
 - blk_desc, 119
 - header, 119
 - page, 119
- cd_mode_data_10, 120
 - blk_desc, 120
 - header, 120
 - page, 120
- cd_mode_data_6_10, 121
 - mode_data_10, 121
 - mode_data_6, 121
- cd_mode_params, 122
 - alloc_len, 122
 - cdb_size, 122
 - mode_buf, 122
 - STAILQ_ENTRY, 122
- CD_MSF
 - scsi_cd.h, 700
- CD_PA_APR_VALID
 - scsi_cd.h, 700
- CD_PA_FORMAT_LBA
 - scsi_cd.h, 700
- CD_PA_IMMED
 - scsi_cd.h, 700
- CD_PA_SOTC
 - scsi_cd.h, 700
- CD_PAGE_CODE
 - scsi_cd.h, 700
- CD_PAGE_PS
 - scsi_cd.h, 700
- cd_page_size_table
 - scsi_cd.c, 692
- cd_page_sizes, 123
 - page, 123
 - page_size, 123
- cd_pages, 124
 - audio, 124
- cd_params, 125
 - blksize, 125
 - disksize, 125
- CD_Q_10_BYTE_ONLY
 - scsi_cd.c, 663
- CD_Q_BCD_TRACKS
 - scsi_cd.c, 663
- CD_Q_CHANGER
 - scsi_cd.c, 663
- CD_Q_NO_CHANGER
 - scsi_cd.c, 663
- CD_Q_NO_TOUCH
 - scsi_cd.c, 663
- CD_Q_NONE
 - scsi_cd.c, 663
- cd_quirk_entry, 126
 - inq_pat, 126
 - quirks, 126
- cd_quirk_table
 - scsi_cd.c, 692
- cd_quirks
 - scsi_cd.c, 663
- CD_RELADDR
 - scsi_cd.h, 700
- cd_softc, 127
 - bio_queue, 127
 - flags, 127
 - pinfo, 127
 - state, 127
- cd_state
 - scsi_cd.c, 663
- CD_STATE_NORMAL
 - scsi_cd.c, 664
- CD_STATE_PROBE
 - scsi_cd.c, 664
- cd_toc_single, 129
 - entry, 129
 - header, 129
- cd_tocdata, 130
 - entries, 130
 - header, 130
- cdasync
 - scsi_cd.c, 664
- cdb_bytes
 - cdb_t, 131
- cdb_io
 - ccb_accept_tio, 53
 - ccb_scsiio, 99
- cdb_len
 - ccb_accept_tio, 53
 - ccb_scsiio, 100
- cdb_ptr
 - cdb_t, 131
- cdb_size
 - cd_mode_params, 122
- cdb_t, 131
 - cdb_bytes, 131

- cdb_ptr, 131
- cdbg
 - ccb, 48
- cdchanger, 132
 - cur_device, 132
 - devq, 132
 - flags, 132
 - long_handle, 132
 - num_devices, 133
 - path_id, 133
 - short_handle, 133
 - start_time, 133
 - target_id, 133
- cdchangerschedule
 - scsi_cd.c, 665
- cdcheckmedia
 - scsi_cd.c, 665
- cdcleanup
 - scsi_cd.c, 666, 693
- cdcloses
 - scsi_cd.c, 666, 693
- cdcmdsizesysctl
 - scsi_cd.c, 667
- cddone
 - scsi_cd.c, 667
- cddriver
 - scsi_cd.c, 693
- cderror
 - scsi_cd.c, 668
- cdgetccb
 - scsi_cd.c, 669
- cdgetmode
 - scsi_cd.c, 670
- cdgetpage
 - scsi_cd.c, 670
- cdgetpagesize
 - scsi_cd.c, 671
- cdinit
 - scsi_cd.c, 693
- cdioctl
 - scsi_cd.c, 671, 693
- cdm
 - ccb, 48
- cdoninvalidate
 - scsi_cd.c, 672, 693
- cdopen
 - scsi_cd.c, 673, 693
- cdpause
 - scsi_cd.c, 673
- cdplay
 - scsi_cd.c, 674
- cdplaymsf
 - scsi_cd.c, 675
- cdplaytracks
 - scsi_cd.c, 676
- cdprevent
 - scsi_cd.c, 677
- cdreaddvdstructure
 - scsi_cd.c, 678
- cdreadsubchannel
 - scsi_cd.c, 678
- cdreadtoc
 - scsi_cd.c, 679
- cdregister
 - scsi_cd.c, 680, 694
- cdreportkey
 - scsi_cd.c, 681
- cdrunccb
 - scsi_cd.c, 682
- cdrunchangerqueue
 - scsi_cd.c, 682
- cdschedule
 - scsi_cd.c, 683
- cdsendkey
 - scsi_cd.c, 683
- cdsetmode
 - scsi_cd.c, 684
- cdsetspeed
 - scsi_cd.c, 684
- cdshorttimeout
 - scsi_cd.c, 685
- cdsize
 - scsi_cd.c, 686
- cdstart
 - scsi_cd.c, 686, 694
- cdstartunit
 - scsi_cd.c, 687
- cdstopunit
 - scsi_cd.c, 688
- cdstrategy
 - scsi_cd.c, 689, 694
- cdsysctlinit
 - scsi_cd.c, 690
- cee
 - ccb, 48
- cei
 - ccb, 48
- cel
 - ccb, 48
- CFG_ASYNC
 - scsi_low.h, 837
- CFG_NOATTEN
 - scsi_low.h, 838
- CFG_NODISC
 - scsi_low.h, 838
- CFG_NOPARITY
 - scsi_low.h, 838
- CFG_NOQTAG

- scsi_low.h, 838
- CFLEN
 - scsi_ses.c, 957
- cgd
 - ccb, 49
- cgdl
 - ccb, 49
- cgds
 - ccb, 49
- CH_CCB_PROBE
 - scsi_ch.c, 721
- ch_ccb_types
 - scsi_ch.c, 721
- CH_CCB_WAITING
 - scsi_ch.c, 721
- ch_cdevsw
 - scsi_ch.c, 738
- CH_DEVICE_CAP_PAGE
 - scsi_ch.h, 743
- CH_ELEMENT_ADDR_ASSIGN_PAGE
 - scsi_ch.h, 743
- CH_FLAG_INVALID
 - scsi_ch.c, 722
- CH_FLAG_OPEN
 - scsi_ch.c, 722
- ch_flags
 - scsi_ch.c, 721
- CH_Q_NO_DBD
 - scsi_ch.c, 722
- CH_Q_NONE
 - scsi_ch.c, 722
- ch_quirks
 - scsi_ch.c, 722
- ch_softc, 134
 - dev, 134
 - device_stats, 134
 - flags, 134
 - quirks, 134
 - saved_ccb, 135
 - sc_counts, 135
 - sc_exchangemask, 135
 - sc_firsts, 135
 - sc_movemask, 135
 - sc_picker, 135
 - sc_settledelay, 135
 - state, 135
- ch_state
 - scsi_ch.c, 722
- CH_STATE_NORMAL
 - scsi_ch.c, 722
- CH_STATE_PROBE
 - scsi_ch.c, 722
- CH_TIMEOUT_EXCHANGE_MEDIUM
 - scsi_ch.c, 738
- CH_TIMEOUT_INITIALIZE_ELEMENT_STATUS
 - scsi_ch.c, 738
- CH_TIMEOUT_MODE_SENSE
 - scsi_ch.c, 738
- CH_TIMEOUT_MOVE_MEDIUM
 - scsi_ch.c, 738
- CH_TIMEOUT_POSITION_TO_ELEMENT
 - scsi_ch.c, 738
- CH_TIMEOUT_READ_ELEMENT_STATUS
 - scsi_ch.c, 738
- CH_TIMEOUT_SEND_VOLTAGE
 - scsi_ch.c, 738
- CH_TRANS_GEOM_PARAMS_PAGE
 - scsi_ch.h, 743
- challenge_key
 - scsi_report_key_data_challenge, 340
- CHANGE_DEFINITION
 - scsi_all.h, 617
- CHANGER_MANUAL_CALL
 - scsi_cd.c, 663
- CHANGER_MAX_BUSY_SECONDS
 - scsi_cd.c, 662
- changer_max_busy_seconds
 - scsi_cd.c, 694
- CHANGER_MIN_BUSY_SECONDS
 - scsi_cd.c, 662
- changer_min_busy_seconds
 - scsi_cd.c, 694
- CHANGER_NEED_TIMEOUT
 - scsi_cd.c, 663
- CHANGER_SHORT_TMOUOT_SCHED
 - scsi_cd.c, 663
- CHANGER_TIMEOUT_SCHED
 - scsi_cd.c, 663
- CHANNEL
 - scsi_cd.h, 700
- CHANNEL_0
 - scsi_cd.h, 701
- CHANNEL_1
 - scsi_cd.h, 701
- CHANNEL_2
 - scsi_cd.h, 701
- CHANNEL_3
 - scsi_cd.h, 701
- channels
 - cd_audio_page::port_control, 118
- chasync
 - scsi_ch.c, 722
- chcleanup
 - scsi_ch.c, 723, 739
- chclose
 - scsi_ch.c, 723, 739
- chdone

- scsi_ch.c, 723
- chdriver
 - scsi_ch.c, 739
- cherror
 - scsi_ch.c, 724
- chexchange
 - scsi_ch.c, 725
- chgetelemstatus
 - scsi_ch.c, 726
- chgetparams
 - scsi_ch.c, 727
- chielem
 - scsi_ch.c, 728
- chinit
 - scsi_ch.c, 729, 739
- chioctl
 - scsi_ch.c, 730, 739
- chmove
 - scsi_ch.c, 730
- choninvalidate
 - scsi_ch.c, 731, 739
- chopen
 - scsi_ch.c, 732, 739
- chposition
 - scsi_ch.c, 732
- chregister
 - scsi_ch.c, 733, 739
- chsetvoltage
 - scsi_ch.c, 734
- chstart
 - scsi_ch.c, 734, 739
- CHUNIT
 - scsi_ch.c, 721
- cin
 - ccb, 49
- cmd6workaround
 - scsi_da.c, 757
- cmd_spec_info
 - scsi_sense_data, 372
- cmp_eca
 - lun_info::scsi_low_mode_sense_data, 170
- cmp_length
 - lun_info::scsi_low_mode_sense_data, 170
- cmp_page
 - lun_info::scsi_low_mode_sense_data, 170
- cmp_qc
 - lun_info::scsi_low_mode_sense_data, 170
- cmp_rlec
 - lun_info::scsi_low_mode_sense_data, 170
- cmp_spare
 - lun_info::scsi_low_mode_sense_data, 170
- cna
 - ccb, 49
- code
 - scsi_space, 377
- comp_algorithm
 - sa_softc, 201
 - scsi_data_compression_page, 215
- comstat
 - SesComStat, 401
- comstatus
 - SesComStat, 401
- content_desc
 - scsi_read_dvd_struct_data_generic_dcb, 310
- control
 - scsi_changedef, 209
 - scsi_erase, 222
 - scsi_exchange_medium, 223
 - scsi_format_unit, 225
 - scsi_initialize_element_status, 227
 - scsi_inquiry, 228
 - scsi_load_unload, 235
 - scsi_log_select, 238
 - scsi_log_sense, 239
 - scsi_mode_select_10, 265
 - scsi_mode_select_6, 266
 - scsi_mode_sense_10, 267
 - scsi_mode_sense_6, 268
 - scsi_move_medium, 272
 - scsi_pause, 275
 - scsi_play_10, 276
 - scsi_play_12, 277
 - scsi_play_msf, 278
 - scsi_play_rel_12, 280
 - scsi_play_track, 281
 - scsi_position_to_element, 283
 - scsi_prevent, 285
 - scsi_read_block_limits, 286
 - scsi_read_buffer, 288
 - scsi_read_capacity, 289
 - scsi_read_capacity_16, 290
 - scsi_read_cd_capacity, 295
 - scsi_read_defect_data_10, 297
 - scsi_read_defect_data_12, 298
 - scsi_read_dvd_structure, 324
 - scsi_read_element_status, 326
 - scsi_read_header, 329
 - scsi_read_subchannel, 330
 - scsi_read_toc, 332
 - scsi_reassign_blocks, 333
 - scsi_release, 335
 - scsi_report_key, 336
 - scsi_report_luns, 346
 - scsi_request_sense, 349
 - scsi_request_volume_element_address, 350
 - scsi_reserve, 352
 - scsi_reserve_release_unit, 353
 - scsi_rewind, 354

- scsi_rezero_unit, 355
- scsi_rw_10, 356
- scsi_rw_12, 358
- scsi_rw_16, 360
- scsi_rw_6, 362
- scsi_sa_rw, 363
- scsi_send_diag, 364
- scsi_send_key, 365
- scsi_send_receive, 367
- scsi_send_volume_tag, 368
- scsi_sense, 371
- scsi_set_speed, 376
- scsi_space, 377
- scsi_start_stop_unit, 378
- scsi_sync_cache, 381
- scsi_tape_locate, 382
- scsi_test_unit_ready, 387
- scsi_write_buffer, 391
- scsi_write_filemarks, 392
- cookie
 - ccb_dev_position, 60
- copy_element_status
 - scsi_ch.c, 735
- copy_voltag
 - scsi_ch.c, 735
- correction_span
 - scsi_da_rw_recovery_page, 213
- count
 - read_element_status_header, 192
 - scsi_read_element_status, 326
 - scsi_request_volume_element_address, 350
 - scsi_space, 377
- counter
 - xpt_scan_bus_info, 426
- cpi
 - ccb, 49
 - xpt_scan_bus_info, 426
- cpis
 - ccb, 49
- cps_type
 - scsi_read_dvd_struct_data_copyright, 303
- crb
 - ccb, 49
- crn
 - ccb, 49
- crd
 - ccb, 49
- crs
 - ccb, 50
- csa
 - ccb, 50
- csd
 - ccb, 50
- csio
 - ccb, 50
- cstat
 - ses_objstat, 397
- ctio
 - ccb, 50
- ctl_dev
 - sa_devs, 196
- ctrl_mode
 - sa_softc, 201
- cts
 - ccb, 50
- CTS_FC_VALID_PORT
 - cam_ccb.h, 450
- CTS_FC_VALID_SPEED
 - cam_ccb.h, 451
- CTS_FC_VALID_WWNN
 - cam_ccb.h, 451
- CTS_FC_VALID_WWPN
 - cam_ccb.h, 451
- CTS_SAS_VALID_SPEED
 - cam_ccb.h, 451
- CTS_SCSI_FLAGS_TAG_ENB
 - cam_ccb.h, 451
- CTS_SCSI_VALID_TQ
 - cam_ccb.h, 451
- CTS_SPI_FLAGS_DISC_ENB
 - cam_ccb.h, 451
- CTS_SPI_VALID_BUS_WIDTH
 - cam_ccb.h, 451
- CTS_SPI_VALID_DISC
 - cam_ccb.h, 451
- CTS_SPI_VALID_PPR_OPTIONS
 - cam_ccb.h, 452
- CTS_SPI_VALID_SYNC_OFFSET
 - cam_ccb.h, 452
- CTS_SPI_VALID_SYNC_RATE
 - cam_ccb.h, 452
- cts_type
 - cam_ccb.h, 457
- CTS_TYPE_CURRENT_SETTINGS
 - cam_ccb.h, 457
- CTS_TYPE_USER_SETTINGS
 - cam_ccb.h, 457
- cur_device
 - cdchanger, 132
- cyl_skew_0
 - disk_pages::format_device_page, 151
- cyl_skew_1
 - disk_pages::format_device_page, 151
- cylinder
 - scsi_defect_desc_bytes_from_index, 218
 - scsi_defect_desc_phys_sector, 219
- cylinders
 - ccb_calc_geometry, 55

- disk_params, 156
- D
 - scsi_all.c, 591
- D_TAPE
 - scsi_sa.c, 907
- DA_CCB_BUFFER_IO
 - scsi_da.c, 756
- DA_CCB_DUMP
 - scsi_da.c, 756
- DA_CCB_PROBE
 - scsi_da.c, 756
- DA_CCB_PROBE2
 - scsi_da.c, 756
- DA_CCB_RETRY_UA
 - scsi_da.c, 756
- da_ccb_state
 - scsi_da.c, 756
- DA_CCB_TYPE_MASK
 - scsi_da.c, 756
- DA_CCB_WAITING
 - scsi_da.c, 756
- DA_DEFAULT_RETRY
 - scsi_da.c, 755
- DA_DEFAULT_SEND_ORDERED
 - scsi_da.c, 755
- DA_DEFAULT_TIMEOUT
 - scsi_da.c, 755
- da_default_timeout
 - scsi_da.c, 770
- DA_FLAG_NEED_OTAG
 - scsi_da.c, 756
- DA_FLAG_NEW_PACK
 - scsi_da.c, 756
- DA_FLAG_OPEN
 - scsi_da.c, 756
- DA_FLAG_PACK_INVALID
 - scsi_da.c, 756
- DA_FLAG_PACK_LOCKED
 - scsi_da.c, 756
- DA_FLAG_PACK_REMOVABLE
 - scsi_da.c, 756
- DA_FLAG_RETRY_UA
 - scsi_da.c, 756
- DA_FLAG_SCTX_INIT
 - scsi_da.c, 756
- DA_FLAG_TAGGED_QUEUEING
 - scsi_da.c, 756
- DA_FLAG_WENT_IDLE
 - scsi_da.c, 756
- da_flags
 - scsi_da.c, 756
- DA_ORDEREDTAG_INTERVAL
 - scsi_da.c, 755
- DA_Q_NO_6_BYTE
 - scsi_da.c, 756
- DA_Q_NO_PREVENT
 - scsi_da.c, 756
- DA_Q_NO_SYNC_CACHE
 - scsi_da.c, 756
- DA_Q_NONE
 - scsi_da.c, 756
- da_quirk_entry, 137
 - inq_pat, 137
 - quirks, 137
- da_quirk_table
 - scsi_da.c, 770
- da_quirks
 - scsi_da.c, 756
- da_retry_count
 - scsi_da.c, 770
- da_send_ordered
 - scsi_da.c, 770
- da_softc, 138
 - bio_queue, 138
- da_state
 - scsi_da.c, 756
- DA_STATE_NORMAL
 - scsi_da.c, 757
- DA_STATE_PROBE
 - scsi_da.c, 757
- DA_STATE_PROBE2
 - scsi_da.c, 757
- daasync
 - scsi_da.c, 757
- dacleanup
 - scsi_da.c, 758, 770
- daclose
 - scsi_da.c, 758
- dacmdsizesysctl
 - scsi_da.c, 759
- dadone
 - scsi_da.c, 759
- dadrivers
 - scsi_da.c, 770
- dadump
 - scsi_da.c, 760, 771
- daerror
 - scsi_da.c, 761
- dagetcapacity
 - scsi_da.c, 762
- dainit
 - scsi_da.c, 763, 771
- daoninvalidate
 - scsi_da.c, 764, 771
- daprevent
 - scsi_da.c, 764
- daregister

- scsi_da.c, 765, 771
- dasendorderedtag
 - scsi_da.c, 766, 771
- dasetgeom
 - scsi_da.c, 766
- dashutdown
 - scsi_da.c, 766
- dastart
 - scsi_da.c, 767, 771
- dastrategy
 - scsi_da.c, 767, 771
- dasysctlinit
 - scsi_da.c, 768
- data
 - targbh_cmd_desc, 419
- data_increment
 - targbh_cmd_desc, 420
- data_len
 - read_dvd_struct_data_list, 187
 - read_dvd_struct_write_prot, 189
 - scsi_read_dvd_struct_data_bca, 301
 - scsi_read_dvd_struct_data_copy_manage, 302
 - scsi_read_dvd_struct_data_copyright, 303
 - scsi_read_dvd_struct_data_dcb, 304
 - scsi_read_dvd_struct_data_dds, 305
 - scsi_read_dvd_struct_data_disc_id, 306
 - scsi_read_dvd_struct_data_disc_key, 308
 - scsi_read_dvd_struct_data_disc_key_blk, 309
 - scsi_read_dvd_struct_data_header, 311
 - scsi_read_dvd_struct_data_lead_in, 314
 - scsi_read_dvd_struct_data_manufacturer, 317
 - scsi_read_dvd_struct_data_medium_status, 318
 - scsi_read_dvd_struct_data_physical, 319
 - scsi_read_dvd_struct_data_prot_discid, 320
 - scsi_read_dvd_struct_data_rmd, 321
 - scsi_read_dvd_struct_data_rmd_borderout, 322
 - scsi_read_dvd_struct_data_spare_area, 323
 - scsi_read_header, 329
 - scsi_read_subchannel, 330
 - scsi_read_toc, 332
 - scsi_report_key_data_agid, 338
 - scsi_report_key_data_asf, 339
 - scsi_report_key_data_challenge, 340
 - scsi_report_key_data_header, 341
 - scsi_report_key_data_key1_key2, 342
 - scsi_report_key_data_rpc, 343
 - scsi_report_key_data_title, 345
 - scsi_send_key_data_rpc, 366
- data_length
 - scsi_mode_header_10, 262
 - scsi_mode_header_6, 263
- data_ptr
 - ccb_eng_exec, 65
 - ccb_scsiio, 100
- data_resid
 - targbh_cmd_desc, 420
- data_strobe_offset_cnt
 - scsi_da_rw_recovery_page, 213
- datalen
 - scsi_changedef, 209
 - scsi_log_header, 236
 - scsi_mode_hdr_10, 260
 - scsi_mode_hdr_6, 261
- day
 - scsi_read_dvd_struct_data_disc_id, 306
- dcb
 - scsi_read_dvd_struct_data_dcb, 304
- dcb_data
 - scsi_read_dvd_struct_data_generic_dcb, 310
- dce_and_dcc
 - scsi_data_compression_page, 215
- dcomp
 - sa_comp_t, 194
- dconf
 - sa_comp_t, 194
- dde_and_red
 - scsi_data_compression_page, 215
- dds_info
 - scsi_read_dvd_struct_data_dds, 305
- dead_sim_action
 - cam_xpt.c, 527
- dead_sim_poll
 - cam_xpt.c, 527
- DECLARE_MODULE
 - cam_xpt.c, 527
 - scsi_low_pisa.c, 868
- decomp_algorithm
 - scsi_data_compression_page, 215
- defect_descriptor
 - scsi_reassign_blocks_data, 334
- defect_list_length
 - format_defect_list_header, 161
- deferred_ac
 - cam_periph, 34
- deferred_callback
 - cam_periph, 34
- density
 - scsi_mode_blk_desc, 258
 - scsi_read_dvd_struct_data_layer_desc, 312
- density_code
 - scsi_mode_block_descr, 259
- depth
 - xpt_traverse_config, 428
- desc
 - asc_table_entry, 20
 - op_table_entry, 174

- sense_key_table_entry, 393
- dest_len
 - ccb_eng_exec, 66
- dev
 - ch_softc, 134
 - pass_softc, 180
- dev_active
 - cam_ccbq, 24
 - ccb_getdevstats, 74
- dev_allocq_is_runnable
 - cam_xpt.c, 527
- DEV_IDLEN
 - cam_ccb.h, 452
- DEV_MATCH_ANY
 - cam_ccb.h, 458
- DEV_MATCH_BUS
 - cam_ccb.h, 458
- DEV_MATCH_DEVICE
 - cam_ccb.h, 458
- DEV_MATCH_INQUIRY
 - cam_ccb.h, 458
- DEV_MATCH_LUN
 - cam_ccb.h, 458
- DEV_MATCH_NONE
 - cam_ccb.h, 458
- DEV_MATCH_PATH
 - cam_ccb.h, 458
- dev_match_pattern, 139
 - pattern, 139
 - type, 139
- DEV_MATCH_PERIPH
 - cam_ccb.h, 458
- dev_match_result, 140
 - result, 140
 - type, 140
- dev_match_ret
 - cam_xpt.c, 524
- DEV_MATCH_TARGET
 - cam_ccb.h, 458
- dev_match_type
 - cam_ccb.h, 457
- dev_name
 - bus_match_pattern, 22
 - bus_match_result, 23
 - ccb_pathinq, 84
- dev_openings
 - cam_ccbq, 24
 - ccb_getdevstats, 74
- dev_pattern_flags
 - cam_ccb.h, 458
- dev_pos_type
 - cam_ccb.h, 458
- dev_qual2
 - scsi_inquiry_data, 230
- dev_result_flags
 - cam_ccb.h, 458
- DEV_RESULT_NOFLAG
 - cam_ccb.h, 458
- DEV_RESULT_UNCONFIGURED
 - cam_ccb.h, 458
- dev_spec
 - scsi_mode_header_10, 262
 - scsi_mode_header_6, 263
- dev_specific
 - scsi_mode_hdr_10, 260
 - scsi_mode_hdr_6, 261
- dev_type
 - ccb_setdev, 105
- device
 - cam_path, 31
 - ccb_dm_cookie, 61
 - scsi_inquiry_data, 230
 - scsi_vpd_unit_serial_number, 389
- device_is_alloc_queued
 - cam_xpt.c, 527
- device_is_send_queued
 - cam_xpt.c, 527
- device_match_pattern, 141
 - flags, 141
 - inq_pat, 141
 - path_id, 141
 - target_id, 141
 - target_lun, 141
- device_match_result, 143
 - flags, 143
 - inq_data, 143
 - path_id, 143
 - target_id, 143
 - target_lun, 143
- device_pattern
 - match_pattern, 172
- device_result
 - match_result, 173
- device_stats
 - ch_softc, 134
 - pass_softc, 180
 - pt_softc, 186
 - sa_softc, 201
 - targ_softc, 417
 - targbh_softc, 421
- devq
 - cam_sim, 39
 - cdchanger, 132
- devq_openings
 - cam_ccbq, 24
 - ccb_getdevstats, 75
- devq_queued
 - ccb_getdevstats, 75

- devs
 - sa_softc, 201
- disc_key
 - scsi_read_dvd_struct_data_disc_key, 308
- disc_key_pack_data
 - scsi_read_dvd_struct_data_disc_key_blk, 309
- disc_physical_data
 - scsi_read_dvd_struct_data_lead_in, 314
- disc_size_max_rate
 - scsi_read_dvd_struct_data_layer_desc, 312
- disc_type_id
 - scsi_read_dvd_struct_data_medium_status, 318
- DISK_FMT_HSEC
 - scsi_da.h, 775
- DISK_FMT_RMB
 - scsi_da.h, 775
- DISK_FMT_SSEC
 - scsi_da.h, 775
- DISK_FMT_SURF
 - scsi_da.h, 776
- disk_pages, 145
 - flexible_disk, 145
 - format_device, 145
 - rigid_geometry, 145
- disk_pages::flexible_disk_page, 146
 - bytes_s_0, 146
 - bytes_s_1, 146
 - driv_step_0, 147
 - driv_step_1, 147
 - driv_step_pw, 147
 - head_load_del, 147
 - head_stl_del_0, 147
 - head_stl_del_1, 147
 - head_uload_del, 147
 - medium_rot_rate_0, 147
 - medium_rot_rate_1, 147
 - motor_off_del, 147
 - motor_on_del, 147
 - ncyl_0, 148
 - ncyl_1, 148
 - nheads, 148
 - pg_code, 148
 - pg_length, 148
 - pin32_pin2, 148
 - pin4_pint1, 148
 - reserved30, 148
 - reserved31, 148
 - sec_per_track, 148
 - spc, 148
 - st_cyl_rwc_0, 149
 - st_cyl_rwc_1, 149
 - st_cyl_wp_0, 149
 - st_cyl_wp_1, 149
 - trdy_ssn_mo, 149
 - write_comp, 149
 - xfr_rate_0, 149
 - xfr_rate_1, 149
- disk_pages::format_device_page, 150
 - alt_sec_0, 150
 - alt_sec_1, 150
 - alt_trk_v_0, 150
 - alt_trk_v_1, 150
 - alt_trk_z_0, 151
 - alt_trk_z_1, 151
 - bytes_s_0, 151
 - bytes_s_1, 151
 - cyl_skew_0, 151
 - cyl_skew_1, 151
 - flags, 151
 - interleave_0, 151
 - interleave_1, 151
 - pg_code, 151
 - pg_length, 151
 - ph_sec_t_0, 152
 - ph_sec_t_1, 152
 - reserved21, 152
 - reserved22, 152
 - reserved23, 152
 - trk_skew_0, 152
 - trk_skew_1, 152
 - trk_z_0, 152
 - trk_z_1, 152
- disk_pages::rigid_geometry_page, 153
 - driv_step_0, 153
 - driv_step_1, 153
 - land_zone_0, 153
 - land_zone_1, 153
 - land_zone_2, 154
 - medium_rot_rate_0, 154
 - medium_rot_rate_1, 154
 - ncyl_0, 154
 - ncyl_1, 154
 - ncyl_2, 154
 - nheads, 154
 - pg_code, 154
 - pg_length, 154
 - reserved19, 154
 - reserved22, 154
 - reserved23, 155
 - rot_offset, 155
 - rpl, 155
 - st_cyl_rwc_0, 155
 - st_cyl_rwc_1, 155
 - st_cyl_rwc_2, 155
 - st_cyl_wp_0, 155
 - st_cyl_wp_1, 155
 - st_cyl_wp_2, 155

- disk_params, 156
 - cylinders, 156
 - heads, 156
 - secs_per_track, 156
 - secsize, 156
 - sectors, 156
- disksize
 - cd_params, 125
- dlbaddr
 - scsi_reassign_blocks_data, 334
- DM_RET_ACTION_MASK
 - cam_xpt.c, 525
- DM_RET_COPY
 - cam_xpt.c, 524
- DM_RET_DESCEND
 - cam_xpt.c, 525
- DM_RET_ERROR
 - cam_xpt.c, 525
- DM_RET_FLAG_MASK
 - cam_xpt.c, 524
- DM_RET_NONE
 - cam_xpt.c, 524
- DM_RET_STOP
 - cam_xpt.c, 524
- dmax_len
 - ccb_eng_exec, 66
- DoorLock
 - scfg, 207
- driv_step_0
 - disk_pages::flexible_disk_page, 147
 - disk_pages::rigid_geometry_page, 153
- driv_step_1
 - disk_pages::flexible_disk_page, 147
 - disk_pages::rigid_geometry_page, 153
- driv_step_pw
 - disk_pages::flexible_disk_page, 147
- driver_name
 - periph_driver, 182
- dsreg
 - sa_softc, 202
- dst
 - scsi_move_medium, 272
 - scsi_position_to_element, 283
- dt_scsi_addr
 - read_element_status_descriptor, 190
- dt_scsi_flags
 - read_element_status_descriptor, 190
- DVF SCSI_BITS
 - scsi_dvcfg.h, 784
- DVF SCSI_DEF CFG
 - scsi_dvcfg.h, 784
- DVF SCSI_DISC
 - scsi_dvcfg.h, 784
- DVF SCSI_LINK
 - scsi_dvcfg.h, 784
- DVF SCSI_NOPARITY
 - scsi_dvcfg.h, 784
- DVF SCSI_OFFSET
 - scsi_dvcfg.h, 785
- DVF SCSI_PERIOD
 - scsi_dvcfg.h, 785
- DVF SCSI_QTAG
 - scsi_dvcfg.h, 785
- DVF SCSI_SAVESP
 - scsi_dvcfg.h, 785
- DVF SCSI_SPO
 - scsi_dvcfg.h, 785
- DVF SCSI_SPI
 - scsi_dvcfg.h, 785
- DVF SCSI_SYNC
 - scsi_dvcfg.h, 785
- DVF SCSI_SYNCMASK
 - scsi_dvcfg.h, 785
- DVF SCSI_WAIT
 - scsi_dvcfg.h, 785
- dxfer_len
 - ccb_eng_exec, 66
 - ccb_scsiio, 100
- E
 - scsi_all.c, 591
- ea
 - scsi_mode_sense_data, 270
 - scsi_send_volume_tag, 368
- EAD_LZ1V1
 - cam_ccb.h, 459
- EAD_LZ2V1
 - cam_ccb.h, 459
- EAD_LZ2V2
 - cam_ccb.h, 459
- EAD_VUNIQUE
 - cam_ccb.h, 459
- eaddr
 - read_element_status_descriptor, 190
 - scsi_request_volume_element_address, 350
- eca_and_aen
 - scsi_control_page, 211
- edl
 - read_element_status_page_header, 193
- ei_algo
 - cam_ccb.h, 458
- ei_type
 - cam_ccb.h, 459
- EIT_BUFFER
 - cam_ccb.h, 459
- EIT_ENCRYPT
 - cam_ccb.h, 459
- EIT_LOSSLESS

- cam_ccb.h, 459
- EIT_LOSSY
 - cam_ccb.h, 459
- ELEMENT_TYPE_ALL
 - scsi_ch.h, 743
- ELEMENT_TYPE_DT
 - scsi_ch.h, 743
- ELEMENT_TYPE_IE
 - scsi_ch.h, 744
- ELEMENT_TYPE_MASK
 - scsi_ch.h, 744
- ELEMENT_TYPE_MT
 - scsi_ch.h, 744
- ELEMENT_TYPE_ST
 - scsi_ch.h, 744
- enable
 - ccb_en_lun, 63
- enc_maxelt
 - SesThdr, 404
- enc_subenc
 - SesThdr, 404
- enc_tlen
 - SesThdr, 404
- enc_type
 - SesThdr, 404
- ENCL_SVALID
 - scsi_ses.c, 957
- encobj, 157
 - encstat, 157
 - entype, 157
 - priv, 157
 - subenclosure, 157
 - svalid, 157
- encPid
 - SesEncDesc, 402
- encRev
 - SesEncDesc, 402
- encstat
 - encobj, 157
- entyp
 - scsi_ses.c, 966
- entype
 - encobj, 157
- encvec, 158
 - get_encstat, 158
 - get_objstat, 158
 - init_enc, 158
 - set_encstat, 158
 - set_objstat, 158
 - softc_init, 158
- encVen
 - SesEncDesc, 402
- encVid
 - SesEncDesc, 402
- encWWN
 - SesEncDesc, 402
- end_f
 - scsi_play_msf, 278
- end_index
 - scsi_play_track, 281
- end_m
 - scsi_play_msf, 278
- end_s
 - scsi_play_msf, 278
- end_sector_layer0
 - scsi_read_dvd_struct_data_layer_desc, 312
- end_track
 - scsi_play_track, 281
- eng_algo
 - ccb_eng_inq, 68
- eng_memeory
 - ccb_eng_inq, 68
- eng_num
 - ccb_eng_exec, 66
 - ccb_eng_inq, 68
- eng_type
 - ccb_eng_inq, 69
- engage_ch
 - scsi_low_osdep_interface, 248
- engdata_ptr
 - ccb_eng_exec, 66
- entries
 - camq, 43
 - ccb_ppriv_area, 92
 - ccb_spriv_area, 106
 - cd_tocdata, 130
 - read_dvd_struct_data_list, 187
- entry
 - cd_toc_single, 129
- eot_reten_load
 - scsi_load_unload, 235
- er_dev
 - sa_devs::sa_mode_devs, 197
- ERASE
 - scsi_sa.h, 938
- ERASE_TIMEOUT
 - scsi_sa.c, 907
- errinfo
 - sa_softc, 202
- error_bits
 - scsi_low_error_code, 241
- error_code
 - scsi_low_error_code, 241
 - scsi_sense_data, 372
- event
 - ccb_notify_ack, 82
- event_enable
 - ccb_setasync, 104

- ew_bufsize
 - scsi_dev_conf_page, 220
- EXCHANGE_MEDIUM
 - scsi_ch.h, 744
- EXCHANGE_MEDIUM_INV1
 - scsi_ch.h, 744
- EXCHANGE_MEDIUM_INV2
 - scsi_ch.h, 744
- exchange_with
 - page_device_capabilities, 175
- EXCHANGE_WITH_DT
 - scsi_ch.h, 744
- EXCHANGE_WITH_IE
 - scsi_ch.h, 744
- EXCHANGE_WITH_MT
 - scsi_ch.h, 744
- EXCHANGE_WITH_ST
 - scsi_ch.h, 745
- extra_bytes
 - scsi_sense_data, 372
- extra_len
 - scsi_sense_data, 373
- extra_wp
 - scsi_dev_conf_page, 220
- FATALIO
 - scsi_low.h, 838
- fc
 - ccb_pathinq, 84
 - ccb_trans_settings, 109
- FCD_CODE_MASK
 - scsi_da.h, 776
- FCD_FORMATTED
 - scsi_da.h, 776
- FCD_NOMEDIA
 - scsi_da.h, 776
- FCD_UNFORMATTED
 - scsi_da.h, 776
- fdst
 - scsi_exchange_medium, 223
- fdtea
 - page_element_address_assignment, 177
- fear
 - read_element_status_header, 192
- fetchtableentries
 - scsi_all.c, 592
- fieea
 - page_element_address_assignment, 177
- field
 - ccb_priv_entry, 93
- field_id_1
 - scsi_read_dvd_struct_data_leadin, 314
- field_id_2
 - scsi_read_dvd_struct_data_leadin, 315
- field_id_3
 - scsi_read_dvd_struct_data_leadin, 315
- field_id_4
 - scsi_read_dvd_struct_data_leadin, 315
- field_id_5
 - scsi_read_dvd_struct_data_leadin, 315
- filemarks
 - sa_softc, 202
- fileno
 - sa_softc, 202
- find_mode_page_10
 - scsi_all.h, 645
- find_mode_page_6
 - scsi_all.h, 645
- firstblk
 - scsi_tape_position_data, 384
- flag1
 - scfg, 207
- flag2
 - scfg, 207
- flags
 - bus_match_pattern, 22
 - cam_periph, 34
 - cam_sim, 40
 - ccb_debug, 57
 - ccb_hdr, 76
 - ccb_rescan, 96
 - ccb_trans_settings_scsi, 113
 - ccb_trans_settings_spi, 114
 - cd_audio_page, 116
 - cd_softc, 127
 - cdchanger, 132
 - ch_softc, 134
 - device_match_pattern, 141
 - device_match_result, 143
 - disk_pages::format_device_page, 151
 - pass_softc, 180
 - periph_match_pattern, 183
 - read_element_status_page_header, 193
 - sa_softc, 202
 - scsi_inquiry_data, 231
 - scsi_sense_data, 373
 - scsi_status_iu_header, 380
 - scsi_tape_position_data, 384
 - targbh_softc, 422
 - xpt_softc, 427
- flags1
 - read_element_status_descriptor, 191
- flags2
 - read_element_status_descriptor, 191
- flexible_disk
 - disk_pages, 145
- format
 - scsi_read_defect_data_10, 297

- scsi_read_defect_data_12, [298](#)
- scsi_read_defect_data_hdr_10, [299](#)
- scsi_read_defect_data_hdr_12, [300](#)
- scsi_read_dvd_structure, [324](#)
- format_capacity_descriptor, [159](#)
 - block_length, [159](#)
 - byte4, [159](#)
 - nblocks, [159](#)
- format_capacity_list_header, [160](#)
 - capacity_list_length, [160](#)
 - unused, [160](#)
- format_code
 - read_dvd_struct_list_entry, [188](#)
- format_defect_list_header, [161](#)
 - byte2, [161](#)
 - defect_list_length, [161](#)
 - reserved, [161](#)
- format_device
 - disk_pages, [145](#)
- format_ipat_descriptor, [162](#)
 - byte1, [162](#)
 - pat_length, [162](#)
 - pattern_type, [162](#)
- format_lba
 - cd_audio_page, [116](#)
- FORMAT_UNIT
 - scsi_da.h, [776](#)
- from_track
 - scsi_read_toc, [332](#)
- fru
 - scsi_sense_data, [373](#)
- fsea
 - page_element_address_assignment, [177](#)
- FU_BFI_FORMAT
 - scsi_da.h, [776](#)
- FU_BLOCK_FORMAT
 - scsi_da.h, [776](#)
- FU_CMPLST
 - scsi_da.h, [776](#)
- FU_DLH_DCRT
 - scsi_da.h, [776](#)
- FU_DLH_DPRY
 - scsi_da.h, [776](#)
- FU_DLH_DSP
 - scsi_da.h, [777](#)
- FU_DLH_FOV
 - scsi_da.h, [777](#)
- FU_DLH_IMMED
 - scsi_da.h, [777](#)
- FU_DLH_IP
 - scsi_da.h, [777](#)
- FU_DLH_STPF
 - scsi_da.h, [777](#)
- FU_DLH_VS
 - scsi_da.h, [777](#)
- FU_FMT_DATA
 - scsi_da.h, [777](#)
- FU_FORMAT_MASK
 - scsi_da.h, [777](#)
- FU_INIT_LBA_EACH
 - scsi_da.h, [777](#)
- FU_INIT_LBA_MSB
 - scsi_da.h, [777](#)
- FU_INIT_NO_HDR
 - scsi_da.h, [777](#)
- FU_INIT_PAT_DEFAULT
 - scsi_da.h, [778](#)
- FU_INIT_PAT_REPEAT
 - scsi_da.h, [778](#)
- FU_INIT_SI
 - scsi_da.h, [778](#)
- FU_PHYS_FORMAT
 - scsi_da.h, [778](#)
- func_code
 - ccb_hdr, [77](#)
- gap_size
 - scsi_dev_conf_page, [221](#)
- gbit
 - scsi_ses.c, [957](#)
- gbyte
 - scsi_ses.c, [957](#)
- GenCode
 - SesCfgHdr, [400](#)
- generation
 - cam_pinfo, [38](#)
 - camq, [43](#)
 - ccb_getdevlist, [72](#)
 - periph_driver, [182](#)
 - xpt_softc, [427](#)
- GENERATIONCMP
 - cam.h, [437](#)
- generations
 - ccb_dev_position, [60](#)
- get_encstat
 - encvec, [158](#)
- get_objstat
 - encvec, [158](#)
- gget16
 - scsi_ses.c, [957](#)
- gget24
 - scsi_ses.c, [957](#)
- gget32
 - scsi_ses.c, [958](#)
- gget8
 - scsi_ses.c, [958](#)
- gran
 - scsi_read_block_limits_data, [287](#)

- grp6_len
 - ccb_en_lun, 63
 - ioc_enable_lun, 163
- grp7_len
 - ccb_en_lun, 64
 - ioc_enable_lun, 163
- hba_eng_cnt
 - ccb_pathinq, 84
- HBA_IDLEN
 - cam_ccb.h, 452
- hba_inquiry
 - ccb_pathinq, 84
- hba_misc
 - ccb_pathinq, 85
- hba_vid
 - ccb_pathinq, 85
- hdr
 - sa_comp_t, 194
- head
 - scsi_defect_desc_bytes_from_index, 218
 - scsi_defect_desc_phys_sector, 219
- head_load_del
 - disk_pages::flexible_disk_page, 147
- head_offset_count
 - scsi_da_rw_recovery_page, 213
- head_stl_del_0
 - disk_pages::flexible_disk_page, 147
- head_stl_del_1
 - disk_pages::flexible_disk_page, 147
- head_uoload_del
 - disk_pages::flexible_disk_page, 147
- header
 - cd_mode_data, 119
 - cd_mode_data_10, 120
 - cd_toc_single, 129
 - cd_tocdata, 130
 - scsi_mode_sense_data, 270
- heads
 - ccb_calc_geometry, 56
 - disk_params, 156
- heap_down
 - cam_queue.c, 498
- heap_up
 - cam_queue.c, 498
- held
 - cam_ccbq, 25
 - ccb_getdevstats, 75
- hour
 - scsi_read_dvd_struct_data_disc_id, 306
- how
 - scsi_changedef, 209
 - scsi_prevent, 285
 - scsi_start_stop_unit, 378
- hpath_id
 - ccb_pathinq, 85
- HW_INACTIVE
 - scsi_low.h, 838
- HW_INITIALIZING
 - scsi_low.h, 838
- HW_PDMASRT
 - scsi_low.h, 838
- HW_POWDOWN
 - scsi_low.h, 839
- HW_POWERCTRL
 - scsi_low.h, 839
- HW_READ_PADDING
 - scsi_low.h, 839
- HW_RESUME
 - scsi_low.h, 839
- HW_WRITE_PADDING
 - scsi_low.h, 839
- immed_notify_slist
 - targbh_softc, 422
- immediate
 - scsi_load_unload, 235
 - scsi_rewind, 354
- immediate_priority
 - cam_periph, 34
- index
 - cam_pinfo, 38
 - ccb_getdevlist, 72
- info
 - scsi_sense_data, 373
- init
 - periph_driver, 182
- init_enc
 - encvec, 158
- init_id
 - ccb_accept_tio, 54
 - ccb_scsiio, 100
- init_level
 - targbh_softc, 422
- init_scsi_delay
 - scsi_all.c, 593
- INITIALIZE_ELEMENT_STATUS
 - scsi_ch.h, 745
- initiator_id
 - ccb_immed_notify, 80
 - ccb_pathinq, 85
- inq_cmd
 - scsi_low.c, 827
- inq_data
 - ccb_getdev, 70
 - device_match_result, 143
- inq_data_len
 - targbh_softc, 422

- INQ_DATA_TQ_ENABLED
 - scsi_all.h, 617
- inq_pat
 - cd_quirk_entry, 126
 - da_quirk_entry, 137
 - device_match_pattern, 141
 - sa_quirk_entry, 198
 - scsi_op_quirk_entry, 274
 - scsi_sense_quirk_entry, 374
 - xpt_quirk_entry, 425
- INQUIRY
 - scsi_all.h, 617
- interleave
 - scsi_format_unit, 225
- interleave_0
 - disk_pages::format_device_page, 151
- interleave_1
 - disk_pages::format_device_page, 151
- invert
 - scsi_exchange_medium, 223
 - scsi_move_medium, 272
 - scsi_position_to_element, 283
- IO_TIMEOUT
 - scsi_sa.c, 907
- ioc_enable_lun, 163
 - grp6_len, 163
 - grp7_len, 163
 - lun_id, 163
 - path_id, 163
 - target_id, 163
- IOCDBLEN
 - cam_ccb.h, 452
- key1
 - scsi_report_key_data_key1_key2, 342
- L
 - scsi_all.c, 591
- land_zone_0
 - disk_pages::rigid_geometry_page, 153
- land_zone_1
 - disk_pages::rigid_geometry_page, 153
- land_zone_2
 - disk_pages::rigid_geometry_page, 154
- last_addr
 - scsi_read_dvd_struct_data_lead_in, 315
- last_ctl_cdb
 - scsi_sa.c, 907
- last_ctl_resid
 - scsi_sa.c, 907
- last_ctl_sense
 - scsi_sa.c, 907
- last_io_cdb
 - scsi_sa.c, 907
- last_io_resid
 - scsi_sa.c, 907
- last_io_sense
 - scsi_sa.c, 907
- last_media_blksize
 - sa_softc, 202
- last_reset
 - ccb_getdevstats, 75
 - ccb_pathstats, 91
- last_resid_was_io
 - sa_softc, 202
- last_sector_num
 - scsi_read_dvd_struct_data_rmd, 321
- lastblk
 - scsi_tape_position_data, 384
- layer_desc
 - scsi_read_dvd_struct_data_physical, 319
- layer_info
 - scsi_read_dvd_struct_data_layer_desc, 312
- layer_number
 - scsi_read_dvd_structure, 324
- lb_count
 - scsi_sync_cache, 381
- lb_per_sec
 - cd_audio_page, 116
- lba
 - scsi_report_key, 336
- LEADOUT
 - scsi_cd.c, 662
- LEFT_CHANNEL
 - scsi_cd.h, 701
- LEFT_PORT
 - scsi_cd.h, 701
- len
 - scsi_read_element_status, 326
 - scsi_request_volume_element_address, 350
 - scsi_verify, 388
 - scsi_write_and_verify, 390
- length
 - scsi_inquiry, 228
 - scsi_log_select, 238
 - scsi_log_sense, 239
 - scsi_mode_select_10, 265
 - scsi_mode_select_6, 266
 - scsi_mode_sense_10, 267
 - scsi_mode_sense_6, 268
 - scsi_read_buffer, 288
 - scsi_read_capacity_data, 291
 - scsi_read_capacity_data_long, 292
 - scsi_read_defect_data_hdr_10, 299
 - scsi_read_defect_data_hdr_12, 300
 - scsi_reassign_blocks_data, 334
 - scsi_release, 335
 - scsi_report_luns, 346

- scsi_report_luns_data, 348
- scsi_request_sense, 349
- scsi_reserve, 352
- scsi_rw_10, 356
- scsi_rw_12, 358
- scsi_rw_16, 360
- scsi_rw_6, 362
- scsi_sa_rw, 363
- scsi_sense, 371
- scsi_vpd_unit_serial_number, 389
- scsi_write_buffer, 391
- length_0
 - scsi_read_cd_cap_data, 293
- length_1
 - scsi_read_cd_cap_data, 293
- length_2
 - scsi_read_cd_cap_data, 293
- length_3
 - scsi_read_cd_cap_data, 294
- li
 - slccb, 408
- li_cfgflags
 - lun_info, 165
- li_disc
 - lun_info, 165
- li_discq
 - lun_info, 165
- li_diskflags
 - lun_info, 165
- li_flags
 - lun_info, 165
- li_flags_valid
 - lun_info, 165
- li_inq
 - lun_info, 165
- li_lun
 - lun_info, 165
- li_maxnexus
 - lun_info, 166
- li_maxnqio
 - lun_info, 166
- li_nqio
 - lun_info, 166
- li_qd
 - lun_info, 166
- li_qflags
 - lun_info, 166
- li_qtagarray
 - lun_info, 166
- li_qtagbits
 - lun_info, 166
- li_quirks
 - lun_info, 166
- li_sloi
 - lun_info, 166
- li_sms
 - lun_info, 167
- li_state
 - lun_info, 167
- li_ti
 - lun_info, 167
- LIST_ENTRY
 - camq_entry, 45
 - lun_info, 165
 - scsi_low_softc, 253
- LIST_HEAD
 - cam_queue.h, 507
 - scsi_low.h, 858
- LOAD_UNLOAD
 - scsi_sa.h, 938
- LOCATE
 - scsi_sa.h, 938
- LOG_SELECT
 - scsi_all.h, 617
- LOG_SENSE
 - scsi_all.h, 617
- long_handle
 - cdchanger, 132
- lun_id
 - ioc_enable_lun, 163
- lun_id_t
 - cam.h, 438
- lun_imm_long
 - scsi_erase, 222
- lun_info, 164
 - li_cfgflags, 165
 - li_disc, 165
 - li_discq, 165
 - li_diskflags, 165
 - li_flags, 165
 - li_flags_valid, 165
 - li_inq, 165
 - li_lun, 165
 - li_maxnexus, 166
 - li_maxnqio, 166
 - li_nqio, 166
 - li_qd, 166
 - li_qflags, 166
 - li_qtagarray, 166
 - li_qtagbits, 166
 - li_quirks, 166
 - li_sloi, 166
 - li_sms, 167
 - li_state, 167
 - li_ti, 167
 - LIST_ENTRY, 165
- lun_info::scsi_low_inq_data, 168
 - sd_len, 168

- sd_resp, 168
- sd_sp1, 168
- sd_sp2, 168
- sd_support, 168
- sd_type, 168
- sd_version, 168
- lun_info::scsi_low_mode_sense_data, 170
 - cmp_eca, 170
 - cmp_length, 170
 - cmp_page, 170
 - cmp_qc, 170
 - cmp_rlec, 170
 - cmp_spare, 170
 - sms_cmp, 171
 - sms_header, 171
- lun_thirdparty
 - scsi_reserve_release_unit, 353
- lundata
 - scsi_report_luns_data, 348
- luns
 - scsi_report_luns_data, 348
- M
 - scsi_all.c, 591
- main_data_end
 - scsi_read_dvd_struct_data_layer_desc, 313
- main_data_start
 - scsi_read_dvd_struct_data_layer_desc, 313
- MALLOC_DEFINE
 - cam_periph.c, 478
 - cam_queue.c, 498, 499
 - cam_sim.c, 510
 - cam_xpt.c, 528
 - scsi_low.c, 800
 - scsi_sa.c, 913
 - scsi_ses.c, 966
 - scsi_targ_bh.c, 997
 - scsi_target.c, 1007
- manuf_id_11_6
 - scsi_read_dvd_struct_data_lead_in, 315
- manuf_id_17_12
 - scsi_read_dvd_struct_data_lead_in, 315
- manuf_id_5_0
 - scsi_read_dvd_struct_data_lead_in, 315
- manuf_info
 - scsi_read_dvd_struct_data_manufacturer, 317
- mapinfo
 - targ_cmd_descr, 410
- mask
 - scsi_low.h, 865
- match_buf_len
 - ccb_dev_match, 58
- match_pattern, 172
 - bus_pattern, 172
 - device_pattern, 172
 - periph_pattern, 172
- match_result, 173
 - bus_result, 173
 - device_result, 173
 - periph_result, 173
- matches
 - ccb_dev_match, 58
- MAX_ACCEPT
 - scsi_targ_bh.c, 996
- max_blk
 - sa_softc, 202
- MAX_BUF_SIZE
 - scsi_targ_bh.c, 996
- max_dev_openings
 - cam_sim, 40
- MAX_IMMEDIATE
 - scsi_targ_bh.c, 996
- max_lun
 - ccb_pathinq, 85
- max_size
 - targbh_cmd_desc, 420
- max_tagged_dev_openings
 - cam_sim, 40
- max_target
 - ccb_pathinq, 85
- maximum
 - scsi_read_block_limits_data, 287
- maxtags
 - ccb_getdevstats, 75
 - xpt_quirk_entry, 425
- maxvsn
 - scsi_send_volume_tag_parameters, 370
- md_condition
 - scsi_low_msgout_data, 246
- md_errfunc
 - scsi_low_msgout_data, 246
- md_flags
 - scsi_low_msgout_data, 246
- md_len
 - scsi_low_msgin_data, 245
- md_msg
 - scsi_low_msgout_data, 246
- md_msgfunc
 - scsi_low_msgin_data, 245
 - scsi_low_msgout_data, 246
- media_blksize
 - sa_softc, 203
- media_density
 - sa_softc, 203
- media_numblks
 - sa_softc, 203
- media_specific
 - scsi_read_dvd_struct_data_layer_desc, 313

- media_type
 - scsi_inquiry_pattern, 234
 - scsi_static_inquiry_pattern, 379
- medium_rot_rate_0
 - disk_pages::flexible_disk_page, 147
 - disk_pages::rigid_geometry_page, 154
- medium_rot_rate_1
 - disk_pages::flexible_disk_page, 147
 - disk_pages::rigid_geometry_page, 154
- medium_type
 - scsi_mode_hdr_10, 260
 - scsi_mode_hdr_6, 261
 - scsi_mode_header_10, 262
 - scsi_mode_header_6, 263
- member
 - page_transport_geometry_parameters, 179
- MEMCPY
 - scsi_ses.c, 958
- MEMZERO
 - scsi_ses.c, 958
- message_args
 - ccb_immed_notify, 80
- microp
 - cam_xpt.c, 567
 - scsi_da.c, 771
- min_blk
 - sa_softc, 203
- minimum
 - scsi_read_block_limits_data, 287
- mintags
 - ccb_getdevstats, 75
 - xpt_quirk_entry, 425
- minute
 - scsi_read_dvd_struct_data_disc_id, 306
- minvsn
 - scsi_send_volume_tag_parameters, 370
- misc
 - page_transport_geometry_parameters, 179
- MKMSG_EXTEND
 - scsi_low.h, 839
- mode_buf
 - cd_mode_params, 122
- mode_data_10
 - cd_mode_data_6_10, 121
- mode_data_6
 - cd_mode_data_6_10, 121
- mode_devs
 - sa_devs, 196
- MODE_SELECT
 - scsi_da.h, 778
- MODE_SELECT_10
 - scsi_all.h, 617
- MODE_SELECT_6
 - scsi_all.h, 617
- MODE_SENSE
 - scsi_da.h, 778
- MODE_SENSE_10
 - scsi_all.h, 617
- MODE_SENSE_6
 - scsi_all.h, 618
- MODULE_DEPEND
 - scsi_low_pisa.c, 868
- MODULE_VERSION
 - cam_xpt.c, 528
 - scsi_low_pisa.c, 868
- month
 - scsi_read_dvd_struct_data_disc_id, 306
- motor_off_del
 - disk_pages::flexible_disk_page, 147
- motor_on_del
 - disk_pages::flexible_disk_page, 147
- move_from
 - page_device_capabilities, 175
- MOVE_MEDIUM
 - scsi_all.h, 618
 - scsi_ch.h, 745
- MOVE_MEDIUM_INVERT
 - scsi_ch.h, 745
- MOVE_TO_DT
 - scsi_ch.h, 745
- MOVE_TO_IE
 - scsi_ch.h, 745
- MOVE_TO_MT
 - scsi_ch.h, 745
- MOVE_TO_ST
 - scsi_ch.h, 745
- msg
 - scsi_low.h, 865
 - scsi_low_msg_log, 244
- MSG_ABORT
 - scsi_low.h, 839
 - scsi_message.h, 871
- MSG_ABORT_QTAG
 - scsi_low.h, 839
- MSG_ABORT_TAG
 - scsi_message.h, 871
- MSG_ABORT_TASK
 - scsi_message.h, 871
- MSG_ABORT_TASK_SET
 - scsi_message.h, 871
- MSG_ACA_TASK
 - scsi_message.h, 871
- MSG_BUS_DEV_RESET
 - scsi_message.h, 872
- MSG_CLEAR_ACA
 - scsi_message.h, 872
- MSG_CLEAR_QTAG
 - scsi_low.h, 839

- MSG_CLEAR_QUEUE
 - scsi_message.h, 872
- MSG_CLEAR_TASK_SET
 - scsi_message.h, 872
- MSG_CMDCOMPLETE
 - scsi_message.h, 872
- MSG_COMP
 - scsi_low.h, 839
- MSG_DISCON
 - scsi_low.h, 839
- MSG_DISCONNECT
 - scsi_message.h, 872
- MSG_EXT_PPR
 - scsi_message.h, 872
- MSG_EXT_PPR_DT_REQ
 - scsi_message.h, 872
- MSG_EXT_PPR_HOLD_MCS
 - scsi_message.h, 872
- MSG_EXT_PPR_IU_REQ
 - scsi_message.h, 872
- MSG_EXT_PPR_LEN
 - scsi_message.h, 872
- MSG_EXT_PPR_PCOMP_EN
 - scsi_message.h, 873
- MSG_EXT_PPR_QAS_REQ
 - scsi_message.h, 873
- MSG_EXT_PPR_RD_STRM
 - scsi_message.h, 873
- MSG_EXT_PPR_RTI
 - scsi_message.h, 873
- MSG_EXT_PPR_WR_FLOW
 - scsi_message.h, 873
- MSG_EXT_SDTR
 - scsi_message.h, 873
- MSG_EXT_SDTR_LEN
 - scsi_message.h, 873
- MSG_EXT_WDTR
 - scsi_message.h, 873
- MSG_EXT_WDTR_BUS_16_BIT
 - scsi_message.h, 873
- MSG_EXT_WDTR_BUS_32_BIT
 - scsi_message.h, 873
- MSG_EXT_WDTR_BUS_8_BIT
 - scsi_message.h, 874
- MSG_EXT_WDTR_LEN
 - scsi_message.h, 874
- MSG_EXTEND
 - scsi_low.h, 839
- MSG_EXTEND_MDPCODE
 - scsi_low.h, 839
- MSG_EXTEND_MDPLEN
 - scsi_low.h, 839
- MSG_EXTEND_SYNCHCODE
 - scsi_low.h, 840
- MSG_EXTEND_SYNCHLEN
 - scsi_low.h, 840
- MSG_EXTEND_WIDECODE
 - scsi_low.h, 840
- MSG_EXTEND_WIDELEN
 - scsi_low.h, 840
- MSG_EXTENDED
 - scsi_message.h, 874
- MSG_HEAD_OF_Q_TAG
 - scsi_message.h, 874
- MSG_HEAD_OF_QUEUE_TASK
 - scsi_message.h, 874
- MSG_HEAD_QTAG
 - scsi_low.h, 840
- MSG_I_ERROR
 - scsi_low.h, 840
- MSG_IDENTIFY
 - scsi_low.h, 840
 - scsi_message.h, 874
- MSG_IDENTIFY_DISCFLAG
 - scsi_message.h, 874
- MSG_IDENTIFY_DISCPRIV
 - scsi_low.h, 840
- MSG_IDENTIFY_LUNMASK
 - scsi_message.h, 874
- MSG_IDENTIFYFLAG
 - scsi_message.h, 874
- MSG_IGN_WIDE_RESIDUE
 - scsi_message.h, 874
- MSG_INIT_RECOVERY
 - scsi_message.h, 874
- MSG_INITIATOR_DET_ERR
 - scsi_message.h, 875
- MSG_ISIDENTIFY
 - scsi_message.h, 875
- MSG_LCOMP
 - scsi_low.h, 840
- MSG_LCOMP_F
 - scsi_low.h, 840
- msg_len
 - ccb_scsiio, 100
- MSG_LINK_CMD_COMPLETE
 - scsi_message.h, 875
- MSG_LINK_CMD_COMPLETEF
 - scsi_message.h, 875
- MSG_LOGICAL_UNIT_RESET
 - scsi_message.h, 875
- MSG_MESSAGE_REJECT
 - scsi_message.h, 875
- MSG_NOOP
 - scsi_low.h, 840
 - scsi_message.h, 875
- MSG_ORDERED_Q_TAG
 - scsi_message.h, 875

- MSG_ORDERED_QTAG
 - scsi_low.h, 840
- MSG_ORDERED_TASK
 - scsi_message.h, 875
- MSG_PARITY
 - scsi_low.h, 841
- MSG_PARITY_ERROR
 - scsi_message.h, 875
- msg_ptr
 - ccb_scsiio, 100
- MSG_QAS_REQUEST
 - scsi_message.h, 875
- MSG_REJECT
 - scsi_low.h, 841
- MSG_REL_RECOVERY
 - scsi_message.h, 876
- MSG_RELEASE_ATN
 - scsi_low.c, 794
- MSG_RESET
 - scsi_low.h, 841
- MSG_RESTOREPOINTERS
 - scsi_message.h, 876
- MSG_RESTORESP
 - scsi_low.h, 841
- MSG_SAVEDATAPOINTER
 - scsi_message.h, 876
- MSG_SAVESP
 - scsi_low.h, 841
- MSG_SIMPLE_Q_TAG
 - scsi_message.h, 876
- MSG_SIMPLE_QTAG
 - scsi_low.h, 841
- MSG_SIMPLE_TASK
 - scsi_message.h, 876
- MSG_TARGET_RESET
 - scsi_message.h, 876
- MSG_TASK_COMPLETE
 - scsi_message.h, 876
- MSG_TERM_IO
 - scsi_low.h, 841
- MSG_TERM_IO_PROC
 - scsi_message.h, 876
- MSGCMD_LUN
 - scsi_low.c, 794
- MSGERR
 - scsi_low.h, 841
- MSGIN_DATA_LAST
 - scsi_low.c, 794
- MSGIN_OFFSET
 - scsi_low.c, 794
- MSGIN_PERIOD
 - scsi_low.c, 794
- MSGIN_WIDTHP
 - scsi_low.c, 794
- MSGINPTR_CLR
 - scsi_low.c, 795
- MSGPH_ABORT
 - scsi_low.h, 841
- MSGPH_CMDC
 - scsi_low.h, 841
- MSGPH_DISC
 - scsi_low.h, 841
- MSGPH_LCTERM
 - scsi_low.h, 841
- MSGPH_NULL
 - scsi_low.h, 841
- MSGPH_RESET
 - scsi_low.h, 841
- MSGPH_TERM
 - scsi_low.h, 842
- mtea
 - page_element_address_assignment, 177
- Nalarm
 - scfg, 207
- nblocks
 - format_capacity_descriptor, 159
 - scsi_mode_blk_desc, 258
- nbufblk
 - scsi_tape_position_data, 384
- nbufbyte
 - scsi_tape_position_data, 384
- nbytes
 - read_element_status_header, 192
 - read_element_status_page_header, 193
- ncyl_0
 - disk_pages::flexible_disk_page, 148
 - disk_pages::rigid_geometry_page, 154
- ncyl_1
 - disk_pages::flexible_disk_page, 148
 - disk_pages::rigid_geometry_page, 154
- ncyl_2
 - disk_pages::rigid_geometry_page, 154
- ndte
 - page_element_address_assignment, 177
- next_ccb
 - ccb_scsiio, 100
- nexus_conflict
 - scsi_low_statics, 257
- nexus_disconnected
 - scsi_low_statics, 257
- nexus_fail
 - scsi_low_statics, 257
- nexus_reselected
 - scsi_low_statics, 257
- nexus_win
 - scsi_low_statics, 257
- Nfans

- scfg, 207
- nheads
 - disk_pages::flexible_disk_page, 148
 - disk_pages::rigid_geometry_page, 154
- niece
 - page_element_address_assignment, 178
- nmte
 - page_element_address_assignment, 178
- no_lun_inq_data
 - scsi_targ_bh.c, 1002
- no_lun_sense_data
 - scsi_targ_bh.c, 1002
- notify_user
 - scsi_target.c, 1007
- notreviewed.dox, 429
- now
 - scsi_low.h, 865
- nperiph_drivers
 - cam_periph.c, 479
- NPSEUDO_ALARM
 - scsi_ses.c, 958
- NPSEUDO_THERM
 - scsi_ses.c, 958
- Npwr
 - scfg, 207
- nr_dev
 - sa_devs::sa_mode_devs, 197
- nse
 - page_element_address_assignment, 178
- Nslots
 - scfg, 208
- Nspkrs
 - scfg, 208
- Nsubenc
 - SesCfgHdr, 400
- Ntherm
 - scfg, 208
- Ntypes
 - SesEncHdr, 403
- num_ascs
 - scsi_sense_quirk_entry, 374
- num_blocks
 - scsi_mode_block_descr, 259
- num_bufs_used
 - cam_periph_map_info, 37
- num_cam_status_entries
 - cam.c, 433
 - cam.h, 442
- num_changers
 - scsi_cd.c, 694
- num_devices
 - cdchanger, 133
- num_marks
 - scsi_write_filemarks, 392
- num_matches
 - ccb_dev_match, 58
- num_ops
 - scsi_op_quirk_entry, 274
- num_patterns
 - ccb_dev_match, 59
- num_sense_keys
 - scsi_sense_quirk_entry, 374
- O
 - scsi_all.c, 591
- obj_id
 - ses_hlptxt, 395
 - ses_object, 396
 - ses_objstat, 397
- obj_text
 - ses_hlptxt, 395
- object_type
 - ses_object, 396
- offset
 - scsi_read_buffer, 288
 - scsi_write_buffer, 391
 - targ_info::synch, 416
- op_code
 - scsi_pause, 275
 - scsi_play_10, 276
 - scsi_play_12, 277
 - scsi_play_msf, 278
 - scsi_play_rel_12, 280
 - scsi_play_track, 281
 - scsi_read_cd_capacity, 295
 - scsi_read_header, 329
 - scsi_read_subchannel, 330
 - scsi_read_toc, 332
- op_table
 - scsi_op_quirk_entry, 274
- op_table_entry, 174
 - desc, 174
 - opcode, 174
 - opmask, 174
- opcode
 - op_table_entry, 174
 - scsi_changedef, 209
 - scsi_erase, 222
 - scsi_exchange_medium, 223
 - scsi_format_unit, 225
 - scsi_generic, 226
 - scsi_initialize_element_status, 227
 - scsi_inquiry, 228
 - scsi_load_unload, 235
 - scsi_log_select, 238
 - scsi_log_sense, 239
 - scsi_mode_select_10, 265
 - scsi_mode_select_6, 266

- scsi_mode_sense_10, 267
- scsi_mode_sense_6, 268
- scsi_move_medium, 272
- scsi_position_to_element, 283
- scsi_prevent, 285
- scsi_read_block_limits, 286
- scsi_read_buffer, 288
- scsi_read_capacity, 289
- scsi_read_capacity_16, 290
- scsi_read_defect_data_10, 297
- scsi_read_defect_data_12, 298
- scsi_read_dvd_structure, 324
- scsi_read_element_status, 326
- scsi_read_format_capacities, 328
- scsi_reassign_blocks, 333
- scsi_release, 335
- scsi_report_key, 336
- scsi_report_luns, 346
- scsi_request_sense, 349
- scsi_request_volume_element_address, 350
- scsi_reserve, 352
- scsi_reserve_release_unit, 353
- scsi_rewind, 354
- scsi_rezero_unit, 355
- scsi_rw_10, 356
- scsi_rw_12, 358
- scsi_rw_16, 360
- scsi_rw_6, 362
- scsi_sa_rw, 363
- scsi_send_diag, 364
- scsi_send_key, 365
- scsi_send_receive, 367
- scsi_send_volume_tag, 368
- scsi_sense, 371
- scsi_set_speed, 376
- scsi_space, 377
- scsi_start_stop_unit, 378
- scsi_sync_cache, 381
- scsi_tape_locate, 382
- scsi_tape_read_position, 386
- scsi_test_unit_ready, 387
- scsi_verify, 388
- scsi_write_and_verify, 390
- scsi_write_buffer, 391
- scsi_write_filemarks, 392
- open_pending_mount
 - sa_softc, 203
- open_rdonly
 - sa_softc, 203
- openings
 - ccb_relsim, 94
- opmask
 - op_table_entry, 174
- optimum_write_strategy
 - scsi_read_dvd_struct_data_lead, 315
- osdep
 - slccb, 408
- P
 - scsi_all.c, 591
- PA_PAUSE
 - scsi_cd.h, 701
- PA_RESUME
 - scsi_cd.h, 701
- page
 - cd_mode_data, 119
 - cd_mode_data_10, 120
 - cd_page_sizes, 123
 - scsi_log_header, 236
 - scsi_log_select, 238
 - scsi_log_sense, 239
 - scsi_mode_sense_10, 267
 - scsi_mode_sense_6, 268
- page_code
 - cd_audio_page, 116
 - scsi_control_page, 211
 - scsi_da_rw_recovery_page, 213
 - scsi_data_compression_page, 215
 - scsi_inquiry, 228
 - scsi_mode_page_header, 264
 - scsi_vpd_unit_serial_number, 389
- page_device_capabilities, 175
 - exchange_with, 175
 - move_from, 175
 - pg_code, 175
 - pg_length, 175
 - reserved0, 175
 - reserved1, 175
 - stor, 175
- page_element_address_assignment, 177
 - fdtea, 177
 - fieea, 177
 - fsea, 177
 - mtea, 177
 - ndte, 177
 - niece, 178
 - nmte, 178
 - nse, 178
 - pg_code, 178
 - pg_length, 178
 - reserved, 178
- page_length
 - scsi_control_page, 211
 - scsi_da_rw_recovery_page, 213
 - scsi_data_compression_page, 215
 - scsi_mode_page_header, 264
- page_size
 - cd_page_sizes, 123

- page_transport_geometry_parameters, 179
 - member, 179
 - misc, 179
 - pg_code, 179
 - pg_length, 179
- pagecode
 - sa_comp_t, 194
 - scsi_dev_conf_page, 221
- pagelength
 - sa_comp_t, 194
 - scsi_dev_conf_page, 221
- pages
 - scsi_mode_sense_data, 270
- param_code
 - scsi_log_param_header, 237
- param_control
 - scsi_log_param_header, 237
- param_len
 - cd_audio_page, 117
 - scsi_log_param_header, 237
 - scsi_send_key, 365
- paramlen
 - scsi_send_diag, 364
- paramptr
 - scsi_log_sense, 239
- PARITYERR
 - scsi_low.h, 842
- partition
 - scsi_tape_locate, 382
 - scsi_tape_position_data, 384
- PASS_CCB_BUFFER_IO
 - scsi_pass.c, 879
- pass_ccb_types
 - scsi_pass.c, 879
- PASS_CCB_WAITING
 - scsi_pass.c, 879
- pass_cdevsw
 - scsi_pass.c, 886
- PASS_FLAG_INVALID
 - scsi_pass.c, 880
- PASS_FLAG_LOCKED
 - scsi_pass.c, 880
- PASS_FLAG_OPEN
 - scsi_pass.c, 880
- pass_flags
 - scsi_pass.c, 880
- pass_softc, 180
 - dev, 180
 - device_stats, 180
 - flags, 180
 - pd_type, 180
 - saved_ccb, 180
 - state, 181
- pass_state
 - scsi_pass.c, 880
- PASS_STATE_NORMAL
 - scsi_pass.c, 880
- passasync
 - scsi_pass.c, 880
- passcleanup
 - scsi_pass.c, 881, 886
- passclose
 - scsi_pass.c, 881, 886
- passdone
 - scsi_pass.c, 881
- passdriver
 - scsi_pass.c, 887
- passerror
 - scsi_pass.c, 882
- passinit
 - scsi_pass.c, 883, 887
- passioctl
 - scsi_pass.c, 883, 887
- passoninvalidate
 - scsi_pass.c, 884, 887
- passopen
 - scsi_pass.c, 884, 887
- passregister
 - scsi_pass.c, 885, 887
- passendccb
 - scsi_pass.c, 885
- passstart
 - scsi_pass.c, 886, 887
- pat_length
 - format_ipat_descriptor, 162
- path
 - cam_periph, 34
 - ccb_hdr, 77
 - scsi_low_osdep_interface, 248
 - targ_softc, 417
- path_id
 - bus_match_pattern, 22
 - bus_match_result, 23
 - cam_sim, 40
 - ccb_hdr, 77
 - cdchanger, 133
 - device_match_pattern, 141
 - device_match_result, 143
 - ioc_enable_lun, 163
 - periph_match_pattern, 183
 - periph_match_result, 184
- path_id_t
 - cam.h, 438
- PATHINQ_SETTINGS_SIZE
 - cam_ccb.h, 452
- pattern
 - dev_match_pattern, 139
- pattern_buf_len

- ccb_dev_match, 59
- pattern_type
 - format_ipat_descriptor, 162
- patterns
 - ccb_dev_match, 59
- PAUSE
 - scsi_cd.h, 701
- pd_type
 - pass_softc, 180
- PDMAERR
 - scsi_low.h, 842
- pdrv
 - ccb_dm_cookie, 61
- pdrv_ptr
 - ccb_eng_exec, 66
- pending_ccb_queue
 - targ_softc, 417
- PENDING_MOUNT_CHECK
 - scsi_sa.c, 908
- pending_queue
 - targbh_softc, 422
- PENDINGIO
 - scsi_low.h, 842
- perf_slotop
 - scsi_ses.c, 966
- period
 - scsi_all.c, 604
 - targ_info::synch, 416
- period_factor
 - scsi_all.c, 604
- periph
 - cam_path, 31
 - ccb_dm_cookie, 61
 - ses_softc, 398
 - targ_softc, 418
- periph_busy_delay
 - cam_periph.c, 479
- periph_ctor_t
 - cam_periph.h, 483
- periph_driver, 182
 - driver_name, 182
 - generation, 182
 - init, 182
 - TAILQ_HEAD, 182
- periph_drivers
 - cam_periph.c, 479
 - cam_periph.h, 491
- periph_dtor
 - cam_periph, 34
- periph_dtor_t
 - cam_periph.h, 483
- periph_init_func_t
 - cam_periph.h, 483
- periph_init_t
 - cam_periph.h, 484
- periph_is_queued
 - cam_xpt.c, 528
- periph_links
 - ccb_hdr, 77
- PERIPH_MATCH_ANY
 - cam_ccb.h, 459
- PERIPH_MATCH_LUN
 - cam_ccb.h, 459
- PERIPH_MATCH_NAME
 - cam_ccb.h, 459
- PERIPH_MATCH_NONE
 - cam_ccb.h, 459
- PERIPH_MATCH_PATH
 - cam_ccb.h, 459
- periph_match_pattern, 183
 - flags, 183
 - path_id, 183
 - periph_name, 183
 - target_id, 183
 - target_lun, 183
 - unit_number, 183
- periph_match_result, 184
 - path_id, 184
 - periph_name, 184
 - target_id, 184
 - target_lun, 184
 - unit_number, 184
- PERIPH_MATCH_TARGET
 - cam_ccb.h, 459
- PERIPH_MATCH_UNIT
 - cam_ccb.h, 459
- periph_name
 - cam_periph, 35
 - ccb_getdevlist, 73
 - periph_match_pattern, 183
 - periph_match_result, 184
- periph_noresrc_delay
 - cam_periph.c, 479
- periph_oninv_t
 - cam_periph.h, 484
- periph_oninval
 - cam_periph, 35
- periph_pattern
 - match_pattern, 172
- periph_pattern_flags
 - cam_ccb.h, 459
- periph_priv
 - ccb_hdr, 77
- periph_result
 - match_result, 173
- periph_selto_delay
 - cam_periph.c, 479
- periph_start

- cam_periph, 35
- periph_start_t
 - cam_periph.h, 484
- PERIPHDRIVER_DECLARE
 - cam_periph.h, 482
 - cam_xpt.c, 528
 - scsi_cd.c, 690
 - scsi_ch.c, 735
 - scsi_da.c, 768
 - scsi_pass.c, 886
 - scsi_pt.c, 892
 - scsi_sa.c, 913
 - scsi_ses.c, 967
 - scsi_targ_bh.c, 997
 - scsi_target.c, 1008
- periphdriver_register
 - cam_periph.c, 478
 - cam_periph.h, 491
- pg_code
 - disk_pages::flexible_disk_page, 148
 - disk_pages::format_device_page, 151
 - disk_pages::rigid_geometry_page, 154
 - page_device_capabilities, 175
 - page_element_address_assignment, 178
 - page_transport_geometry_parameters, 179
- pg_length
 - disk_pages::flexible_disk_page, 148
 - disk_pages::format_device_page, 151
 - disk_pages::rigid_geometry_page, 154
 - page_device_capabilities, 175
 - page_element_address_assignment, 178
 - page_transport_geometry_parameters, 179
- PGCODE_MASK
 - scsi_ch.h, 745
- PGCODE_PS
 - scsi_ch.h, 745
- PH_ARBSTART
 - scsi_low.h, 842
- PH_CMD
 - scsi_low.h, 842
- PH_DATA
 - scsi_low.h, 842
- PH_DISC
 - scsi_low.h, 842
- PH_MSGIN
 - scsi_low.h, 842
- PH_MSGOUT
 - scsi_low.h, 842
- PH_NULL
 - scsi_low.h, 843
- PH_RESEL
 - scsi_low.h, 843
- ph_sec_t_0
 - disk_pages::format_device_page, 152
- ph_sec_t_1
 - disk_pages::format_device_page, 152
- PH_SELECTED
 - scsi_low.h, 843
- PH_SELSTART
 - scsi_low.h, 843
- PH_STAT
 - scsi_low.h, 843
- phase
 - scsi_low.c, 827
- pi_inqflag
 - cam_ccb.h, 459
- PI_LINKED_CDB
 - cam_ccb.h, 459
- PI_MDP_ABLE
 - cam_ccb.h, 459
- pi_miscflag
 - cam_ccb.h, 459
- PI_SDTR_ABLE
 - cam_ccb.h, 459
- PI_SOFT_RST
 - cam_ccb.h, 459
- PI_TAG_ABLE
 - cam_ccb.h, 459
- pi_tmflag
 - cam_ccb.h, 460
- PI_WIDE_16
 - cam_ccb.h, 459
- PI_WIDE_32
 - cam_ccb.h, 459
- PIM_NO_6_BYTE
 - cam_ccb.h, 460
- PIM_NOBUSRESET
 - cam_ccb.h, 460
- PIM_NOINITIATOR
 - cam_ccb.h, 460
- PIM_NOREMOVE
 - cam_ccb.h, 460
- PIM_SCANHILO
 - cam_ccb.h, 460
- PIM_SEQSCAN
 - cam_ccb.h, 460
- pin32_pin2
 - disk_pages::flexible_disk_page, 148
- pin4_pint1
 - disk_pages::flexible_disk_page, 148
- pinfo
 - cam_periph, 35
 - ccb_hdr, 78
 - cd_softc, 127
- PIT_DISCONNECT
 - cam_ccb.h, 460
- PIT_GRP_6
 - cam_ccb.h, 460

- PIT_GRP_7
 - cam_ccb.h, 460
- PIT_PHASE
 - cam_ccb.h, 460
- PIT_PROCESSOR
 - cam_ccb.h, 460
- PIT_TERM_IO
 - cam_ccb.h, 460
- pkt_failures
 - scsi_status_iu_header, 380
- pkt_failures_length
 - scsi_status_iu_header, 380
- PLAY_10
 - scsi_cd.h, 701
- PLAY_12
 - scsi_cd.h, 702
- PLAY_MSOF
 - scsi_cd.h, 702
- PLAY_TRACK
 - scsi_cd.h, 702
- PLAY_TRACK_REL
 - scsi_cd.h, 702
- PLAY_TRACK_REL_BIG
 - scsi_cd.h, 702
- plexor_cd_ops
 - scsi_all.c, 604
- pll
 - scsi_send_volume_tag, 368
- PLURAL
 - scsi_ch.c, 721
- port
 - ccb_pathinq_settings_fc, 88
 - ccb_trans_settings_fc, 111
 - cd_audio_page, 117
- pos
 - ccb_dev_match, 59
- POSITION_TO_ELEMENT
 - scsi_all.h, 618
 - scsi_ch.h, 745
- POSITION_TO_ELEMENT_INVERT
 - scsi_ch.h, 746
- position_type
 - ccb_dev_position, 60
- ppr_options
 - ccb_pathinq_settings_spi, 90
 - ccb_trans_settings_spi, 114
- ppriv_field0
 - cam_periph.h, 483
- ppriv_field1
 - cam_periph.h, 483
- ppriv_ptr0
 - cam_periph.h, 483
- ppriv_ptr1
 - cam_periph.h, 483
- PR_ALLOW
 - scsi_all.h, 618
- PR_PREVENT
 - scsi_all.h, 618
- prefblk
 - sa_quirk_entry, 198
- PREVENT_ALLOW
 - scsi_all.h, 618
- PRINTF
 - scsi_ses.c, 958
- priority
 - cam_pinfo, 38
- priv
 - encobj, 157
- probe_action
 - cam_xpt.c, 525
- probe_driver
 - cam_xpt.c, 567
- PROBE_DV_EXIT
 - cam_xpt.c, 525
- probe_flags
 - cam_xpt.c, 525
- PROBE_FULL_INQUIRY
 - cam_xpt.c, 525
- PROBE_INQUIRY
 - cam_xpt.c, 525
- PROBE_INQUIRY_BASIC_DV1
 - cam_xpt.c, 525
- PROBE_INQUIRY_BASIC_DV2
 - cam_xpt.c, 525
- PROBE_INQUIRY_CKSUM
 - cam_xpt.c, 525
- PROBE_MODE_SENSE
 - cam_xpt.c, 525
- PROBE_NO_ANNOUNCE
 - cam_xpt.c, 525
- probe_periph_init
 - cam_xpt.c, 528
- PROBE_SERIAL_CKSUM
 - cam_xpt.c, 525
- PROBE_SERIAL_NUM
 - cam_xpt.c, 525
- probe_softc, 185
- PROBE_TUR
 - cam_xpt.c, 525
- PROBE_TUR_FOR_NEGOTIATION
 - cam_xpt.c, 525
- probecleanup
 - cam_xpt.c, 528
- probedone
 - cam_xpt.c, 528
- proberegister
 - cam_xpt.c, 529
- proberequestbackoff

- cam_xpt.c, 529
- proberequestdefaultnegotiation
 - cam_xpt.c, 530
- probeschedule
 - cam_xpt.c, 530
- probestart
 - cam_xpt.c, 531
- product
 - scsi_inquiry_data, 231
 - scsi_inquiry_pattern, 234
 - scsi_static_inquiry_pattern, 379
- prot_discid_data
 - scsi_read_dvd_struct_data_prot_discid, 320
- PROTO_ATA
 - cam_ccb.h, 455
- PROTO_ATAPI
 - cam_ccb.h, 455
- PROTO_SCSI
 - cam_ccb.h, 455
- proto_specific
 - ccb_trans_settings, 109
- PROTO_UNKNOWN
 - cam_ccb.h, 455
- PROTO_UNSPECIFIED
 - cam_ccb.h, 455
- PROTO_VERSION_UNKNOWN
 - cam_ccb.h, 452
- PROTO_VERSION_UNSPECIFIED
 - cam_ccb.h, 452
- protocol
 - ccb_pathinq, 85
 - ccb_trans_settings, 109
- protocol_version
 - ccb_pathinq, 85
 - ccb_trans_settings, 109
- PT_CCB_BUFFER_IO
 - scsi_pt.c, 892
- PT_CCB_BUFFER_IO_UA
 - scsi_pt.c, 892
- PT_CCB_RETRY_UA
 - scsi_pt.c, 892
- pt_ccb_state
 - scsi_pt.c, 892
- PT_CCB_WAITING
 - scsi_pt.c, 892
- pt_cdevsw
 - scsi_pt.c, 899
- PT_FLAG_DEVICE_INVALID
 - scsi_pt.c, 892
- PT_FLAG_NONE
 - scsi_pt.c, 892
- PT_FLAG_OPEN
 - scsi_pt.c, 892
- PT_FLAG_RETRY_UA
 - scsi_pt.c, 892
- scsi_pt.c, 892
- pt_flags
 - scsi_pt.c, 892
- pt_softc, 186
 - bio_queue, 186
 - device_stats, 186
- pt_state
 - scsi_pt.c, 892
- PT_STATE_NORMAL
 - scsi_pt.c, 892
- PT_STATE_PROBE
 - scsi_pt.c, 892
- ptasync
 - scsi_pt.c, 892
- ptclose
 - scsi_pt.c, 893, 899
- ptctor
 - scsi_pt.c, 893, 899
- ptdone
 - scsi_pt.c, 894
- ptdriver
 - scsi_pt.c, 899
- ptdtor
 - scsi_pt.c, 894, 899
- pterror
 - scsi_pt.c, 895
- ptinit
 - scsi_pt.c, 895, 899
- ptioctl
 - scsi_pt.c, 896, 899
- ptoninvalidate
 - scsi_pt.c, 896, 899
- ptopen
 - scsi_pt.c, 897, 900
- ptr
 - ccb_priv_entry, 93
- ptstart
 - scsi_pt.c, 897, 900
- ptstrategy
 - scsi_pt.c, 898, 900
- pvoltag
 - read_element_status_descriptor, 191
- pwroff
 - scfg, 208
- QFRLS
 - scsi_sa.c, 908
- qfrozen_cnt
 - camq, 43
 - ccb_relsim, 94
- quantum
 - cam_xpt.c, 568
 - scsi_da.c, 772
- quantum_fireball_entries

- scsi_all.c, 604
- queue
 - cam_ccbq, 25
- queue_array
 - camq, 44
- queue_cmp
 - cam_queue.c, 499
- queue_count
 - sa_softc, 203
- queue_flags
 - scsi_control_page, 211
- quirks
 - cd_quirk_entry, 126
 - ch_softc, 134
 - da_quirk_entry, 137
 - sa_quirk_entry, 198
 - sa_softc, 203
 - xpt_quirk_entry, 425
- R
 - scsi_all.c, 592
- r_dev
 - sa_devs::sa_mode_devs, 197
- ram_swi_info
 - scsi_read_dvd_struct_data_medium_status, 318
- random_num
 - scsi_read_dvd_struct_data_disc_id, 306
- rb_empty_ratio
 - scsi_dev_conf_page, 221
- RBL_GRAN
 - scsi_sa.h, 938
- RBL_GRAN_MASK
 - scsi_sa.h, 938
- RDS_FORMAT_BCA
 - scsi_cd.h, 702
- RDS_FORMAT_CMGS_CPM
 - scsi_cd.h, 702
- RDS_FORMAT_COPYRIGHT
 - scsi_cd.h, 702
- RDS_FORMAT_DCB
 - scsi_cd.h, 702
- RDS_FORMAT_DDS
 - scsi_cd.h, 702
- RDS_FORMAT_DISC_ID
 - scsi_cd.h, 703
- RDS_FORMAT_DISC_KEY
 - scsi_cd.h, 703
- RDS_FORMAT_DISC_KEY_BLOCK
 - scsi_cd.h, 703
- RDS_FORMAT_DVDRAM_MEDIA_STAT
 - scsi_cd.h, 703
- RDS_FORMAT_LEADIN
 - scsi_cd.h, 703
- RDS_FORMAT_MANUFACTURER
 - scsi_cd.h, 703
- RDS_FORMAT_PHYSICAL
 - scsi_cd.h, 703
- RDS_FORMAT_PROT_DISCID
 - scsi_cd.h, 703
- RDS_FORMAT_RMD
 - scsi_cd.h, 703
- RDS_FORMAT_RMD_BORDEROUT
 - scsi_cd.h, 703
- RDS_FORMAT_SPARE_AREA
 - scsi_cd.h, 703
- RDS_FORMAT_STRUCTURE_LIST
 - scsi_cd.h, 704
- RDS_FORMAT_WRITE_PROT
 - scsi_cd.h, 704
- RDSD_ACTION_FORMAT
 - scsi_cd.h, 704
- RDSD_ACTION_MODIFY_DCB
 - scsi_cd.h, 704
- RDSD_ACTION_READING
 - scsi_cd.h, 704
- RDSD_ACTION_RECORDING
 - scsi_cd.h, 704
- RDSD_BCA
 - scsi_cd.h, 704
- RDSD_BCA_MASK
 - scsi_cd.h, 704
- RDSD_BCA_SHIFT
 - scsi_cd.h, 704
- RDSD_BOOK_TYPE_DVD_PRW
 - scsi_cd.h, 704
- RDSD_BOOK_TYPE_DVD_R
 - scsi_cd.h, 704
- RDSD_BOOK_TYPE_DVD_RAM
 - scsi_cd.h, 705
- RDSD_BOOK_TYPE_DVD_ROM
 - scsi_cd.h, 705
- RDSD_BOOK_TYPE_DVD_RW
 - scsi_cd.h, 705
- RDSD_BOOK_TYPE_MASK
 - scsi_cd.h, 705
- RDSD_BOOK_TYPE_SHIFT
 - scsi_cd.h, 705
- RDSD_BOOK_VERSION_MASK
 - scsi_cd.h, 705
- RDSD_CMGS_COPY_ALLOWED
 - scsi_cd.h, 705
- RDSD_CMGS_MASK
 - scsi_cd.h, 705
- RDSD_CMGS_NO_COPIES
 - scsi_cd.h, 705
- RDSD_CMGS_ONE_COPY
 - scsi_cd.h, 705

- RDS_D_CPM_HAS_COPYRIGHT
scsi_cd.h, 706
- RDS_D_CPM_MASK
scsi_cd.h, 706
- RDS_D_CPM_NO_COPYRIGHT
scsi_cd.h, 706
- RDS_D_CPS_DATA_EXISTS
scsi_cd.h, 706
- RDS_D_CPS_NOT_PRESENT
scsi_cd.h, 706
- RDS_D_DISC_SIZE_120MM
scsi_cd.h, 706
- RDS_D_DISC_SIZE_80MM
scsi_cd.h, 706
- RDS_D_DISC_SIZE_MASK
scsi_cd.h, 706
- RDS_D_DISC_SIZE_SHIFT
scsi_cd.h, 706
- RDS_D_DT_NEED_CARTRIDGE
scsi_cd.h, 706
- RDS_D_DT_NO_CART_NEEDED
scsi_cd.h, 706
- RDS_D_LAYER_TYPE_MASK
scsi_cd.h, 707
- RDS_D_LAYER_TYPE_RECORD
scsi_cd.h, 707
- RDS_D_LAYER_TYPE_RO
scsi_cd.h, 707
- RDS_D_LAYER_TYPE_RW
scsi_cd.h, 707
- RDS_D_LIN_DENSITY_0267
scsi_cd.h, 707
- RDS_D_LIN_DENSITY_0280_0291
scsi_cd.h, 707
- RDS_D_LIN_DENSITY_0293
scsi_cd.h, 707
- RDS_D_LIN_DENSITY_0353
scsi_cd.h, 707
- RDS_D_LIN_DENSITY_0409_0435
scsi_cd.h, 707
- RDS_D_LIN_DENSITY_MASK
scsi_cd.h, 707
- RDS_D_LIN_DENSITY_SHIFT
scsi_cd.h, 707
- RDS_D_MAIN_DATA_START_DVD_RO
scsi_cd.h, 708
- RDS_D_MAIN_DATA_START_DVD_RW
scsi_cd.h, 708
- RDS_D_MAX_RATE_0252
scsi_cd.h, 708
- RDS_D_MAX_RATE_0504
scsi_cd.h, 708
- RDS_D_MAX_RATE_1008
scsi_cd.h, 708
- RDS_D_MAX_RATE_MASK
scsi_cd.h, 708
- RDS_D_MAX_RATE_NOT_SPEC
scsi_cd.h, 708
- RDS_D_MS_CARTRIDGE
scsi_cd.h, 708
- RDS_D_MS_CWP
scsi_cd.h, 708
- RDS_D_MS_MSWI
scsi_cd.h, 708
- RDS_D_MS_OUT
scsi_cd.h, 708
- RDS_D_MS_PWP
scsi_cd.h, 709
- RDS_D_NL_ONE_LAYER
scsi_cd.h, 709
- RDS_D_NL_TWO_LAYERS
scsi_cd.h, 709
- RDS_D_NUM_LAYERS_MASK
scsi_cd.h, 709
- RDS_D_NUM_LAYERS_SHIFT
scsi_cd.h, 709
- RDS_D_RDS_MASK
scsi_cd.h, 709
- RDS_D_RDS_NOT_READABLE
scsi_cd.h, 709
- RDS_D_RDS_READABLE
scsi_cd.h, 709
- RDS_D_SDS_MASK
scsi_cd.h, 709
- RDS_D_SDS_NOT_WRITEABLE
scsi_cd.h, 709
- RDS_D_SDS_WRITEABLE
scsi_cd.h, 709
- RDS_D_SWI_NO_BARE
scsi_cd.h, 710
- RDS_D_SWI_UNSPEC
scsi_cd.h, 710
- RDS_D_TP_OTP
scsi_cd.h, 710
- RDS_D_TP_PTP
scsi_cd.h, 710
- RDS_D_TRACK_DENSITY_0615
scsi_cd.h, 710
- RDS_D_TRACK_DENSITY_074
scsi_cd.h, 710
- RDS_D_TRACK_DENSITY_080
scsi_cd.h, 710
- RDS_D_TRACK_DENSITY_MASK
scsi_cd.h, 710
- RDS_D_TRACK_PATH_MASK
scsi_cd.h, 710
- RDS_D_TRACK_PATH_SHIFT
scsi_cd.h, 710

- RDS_D_WPS_CWP
 - scsi_cd.h, 711
- RDS_D_WPS_MS_WI
 - scsi_cd.h, 711
- RDS_D_WPS_PWP
 - scsi_cd.h, 711
- RDS_D_WPS_SWPP
 - scsi_cd.h, 711
- READ_10
 - scsi_all.h, 618
- READ_12
 - scsi_all.h, 618
- READ_16
 - scsi_all.h, 619
- READ_6
 - scsi_all.h, 619
- READ_BLOCK_LIMITS
 - scsi_sa.h, 938
- READ_BUFFER
 - scsi_all.h, 619
- READ_CAPACITY
 - scsi_all.h, 619
- READ_CD_CAPACITY
 - scsi_cd.h, 711
- READ_DEFECT_DATA_10
 - scsi_da.h, 778
- READ_DEFECT_DATA_12
 - scsi_da.h, 778
- read_dvd_struct_data_list, 187
 - data_len, 187
 - entries, 187
 - reserved, 187
- read_dvd_struct_list_entry, 188
 - format_code, 188
 - sds_rds, 188
 - struct_len, 188
- read_dvd_struct_write_prot, 189
 - data_len, 189
 - reserved, 189
 - reserved0, 189
 - write_prot_status, 189
- READ_DVD_STRUCTURE
 - scsi_cd.h, 711
- READ_ELEMENT_STATUS
 - scsi_all.h, 619
 - scsi_ch.h, 746
- READ_ELEMENT_STATUS_ACCESS
 - scsi_ch.h, 746
- READ_ELEMENT_STATUS_AVOLTAG
 - scsi_ch.h, 746
- read_element_status_descriptor, 190
 - avoltag, 190
 - dt_scsi_addr, 190
 - dt_scsi_flags, 190
 - eaddr, 190
 - flags1, 191
 - flags2, 191
 - pvoltag, 191
 - reserved0, 191
 - reserved1, 191
 - sense_code, 191
 - sense_qual, 191
 - ssea, 191
- READ_ELEMENT_STATUS_DT_IDVALID
 - scsi_ch.h, 746
- READ_ELEMENT_STATUS_DT_LUNMASK
 - scsi_ch.h, 746
- READ_ELEMENT_STATUS_DT_LUVALID
 - scsi_ch.h, 746
- READ_ELEMENT_STATUS_DT_MASK1
 - scsi_ch.h, 746
- READ_ELEMENT_STATUS_DT_NOTBUS
 - scsi_ch.h, 746
- READ_ELEMENT_STATUS_EXCEPT
 - scsi_ch.h, 746
- READ_ELEMENT_STATUS_EXENAB
 - scsi_ch.h, 746
- READ_ELEMENT_STATUS_FULL
 - scsi_ch.h, 747
- read_element_status_header, 192
 - count, 192
 - fear, 192
 - nbytes, 192
 - reserved, 192
- READ_ELEMENT_STATUS_IE_MASK1
 - scsi_ch.h, 747
- READ_ELEMENT_STATUS_IMPEXP
 - scsi_ch.h, 747
- READ_ELEMENT_STATUS_INENAB
 - scsi_ch.h, 747
- READ_ELEMENT_STATUS_INVERT
 - scsi_ch.h, 747
- READ_ELEMENT_STATUS_MT_MASK1
 - scsi_ch.h, 747
- read_element_status_page_header, 193
 - edl, 193
 - flags, 193
 - nbytes, 193
 - reserved, 193
 - type, 193
- READ_ELEMENT_STATUS_PVOLTAG
 - scsi_ch.h, 747
- READ_ELEMENT_STATUS_ST_MASK1
 - scsi_ch.h, 747
- READ_ELEMENT_STATUS_SVALID
 - scsi_ch.h, 747
- READ_ELEMENT_STATUS_VOLTAG
 - scsi_ch.h, 747

- READ_FORMAT_CAPACITIES
 - scsi_da.h, 778
- READ_HEADER
 - scsi_cd.h, 711
- READ_POSITION
 - scsi_sa.h, 939
- read_retry_count
 - scsi_da_rw_recovery_page, 213
- read_select
 - targ_softc, 418
- READ_SUBCHANNEL
 - scsi_cd.h, 711
- READ_TOC
 - scsi_cd.h, 711
- readspeed
 - scsi_set_speed, 376
- REASSIGN_BLOCKS
 - scsi_da.h, 778
- RECEIVE
 - scsi_pt.h, 901
- RECEIVE_DIAGNOSTIC
 - scsi_all.h, 619
- recovery_time_limit
 - scsi_da_rw_recovery_page, 214
- refcount
 - cam_periph, 35
- region_code
 - scsi_send_key_data_rpc, 366
- region_info
 - scsi_read_dvd_struct_data_copyright, 303
- region_mask
 - scsi_report_key_data_rpc, 343
- reladr
 - scsi_read_capacity_16, 290
- RELEASE
 - scsi_all.h, 619
- release_flags
 - ccb_relsim, 95
- release_timeout
 - ccb_relsim, 95
- RELEASE_UNIT
 - scsi_sa.h, 939
- RELSIM_ADJUST_OPENINGS
 - cam_ccb.h, 453
- RELSIM_RELEASE_AFTER_CMDCMPLT
 - cam_ccb.h, 453
- RELSIM_RELEASE_AFTER_QEMPTY
 - cam_ccb.h, 453
- RELSIM_RELEASE_AFTER_TIMEOUT
 - cam_ccb.h, 453
- REPORT_KEY
 - scsi_cd.h, 711
- REPORT_LUNS
 - scsi_all.h, 619
- req_map
 - ccb_eng_exec, 66
 - ccb_scsiio, 100
- request_ccb
 - xpt_scan_bus_info, 426
- REQUEST_SENSE
 - scsi_all.h, 619
- request_sense_size
 - scsi_targ_bh.c, 1002
- REQUEST_VOLUME_ELEMENT_ADDRESS
 - scsi_ch.h, 747
- REQUEST_VOLUME_ELEMENT_ADDRESS_VOLTAG
 - scsi_ch.h, 748
- RESERVE
 - scsi_all.h, 620
- RESERVE_UNIT
 - scsi_sa.h, 939
- reserved
 - ccb_getdev, 70
 - format_defect_list_header, 161
 - page_element_address_assignment, 178
 - read_dvd_struct_data_list, 187
 - read_dvd_struct_write_prot, 189
 - read_element_status_header, 192
 - read_element_status_page_header, 193
 - scsi_control_page, 211
 - scsi_da_rw_recovery_page, 214
 - scsi_data_compression_page, 216
 - scsi_erase, 222
 - scsi_initialize_element_status, 227
 - scsi_inquiry, 228
 - scsi_inquiry_data, 231
 - scsi_load_unload, 235
 - scsi_log_header, 236
 - scsi_log_select, 238
 - scsi_log_sense, 239
 - scsi_mode_blk_desc, 258
 - scsi_mode_block_descr, 259
 - scsi_mode_hdr_10, 260
 - scsi_move_medium, 272
 - scsi_position_to_element, 283
 - scsi_read_defect_data_10, 297
 - scsi_read_defect_data_12, 298
 - scsi_read_defect_data_hdr_10, 299
 - scsi_read_defect_data_hdr_12, 300
 - scsi_read_dvd_struct_data_bca, 301
 - scsi_read_dvd_struct_data_dcb, 304
 - scsi_read_dvd_struct_data_dds, 305
 - scsi_read_dvd_struct_data_disc_id, 306
 - scsi_read_dvd_struct_data_disc_key, 308
 - scsi_read_dvd_struct_data_disc_key_blk, 309
 - scsi_read_dvd_struct_data_header, 311
 - scsi_read_dvd_struct_data_manufacturer, 317

- scsi_read_dvd_struct_data_physical, 319
- scsi_read_dvd_struct_data_prot_discid, 320
- scsi_read_dvd_struct_data_rmd, 321
- scsi_read_dvd_struct_data_rmd_borderout, 322
- scsi_read_dvd_struct_data_spare_area, 323
- scsi_read_dvd_structure, 325
- scsi_reassign_blocks_data, 334
- scsi_report_key_data_agid, 338
- scsi_report_key_data_asf, 339
- scsi_report_key_data_header, 341
- scsi_report_luns_data, 348
- scsi_reserve_release_unit, 353
- scsi_rewind, 354
- scsi_rezero_unit, 355
- scsi_rw_10, 356
- scsi_rw_12, 358
- scsi_rw_16, 360
- scsi_send_key, 365
- scsi_set_speed, 376
- scsi_start_stop_unit, 378
- scsi_status_iu_header, 380
- scsi_sync_cache, 381
- scsi_tape_position_data, 384
- scsi_tape_read_position, 386
- scsi_vpd_unit_serial_number, 389
- volume_tag, 424
- reserved0
 - page_device_capabilities, 175
 - read_dvd_struct_write_prot, 189
 - read_element_status_descriptor, 191
 - scsi_read_dvd_struct_data_copy_manage, 302
 - scsi_read_dvd_struct_data_copyright, 303
 - scsi_read_dvd_struct_data_lead_in, 315
 - scsi_read_dvd_struct_data_medium_status, 318
 - scsi_read_element_status, 326
 - scsi_read_format_capacities, 328
 - scsi_report_key, 336
 - scsi_report_key_data_challenge, 340
 - scsi_report_key_data_key1_key2, 342
 - scsi_report_key_data_rpc, 343
 - scsi_report_key_data_title, 345
 - scsi_request_volume_element_address, 350
 - scsi_send_key_data_rpc, 366
 - scsi_verify, 388
 - scsi_write_and_verify, 390
- reserved1
 - page_device_capabilities, 175
 - read_element_status_descriptor, 191
 - scsi_read_dvd_struct_data_copy_manage, 302
 - scsi_read_dvd_struct_data_copyright, 303
 - scsi_read_dvd_struct_data_lead_in, 315
- scsi_read_dvd_struct_data_medium_status, 318
- scsi_read_element_status, 326
- scsi_read_format_capacities, 328
- scsi_report_key, 336
- scsi_report_key_data_challenge, 340
- scsi_report_key_data_key1_key2, 342
- scsi_report_key_data_rpc, 343
- scsi_report_key_data_title, 345
- scsi_request_volume_element_address, 351
- scsi_send_key_data_rpc, 366
- scsi_send_volume_tag_parameters, 370
- scsi_tape_locate, 382
- scsi_verify, 388
- scsi_write_and_verify, 390
- reserved19
 - disk_pages::rigid_geometry_page, 154
- reserved2
 - scsi_da_rw_recovery_page, 214
 - scsi_inquiry_data, 231
 - scsi_read_dvd_struct_data_lead_in, 316
 - scsi_report_luns, 346
 - scsi_send_volume_tag, 368
 - scsi_send_volume_tag_parameters, 370
 - scsi_tape_locate, 382
 - scsi_tape_position_data, 385
- reserved21
 - disk_pages::format_device_page, 152
- reserved22
 - disk_pages::format_device_page, 152
 - disk_pages::rigid_geometry_page, 154
- reserved23
 - disk_pages::format_device_page, 152
 - disk_pages::rigid_geometry_page, 155
- reserved3
 - scsi_inquiry_data, 231
 - scsi_read_dvd_struct_data_lead_in, 316
 - scsi_report_luns, 346
- reserved30
 - disk_pages::flexible_disk_page, 148
- reserved31
 - disk_pages::flexible_disk_page, 148
- reserved4
 - scsi_read_dvd_struct_data_lead_in, 316
 - scsi_send_volume_tag, 368
- reserved5
 - scsi_read_dvd_struct_data_lead_in, 316
 - scsi_send_volume_tag, 369
- resid
 - ccb_scsiio, 101
- response_format
 - scsi_inquiry_data, 231
- result

- dev_match_result, 140
- resume
 - scsi_pause, 275
- retry_count
 - ccb_hdr, 78
- revision
 - scsi_inquiry_data, 231
 - scsi_inquiry_pattern, 234
 - scsi_static_inquiry_pattern, 379
- REWIND
 - scsi_sa.h, 939
- REWIND_TIMEOUT
 - scsi_sa.c, 908
- REZERO_UNIT
 - scsi_da.h, 778
- RF_KF_ASF
 - scsi_cd.h, 712
- RF_KF_INV_AGID
 - scsi_cd.h, 712
- RF_KF_KEY1
 - scsi_cd.h, 712
- RF_KF_RPC_REPORT
 - scsi_cd.h, 712
- RF_KF_TITLE
 - scsi_cd.h, 712
- RIGHT_CHANNEL
 - scsi_cd.h, 712
- RIGHT_PORT
 - scsi_cd.h, 712
- rigid_geometry
 - disk_pages, 145
- RK_KF_AGID
 - scsi_cd.h, 712
- RK_KF_AGID_MASK
 - scsi_cd.h, 712
- RK_KF_AGID_SHIFT
 - scsi_cd.h, 712
- RK_KF_CHALLENGE
 - scsi_cd.h, 713
- RK_KF_KEY2
 - scsi_cd.h, 713
- RK_KF_KEYFORMAT_MASK
 - scsi_cd.h, 713
- RK_KF_RPC_SET
 - scsi_cd.h, 713
- RKD_AGID_MASK
 - scsi_cd.h, 713
- RKD_AGID_SHIFT
 - scsi_cd.h, 713
- RKD_ASF_SUCCESS
 - scsi_cd.h, 713
- RKD_RPC_SCHEME_PHASE_II
 - scsi_cd.h, 713
- RKD_RPC_SCHEME_UNKNOWN
 - scsi_cd.h, 713
- RKD_RPC_TYPE_LAST_CHANCE
 - scsi_cd.h, 713
- RKD_RPC_TYPE_MASK
 - scsi_cd.h, 714
- RKD_RPC_TYPE_NONE
 - scsi_cd.h, 714
- RKD_RPC_TYPE_PERM
 - scsi_cd.h, 714
- RKD_RPC_TYPE_SET
 - scsi_cd.h, 714
- RKD_RPC_TYPE_SHIFT
 - scsi_cd.h, 714
- RKD_RPC_USER_RESET_MASK
 - scsi_cd.h, 714
- RKD_RPC_USER_RESET_SHIFT
 - scsi_cd.h, 714
- RKD_RPC_VENDOR_RESET_MASK
 - scsi_cd.h, 714
- RKD_RPC_VENDOR_RESET_SHIFT
 - scsi_cd.h, 714
- RKD_TITLE_CMGS_1_GEN
 - scsi_cd.h, 714
- RKD_TITLE_CMGS_MASK
 - scsi_cd.h, 715
- RKD_TITLE_CMGS_NO_COPY
 - scsi_cd.h, 715
- RKD_TITLE_CMGS_NO_RST
 - scsi_cd.h, 715
- RKD_TITLE_CMGS_RSVD
 - scsi_cd.h, 715
- RKD_TITLE_CMGS_SHIFT
 - scsi_cd.h, 715
- RKD_TITLE_CP_SEC
 - scsi_cd.h, 715
- RKD_TITLE_CP_SEC_SHIFT
 - scsi_cd.h, 715
- RKD_TITLE_CPM
 - scsi_cd.h, 715
- RKD_TITLE_CPM_SHIFT
 - scsi_cd.h, 715
- rlec
 - scsi_control_page, 211
- rmd
 - scsi_read_dvd_struct_data_rmd_borderout, 322
- rmd_bytes
 - scsi_read_dvd_struct_data_rmd, 321
- rot_offset
 - disk_pages::rigid_geometry_page, 155
- rpc_scheme0
 - scsi_report_key_data_rpc, 343
- rpc_scheme1
 - scsi_report_key_data_rpc, 343

- rpl
 - disk_pages::rigid_geometry_page, 155
- RPL_LUNDATA_ATYP_EXTLUN
 - scsi_all.h, 620
- RPL_LUNDATA_ATYP_FLAT
 - scsi_all.h, 620
- RPL_LUNDATA_ATYP_LUN
 - scsi_all.h, 620
- RPL_LUNDATA_ATYP_MASK
 - scsi_all.h, 620
- RPL_LUNDATA_ATYP_PERIPH
 - scsi_all.h, 620
- RPL_LUNDATA_EXT_EAM_MASK
 - scsi_all.h, 620
- RPL_LUNDATA_EXT_EAM_NOT_SPEC
 - scsi_all.h, 620
- RPL_LUNDATA_EXT_EAM_WK
 - scsi_all.h, 620
- RPL_LUNDATA_EXT_LEN_MASK
 - scsi_all.h, 620
- RPL_LUNDATA_FLAT_LUN_MASK
 - scsi_all.h, 620
- RPL_LUNDATA_LUN_BUS_MASK
 - scsi_all.h, 621
- RPL_LUNDATA_LUN_LUN_MASK
 - scsi_all.h, 621
- RPL_LUNDATA_LUN_TARG_MASK
 - scsi_all.h, 621
- RPL_LUNDATA_PERIPH_BUS_MASK
 - scsi_all.h, 621
- RPL_REPORT_ALL
 - scsi_all.h, 621
- RPL_REPORT_DEFAULT
 - scsi_all.h, 621
- RPL_REPORT_WELLKNOWN
 - scsi_all.h, 621
- RWB_MODE
 - scsi_all.h, 621
- RWB_MODE_DATA
 - scsi_all.h, 621
- RWB_MODE_DOWNLOAD
 - scsi_all.h, 621
- RWB_MODE_DOWNLOAD_SAVE
 - scsi_all.h, 621
- RWB_MODE_HDR_DATA
 - scsi_all.h, 622
- rwp
 - scsi_read_dvd_struct_data_leadin, 316
- rwp_wavelength
 - scsi_read_dvd_struct_data_leadin, 316
- S
 - scsi_all.c, 592
- SA_ASOC_WP
 - scsi_sa.h, 939
- SA_ATYPE_ER
 - scsi_sa.c, 908
- SA_ATYPE_NR
 - scsi_sa.c, 908
- SA_ATYPE_R
 - scsi_sa.c, 908
- SA_AVC
 - scsi_sa.h, 939
- SA_BIS
 - scsi_sa.h, 939
- SA_CCB_BUFFER_IO
 - scsi_sa.c, 909
- SA_CCB_TYPEMASK
 - scsi_sa.c, 909
- SA_CCB_WAITING
 - scsi_sa.c, 909
- sa_cdevsw
 - scsi_sa.c, 933
- SA_COMP_DEFAULT
 - scsi_sa.h, 939
- SA_COMP_NONE
 - scsi_sa.h, 939
- sa_comp_t, 194
 - dcomp, 194
 - dconf, 194
 - hdr, 194
 - pagecode, 194
 - pagelength, 194
- SA_CTLDEV
 - scsi_sa.c, 909
- SA_DATA_COMPRESSION_PAGE
 - scsi_sa.h, 939
- SA_DBR
 - scsi_sa.h, 940
- SA_DCP_DCC
 - scsi_sa.h, 940
- SA_DCP_DCE
 - scsi_sa.h, 940
- SA_DCP_DDE
 - scsi_sa.h, 940
- SA_DCP_RED_0
 - scsi_sa.h, 940
- SA_DCP_RED_1
 - scsi_sa.h, 940
- SA_DCP_RED_2
 - scsi_sa.h, 940
- SA_DCP_RED_MASK
 - scsi_sa.h, 940
- SA_DCP_RED_SHAMT
 - scsi_sa.h, 940
- SA_DEVICE_CONFIGURATION_PAGE
 - scsi_sa.h, 940
- sa_devs, 196

- ctl_dev, [196](#)
- mode_devs, [196](#)
- sa_devs::sa_mode_devs, [197](#)
 - er_dev, [197](#)
 - nr_dev, [197](#)
 - r_dev, [197](#)
- SA_ERASE_TIMEOUT
 - scsi_sa.c, [909](#)
- SA_FLAG_COMP_ENABLED
 - scsi_sa.c, [911](#)
- SA_FLAG_COMP_SUPP
 - scsi_sa.c, [911](#)
- SA_FLAG_COMP_UNSUPP
 - scsi_sa.c, [911](#)
- SA_FLAG_EIO_PENDING
 - scsi_sa.c, [911](#)
- SA_FLAG_EOF_PENDING
 - scsi_sa.c, [911](#)
- SA_FLAG_EOM_PENDING
 - scsi_sa.c, [911](#)
- SA_FLAG_ERR_PENDING
 - scsi_sa.c, [911](#)
- SA_FLAG_FIXED
 - scsi_sa.c, [911](#)
- SA_FLAG_INVALID
 - scsi_sa.c, [911](#)
- SA_FLAG_OPEN
 - scsi_sa.c, [911](#)
- SA_FLAG_TAPE_FROZEN
 - scsi_sa.c, [911](#)
- SA_FLAG_TAPE_LOCKED
 - scsi_sa.c, [911](#)
- SA_FLAG_TAPE_MOUNTED
 - scsi_sa.c, [911](#)
- SA_FLAG_TAPE_WP
 - scsi_sa.c, [911](#)
- SA_FLAG_TAPE_WRITTEN
 - scsi_sa.c, [911](#)
- sa_flags
 - scsi_sa.c, [911](#)
- SA_IO_TIMEOUT
 - scsi_sa.c, [909](#)
- SA_IS_CTRL
 - scsi_sa.c, [909](#)
- SA_MEDIUM_PARTITION_PAGE_1
 - scsi_sa.h, [941](#)
- SA_MEDIUM_PARTITION_PAGE_2
 - scsi_sa.h, [941](#)
- SA_MEDIUM_PARTITION_PAGE_3
 - scsi_sa.h, [941](#)
- SA_MEDIUM_PARTITION_PAGE_4
 - scsi_sa.h, [941](#)
- sa_mode
 - scsi_sa.c, [911](#)
- SA_MODE_NOREWIND
 - scsi_sa.c, [912](#)
- SA_MODE_OFFLINE
 - scsi_sa.c, [912](#)
- SA_MODE_REWIND
 - scsi_sa.c, [912](#)
- SA_NOT_CTLDEV
 - scsi_sa.c, [909](#)
- SA_NUM_MODES
 - scsi_sa.c, [909](#)
- SA_PARAM_ALL
 - scsi_sa.c, [912](#)
- SA_PARAM_BLOCKSIZE
 - scsi_sa.c, [912](#)
- SA_PARAM_BUFF_MODE
 - scsi_sa.c, [912](#)
- SA_PARAM_COMPRESSION
 - scsi_sa.c, [912](#)
- SA_PARAM_DENSITY
 - scsi_sa.c, [912](#)
- SA_PARAM_NONE
 - scsi_sa.c, [912](#)
- SA_PARAM_NUMBLOCKS
 - scsi_sa.c, [912](#)
- SA_PARAM_SPEED
 - scsi_sa.c, [912](#)
- SA_PARAM_WP
 - scsi_sa.c, [912](#)
- sa_params
 - scsi_sa.c, [912](#)
- SA_PERM_WP
 - scsi_sa.h, [941](#)
- SA_PERS_WP
 - scsi_sa.h, [941](#)
- SA_POSITION_UPDATED
 - scsi_sa.c, [910](#)
- SA_QUIRK_1FM
 - scsi_sa.c, [912](#)
- SA_QUIRK_2FM
 - scsi_sa.c, [912](#)
- sa_quirk_entry, [198](#)
 - inq_pat, [198](#)
 - prefblk, [198](#)
 - quirks, [198](#)
- SA_QUIRK_FIXED
 - scsi_sa.c, [912](#)
- SA_QUIRK_NO_CPAGE
 - scsi_sa.c, [912](#)
- SA_QUIRK_NO_MODESEL
 - scsi_sa.c, [912](#)
- SA_QUIRK_NOCOMP
 - scsi_sa.c, [912](#)
- SA_QUIRK_NODREAD
 - scsi_sa.c, [912](#)

- SA_QUIRK_NONE
 - scsi_sa.c, 912
- sa_quirk_table
 - scsi_sa.c, 933
- SA_QUIRK_VARIABLE
 - scsi_sa.c, 912
- sa_quirks
 - scsi_sa.c, 912
- SA_RBO
 - scsi_sa.h, 941
- SA_READ
 - scsi_sa.h, 941
- SA_REW
 - scsi_sa.h, 941
- SA_REWIND_TIMEOUT
 - scsi_sa.c, 910
- SA_RPOS_BCU
 - scsi_sa.h, 941
- SA_RPOS_BOP
 - scsi_sa.h, 941
- SA_RPOS_BPU
 - scsi_sa.h, 942
- SA_RPOS_BYCU
 - scsi_sa.h, 942
- SA_RPOS_EOP
 - scsi_sa.h, 942
- SA_RPOS_PERR
 - scsi_sa.h, 942
- SA_RPOS_UNCERTAIN
 - scsi_sa.h, 942
- SA_RSMK
 - scsi_sa.h, 942
- SA_SOCF_MASK
 - scsi_sa.h, 942
- sa_softc, 199
 - __pad0__, 200
 - _last_ctl_cdb, 200
 - _last_ctl_resid, 200
 - _last_ctl_sense, 200
 - _last_io_cdb, 200
 - _last_io_resid, 200
 - _last_io_sense, 200
 - bio_queue, 200
 - blk_gran, 200
 - blk_mask, 201
 - blk_shift, 201
 - blkno, 201
 - buffer_mode, 201
 - comp_algorithm, 201
 - ctrl_mode, 201
 - device_stats, 201
 - devs, 201
 - dsreg, 202
 - errinfo, 202
 - filemarks, 202
 - fileno, 202
 - flags, 202
 - last_media_blksize, 202
 - last_resid_was_io, 202
 - max_blk, 202
 - media_blksize, 203
 - media_density, 203
 - media_numblks, 203
 - min_blk, 203
 - open_pending_mount, 203
 - open_ronly, 203
 - queue_count, 203
 - quirks, 203
 - saved_ccb, 203
 - saved_comp_algorithm, 204
 - scsi_rev, 204
 - speed, 204
 - state, 204
- SA_SPACE_TIMEOUT
 - scsi_sa.c, 910
- SA_SPOS_BT
 - scsi_sa.h, 942
- SA_SPOS_CP
 - scsi_sa.h, 942
- SA_SPOS_IMMED
 - scsi_sa.h, 942
- sa_state
 - scsi_sa.c, 912
- SA_STATE_ABNORMAL
 - scsi_sa.c, 912
- SA_STATE_NORMAL
 - scsi_sa.c, 912
- SA_WRITE
 - scsi_sa.h, 942
- saasync
 - scsi_sa.c, 913
- sac
 - scsi_send_volume_tag, 369
- sachecked
 - scsi_sa.c, 913
- sacleanup
 - scsi_sa.c, 914, 933
- saclose
 - scsi_sa.c, 914, 933
- SADENSITY
 - scsi_sa.c, 910
- sadone
 - scsi_sa.c, 915
- sadriver
 - scsi_sa.c, 933
- saerase
 - scsi_sa.c, 915
- saerror

- scsi_sa.c, 916
- SAFT_ALARM_OFFSET
 - scsi_ses.c, 959
- SAFT_BAIL
 - scsi_ses.c, 959
- SAFT_FLG1_ALARM
 - scsi_ses.c, 959
- SAFT_FLG1_ENCDRVFAIL
 - scsi_ses.c, 959
- SAFT_FLG1_ENCDRVWARN
 - scsi_ses.c, 959
- SAFT_FLG1_ENCFANFAIL
 - scsi_ses.c, 959
- SAFT_FLG1_ENCPWRFAIL
 - scsi_ses.c, 959
- SAFT_FLG1_ENCPWROFF
 - scsi_ses.c, 959
- SAFT_FLG1_GLOBFAIL
 - scsi_ses.c, 960
- SAFT_FLG1_GLOBWARN
 - scsi_ses.c, 960
- SAFT_FLG2_LOCKDOOR
 - scsi_ses.c, 960
- SAFT_PRIVATE
 - scsi_ses.c, 960
- SAFT_SCRATCH
 - scsi_ses.c, 960
- saft_2little
 - scsi_ses.c, 983
- SAFTE_END
 - scsi_ses.c, 960
- saft_get_encstat
 - scsi_ses.c, 967
- saft_get_objstat
 - scsi_ses.c, 967
- saft_getconfig
 - scsi_ses.c, 968
- saft_init_enc
 - scsi_ses.c, 968
- SAFTE_LEN
 - scsi_ses.c, 960
- SAFTE_RD_RDCFG
 - scsi_ses.c, 960
- SAFTE_RD_RDDSTS
 - scsi_ses.c, 960
- SAFTE_RD_RDESTS
 - scsi_ses.c, 961
- saft_rdstat
 - scsi_ses.c, 969
- saft_set_encstat
 - scsi_ses.c, 969
- saft_set_objstat
 - scsi_ses.c, 970
- saft_softc_init
 - scsi_ses.c, 970
- SAFTE_START
 - scsi_ses.c, 961
- SAFTE_WT_ACTPWS
 - scsi_ses.c, 961
- SAFTE_WT_DSTAT
 - scsi_ses.c, 961
- SAFTE_WT_FANSPD
 - scsi_ses.c, 961
- SAFTE_WT_GLOBAL
 - scsi_ses.c, 961
- SAFTE_WT_SLTOP
 - scsi_ses.c, 961
- sagetparams
 - scsi_sa.c, 917
- sainit
 - scsi_sa.c, 918, 933
- saioctl
 - scsi_sa.c, 918, 933
- saloadunload
 - scsi_sa.c, 920
- samarkswanted
 - scsi_sa.c, 920
- SAMINOR
 - scsi_sa.c, 910
- SAMODE
 - scsi_sa.c, 910
- samount
 - scsi_sa.c, 920
- samsung
 - cam_xpt.c, 568
- saoninvalidate
 - scsi_sa.c, 921, 934
- saopen
 - scsi_sa.c, 922, 934
- saprevent
 - scsi_sa.c, 922
- SAR_SLI
 - scsi_sa.h, 943
- sardpos
 - scsi_sa.c, 923
- saregister
 - scsi_sa.c, 923, 934
- sareservereleaseunit
 - scsi_sa.c, 924
- saretension
 - scsi_sa.c, 924
- sarewind
 - scsi_sa.c, 925
- SARW_FIXED
 - scsi_sa.h, 943
- sas
 - ccb_pathinq, 85
 - ccb_trans_settings, 109

- sasetparams
 - scsi_sa.c, 925
- sasetpos
 - scsi_sa.c, 926
- saspace
 - scsi_sa.c, 927
- sastart
 - scsi_sa.c, 928, 934
- sastrategy
 - scsi_sa.c, 928, 934
- SAUNIT
 - scsi_sa.c, 910
- saved_ccb
 - ch_softc, 135
 - pass_softc, 180
 - sa_softc, 203
- saved_ccb_ptr
 - cam_periph.c, 468
- saved_comp_algorithm
 - sa_softc, 204
- sawritefilemarks
 - scsi_sa.c, 929
- sbit
 - scsi_ses.c, 961
- sbyte
 - scsi_ses.c, 961
- sc_counts
 - ch_softc, 135
- sc_exchangemask
 - ch_softc, 135
- sc_firsts
 - ch_softc, 135
- SC_LOW_ATTEN_T
 - scsi_low.h, 843
- SC_LOW_BUSRST_T
 - scsi_low.h, 843
- SC_LOW_INIT_T
 - scsi_low.h, 843
- SC_LOW_LUN_INIT_T
 - scsi_low.h, 843
- SC_LOW_MSG_T
 - scsi_low.h, 843
- SC_LOW_NEXUS_T
 - scsi_low.h, 844
- SC_LOW_POLL_T
 - scsi_low.h, 844
- SC_LOW_POWER_T
 - scsi_low.h, 844
- SC_LOW_SELECT_T
 - scsi_low.h, 844
- sc_low_t
 - scsi_low.h, 857
- SC_LOW_TARG_INIT_T
 - scsi_low.h, 844
- SC_LOW_TIMEOUT_T
 - scsi_low.h, 844
- sc_movemask
 - ch_softc, 135
- sc_p, 205
 - scp_cmd, 205
 - scp_cmdlen, 205
 - scp_data, 205
 - scp_dataalen, 205
 - scp_direction, 205
 - scp_spare, 206
 - scp_status, 206
- sc_picker
 - ch_softc, 135
- SC_SCSI_1
 - scsi_all.h, 622
- SC_SCSI_2
 - scsi_all.h, 622
- sc_settle delay
 - ch_softc, 135
- SCB_RLEC
 - scsi_all.h, 622
- scfg, 207
 - DoorLock, 207
 - flag1, 207
 - flag2, 207
 - Nalarm, 207
 - Nfans, 207
 - Npwr, 207
 - Nslots, 208
 - Nspkrs, 208
 - Ntherm, 208
 - pwroff, 208
 - slotoff, 208
- scp_cmd
 - sc_p, 205
- scp_cmdlen
 - sc_p, 205
- scp_data
 - sc_p, 205
- scp_dataalen
 - sc_p, 205
- scp_direction
 - sc_p, 205
- SCP_EAENP
 - scsi_all.h, 622
- SCP_EECA
 - scsi_all.h, 622
- SCP_QUEUE_ALG_MASK
 - scsi_all.h, 622
- SCP_QUEUE_ALG_RESTRICTED
 - scsi_all.h, 622
- SCP_QUEUE_ALG_UNRESTRICTED
 - scsi_all.h, 622

- SCP_QUEUE_DQUE
 - scsi_all.h, 622
- SCP_QUEUE_ERR
 - scsi_all.h, 622
- SCP_RAENP
 - scsi_all.h, 623
- scp_spare
 - sc_p, 206
- scp_status
 - sc_p, 206
- SCP_UAAENP
 - scsi_all.h, 623
- scsi
 - ccb_trans_settings, 109
- SCSI2_RESET_DELAY
 - scsi_low.h, 844
- scsi_2btoul
 - scsi_all.h, 645
- scsi_3btol
 - scsi_all.h, 645
- scsi_3btoul
 - scsi_all.h, 645
- scsi_4btoul
 - scsi_all.h, 646
- scsi_8btou64
 - scsi_all.h, 646
- scsi_all.c
 - __FBSDDID, 592
 - A, 591
 - ALL, 591
 - asc_table, 604
 - asc_table_size, 604
 - ascentrycomp, 592
 - C, 591
 - D, 591
 - E, 591
 - fetchtableentries, 592
 - init_scsi_delay, 593
 - L, 591
 - M, 591
 - O, 591
 - P, 591
 - period, 604
 - period_factor, 604
 - plexor_cd_ops, 604
 - quantum_fireball_entries, 604
 - R, 592
 - S, 592
 - scsi_calc_syncparam, 593
 - scsi_calc_synchrates, 593
 - scsi_cdb_string, 593
 - scsi_command_string, 593
 - SCSI_DELAY, 592
 - scsi_delay, 605
 - scsi_error_action, 594
 - scsi_inquiry, 594
 - scsi_inquiry_match, 595
 - scsi_log_select, 595
 - scsi_log_sense, 595
 - SCSI_MIN_DELAY, 592
 - scsi_mode_select, 596
 - scsi_mode_select_len, 596
 - scsi_mode_sense, 596
 - scsi_mode_sense_len, 597
 - scsi_op_codes, 605
 - scsi_op_desc, 597
 - scsi_op_quirk_table, 605
 - scsi_prevent, 597
 - scsi_print_inquiry, 598
 - scsi_read_capacity, 598
 - scsi_read_capacity_16, 598
 - scsi_read_write, 599
 - scsi_report_luns, 599
 - scsi_request_sense, 600
 - scsi_sense_desc, 600
 - scsi_sense_print, 600
 - scsi_sense_sbuf, 601
 - scsi_sense_string, 601
 - scsi_start_stop, 602
 - scsi_static_inquiry_match, 602
 - scsi_status_string, 602
 - scsi_synchronize_cache, 603
 - scsi_synchrates, 605
 - scsi_test_unit_ready, 603
 - sense_key_table, 605
 - sense_key_table_size, 606
 - sense_quirk_table, 606
 - sense_quirk_table_size, 606
 - senseentrycomp, 603
 - set_scsi_delay, 603
 - sony_mo_entries, 606
 - SST, 592
 - SYSCTL_PROC, 604
 - sysctl_scsi_delay, 604
 - SYSINIT, 604
 - T, 592
 - W, 592
- scsi_all.h
 - SS_FAIL, 644
 - SS_MASK, 644
 - SS_NOP, 644
 - SS_REQSENSE, 644
 - SS_RETRY, 644
 - SS_START, 644
 - SS_TUR, 644
 - SSQ_DECREMENT_COUNT, 644
 - SSQ_MANY, 644
 - SSQ_MASK, 644

- SSQ_NONE, 644
- SSQ_PRINT_SENSE, 644
- SSQ_RANGE, 644
- SSS_FLAG_NONE, 645
- SSS_FLAG_PRINT_COMMAND, 645
- scsi_all.h
 - _SCSI_SCSI_ALL_H, 617
 - CHANGE_DEFINITION, 617
 - find_mode_page_10, 645
 - find_mode_page_6, 645
 - INQ_DATA_TQ_ENABLED, 617
 - INQUIRY, 617
 - LOG_SELECT, 617
 - LOG_SENSE, 617
 - MODE_SELECT_10, 617
 - MODE_SELECT_6, 617
 - MODE_SENSE_10, 617
 - MODE_SENSE_6, 618
 - MOVE_MEDIUM, 618
 - POSITION_TO_ELEMENT, 618
 - PR_ALLOW, 618
 - PR_PREVENT, 618
 - PREVENT_ALLOW, 618
 - READ_10, 618
 - READ_12, 618
 - READ_16, 619
 - READ_6, 619
 - READ_BUFFER, 619
 - READ_CAPACITY, 619
 - READ_ELEMENT_STATUS, 619
 - RECEIVE_DIAGNOSTIC, 619
 - RELEASE, 619
 - REPORT_LUNS, 619
 - REQUEST_SENSE, 619
 - RESERVE, 620
 - RPL_LUNDATA_ATYP_EXTLUN, 620
 - RPL_LUNDATA_ATYP_FLAT, 620
 - RPL_LUNDATA_ATYP_LUN, 620
 - RPL_LUNDATA_ATYP_MASK, 620
 - RPL_LUNDATA_ATYP_PERIPH, 620
 - RPL_LUNDATA_EXT_EAM_MASK, 620
 - RPL_LUNDATA_EXT_EAM_NOT_SPEC, 620
 - RPL_LUNDATA_EXT_EAM_WK, 620
 - RPL_LUNDATA_EXT_LEN_MASK, 620
 - RPL_LUNDATA_FLAT_LUN_MASK, 620
 - RPL_LUNDATA_LUN_BUS_MASK, 621
 - RPL_LUNDATA_LUN_LUN_MASK, 621
 - RPL_LUNDATA_LUN_TARG_MASK, 621
 - RPL_LUNDATA_PERIPH_BUS_MASK, 621
 - RPL_REPORT_ALL, 621
 - RPL_REPORT_DEFAULT, 621
 - RPL_REPORT_WELLKNOWN, 621
 - RWB_MODE, 621
 - RWB_MODE_DATA, 621
 - RWB_MODE_DOWNLOAD, 621
 - RWB_MODE_DOWNLOAD_SAVE, 621
 - RWB_MODE_HDR_DATA, 622
 - SC_SCSI_1, 622
 - SC_SCSI_2, 622
 - SCB_RLEC, 622
 - SCP_EAENP, 622
 - SCP_EECA, 622
 - SCP_QUEUE_ALG_MASK, 622
 - SCP_QUEUE_ALG_RESTRICTED, 622
 - SCP_QUEUE_ALG_UNRESTRICTED, 622
 - SCP_QUEUE_DQUE, 622
 - SCP_QUEUE_ERR, 622
 - SCP_RAENP, 623
 - SCP_UAAENP, 623
 - scsi_2btoul, 645
 - scsi_3btol, 645
 - scsi_3btoul, 645
 - scsi_4btoul, 646
 - scsi_8btoul64, 646
 - scsi_calc_syncparam, 646
 - scsi_calc_syncsrate, 646
 - SCSI_CDB6_LEN, 623
 - scsi_cdb_string, 646
 - SCSI_CMD_LUN, 623
 - SCSI_CMD_LUN_SHIFT, 623
 - scsi_command_string, 646
 - SCSI_CTL_FLAG, 623
 - SCSI_CTL_LINK, 623
 - SCSI_CTL_VENDOR, 623
 - SCSI_DEFAULT_DENSITY, 623
 - scsi_delay, 657
 - scsi_error_action, 647
 - scsi_extract_sense, 647
 - scsi_inquiry, 647
 - scsi_inquiry_match, 648
 - scsi_interpret_sense, 648
 - scsi_log_select, 648
 - scsi_log_sense, 648
 - SCSI_MAX_CDBLEN, 623
 - scsi_mode_select, 649
 - scsi_mode_select_len, 649
 - scsi_mode_sense, 649
 - scsi_mode_sense_len, 650
 - scsi_op_desc, 650
 - scsi_prevent, 650
 - scsi_print_inquiry, 651
 - scsi_read_capacity, 651
 - scsi_read_capacity_16, 651
 - scsi_read_write, 652
 - scsi_report_luns, 652
 - scsi_request_sense, 653
 - SCSI_REV_0, 624

- SCSI_REV_2, 624
- SCSI_REV_CCS, 624
- SCSI_REV_SPC, 624
- SCSI_REV_SPC2, 624
- SCSI_SAME_DENSITY, 624
- scsi_sense_action, 644
- scsi_sense_action_qualifier, 644
- scsi_sense_desc, 653
- scsi_sense_key_text, 657
- scsi_sense_print, 653
- scsi_sense_sbuf, 654
- scsi_sense_string, 654
- scsi_sense_string_flags, 644
- scsi_start_stop, 655
- scsi_static_inquiry_match, 655
- SCSI_STATUS_ACA_ACTIVE, 624
- SCSI_STATUS_BUSY, 624
- SCSI_STATUS_CHECK_COND, 624
- SCSI_STATUS_CMD_TERMINATED, 625
- SCSI_STATUS_COND_MET, 625
- SCSI_STATUS_INTERMED, 625
- SCSI_STATUS_INTERMED_COND_MET, 625
- SCSI_STATUS_OK, 625
- SCSI_STATUS_QUEUE_FULL, 625
- SCSI_STATUS_RESERV_CONFLICT, 625
- scsi_status_string, 655
- SCSI_STATUS_TASK_ABORTED, 625
- scsi_synchronize_cache, 656
- scsi_test_unit_ready, 656
- scsi_u64to8b, 656
- scsi_ulto2b, 656
- scsi_ulto3b, 657
- scsi_ulto4b, 657
- SEND_DIAGNOSTIC, 626
- SERVICE_ACTION_IN, 626
- SF_NO_PRINT, 626
- SF_PRINT_ALWAYS, 626
- SF_QUIET_IR, 626
- SF_RETRY_UA, 626
- SHORT_INQUIRY_LENGTH, 626
- SI_EVPD, 626
- SID_ADDITIONAL_LENGTH, 627
- SID_AENC, 627
- SID_ANSI_REV, 627
- SID_CmdQue, 627
- SID_ECMA, 627
- SID_IS_REMOVABLE, 627
- SID_ISO, 627
- SID_Linked, 627
- SID_PRODUCT_SIZE, 628
- SID_QUAL, 628
- SID_QUAL2, 628
- SID_QUAL_BAD_LU, 628
- SID_QUAL_IS_VENDOR_UNIQUE, 628
- SID_QUAL_LU_CONNECTED, 628
- SID_QUAL_LU_OFFLINE, 628
- SID_QUAL_RSVD, 628
- SID_RelAdr, 628
- SID_REVISION_SIZE, 629
- SID_SftRe, 629
- SID_SPI_CLOCK_DT, 629
- SID_SPI_CLOCK_DT_ST, 629
- SID_SPI_CLOCK_ST, 629
- SID_SPI_IUS, 629
- SID_SPI_MASK, 629
- SID_SPI_QAS, 629
- SID_Sync, 629
- SID_TrmIOP, 629
- SID_TYPE, 630
- SID_VENDOR_SIZE, 630
- SID_VENDOR_SPECIFIC_0_SIZE, 630
- SID_VENDOR_SPECIFIC_1_SIZE, 630
- SID_WBus16, 630
- SID_WBus32, 630
- SIP_MEDIA_FIXED, 630
- SIP_MEDIA_REMOVABLE, 630
- SLP_DS, 630
- SLP_DU, 631
- SLP_ETC, 631
- SLP_LBIN, 631
- SLP_LP, 631
- SLP_TMC_ALWAYS, 631
- SLP_TMC_EQUAL, 631
- SLP_TMC_GREATER, 631
- SLP_TMC_MASK, 631
- SLP_TMC_NOTEQUAL, 631
- SLP_TSD, 631
- SLS_ALL_PAGES_PAGE, 631
- SLS_ERROR_LASTN_PAGE, 632
- SLS_ERROR_NONMEDIUM_PAGE, 632
- SLS_ERROR_READ_PAGE, 632
- SLS_ERROR_READREVERSE_PAGE, 632
- SLS_ERROR_VERIFY_PAGE, 632
- SLS_ERROR_WRITE_PAGE, 632
- SLS_OVERRUN_PAGE, 632
- SLS_PAGE_CODE, 632
- SLS_PAGE_CTRL_CUMUL_DEFAULT, 632
- SLS_PAGE_CTRL_CUMULATIVE, 632
- SLS_PAGE_CTRL_MASK, 632
- SLS_PAGE_CTRL_THRESH_DEFAULT, 633
- SLS_PAGE_CTRL_THRESHOLD, 633
- SLS_PCR, 633
- SLS_PPC, 633
- SLS_SP, 633
- SMS_ALL_PAGES_PAGE, 633
- SMS_CONTROL_MODE_PAGE, 633

- SMS_DBD, 633
- SMS_DISCONNECT_RECONNECT_PAGE, 633
- SMS_PAGE_CODE, 633
- SMS_PAGE_CTRL_CHANGEABLE, 634
- SMS_PAGE_CTRL_CURRENT, 634
- SMS_PAGE_CTRL_DEFAULT, 634
- SMS_PAGE_CTRL_MASK, 634
- SMS_PAGE_CTRL_SAVED, 634
- SMS_PERIPHERAL_DEVICE_PAGE, 634
- SMS_PF, 634
- SMS_SP, 634
- SMS_VENDOR_SPECIFIC_PAGE, 634
- SPC2_SID_BQueue, 634
- SPC2_SID_EncServ, 635
- SPC2_SID_MChngr, 635
- SPC2_SID_MultiP, 635
- SRC16_PMI, 635
- SRC16_RELADR, 635
- SRC16_SERVICE_ACTION, 635
- SRW10_DPO, 635
- SRW10_EBP, 635
- SRW10_FUA, 635
- SRW10_RELADDR, 635
- SRW12_DPO, 636
- SRW12_FUA, 636
- SRW12_RELADDR, 636
- SRW16_DPO, 636
- SRW16_FUA, 636
- SRW16_RELADDR, 636
- SRW_TOPADDR, 636
- SS_ERRMASK, 636
- SS_FATAL, 636
- SS_RDEF, 636
- SS_RET, 636
- SSD_BITPTR_VALID, 637
- SSD_BITPTR_VALUE, 637
- SSD_CURRENT_ERROR, 637
- SSD_DEFERRED_ERROR, 637
- SSD_DOL, 637
- SSD_EOM, 637
- SSD_ERRCODE, 637
- SSD_ERRCODE_VALID, 637
- SSD_FIELDPTR_CMD, 637
- SSD_FILEMARK, 637
- SSD_FULL_SIZE, 638
- SSD_ILI, 638
- SSD_KEY, 638
- SSD_KEY_ABORTED_COMMAND, 638
- SSD_KEY_BLANK_CHECK, 638
- SSD_KEY_COPY_ABORTED, 638
- SSD_KEY_DATA_PROTECT, 638
- SSD_KEY_EQUAL, 638
- SSD_KEY_HARDWARE_ERROR, 639
- SSD_KEY_ILLEGAL_REQUEST, 639
- SSD_KEY_MEDIUM_ERROR, 639
- SSD_KEY_MISCOMPARE, 639
- SSD_KEY_NO_SENSE, 639
- SSD_KEY_NOT_READY, 639
- SSD_KEY_RECOVERED_ERROR, 639
- SSD_KEY_RESERVED, 639
- SSD_KEY_UNIT_ATTENTION, 639
- SSD_KEY_Vendor_Specific, 640
- SSD_KEY_VOLUME_OVERFLOW, 640
- SSD_MIN_SIZE, 640
- SSD_PF, 640
- SSD_SCS_VALID, 640
- SSD_SELFTEST, 640
- SSD_UOL, 640
- SSS_IMMED, 640
- SSS_LOEJ, 640
- SSS_START, 640
- START_STOP, 641
- START_STOP_UNIT, 641
- SVPD_SERIAL_NUM_SIZE, 641
- SVPD_UNIT_SERIAL_NUMBER, 641
- SYNCHRONIZE_CACHE, 641
- T_ANY, 641
- T_ASC0, 641
- T_ASC1, 641
- T_CDROM, 641
- T_CHANGER, 641
- T_COMM, 642
- T_DIRECT, 642
- T_ENCLOSURE, 642
- T_FIXED, 642
- T_NODEVICE, 642
- T_OCRW, 642
- T_OPTICAL, 642
- T_PRINTER, 642
- T_PROCESSOR, 642
- T_RBC, 643
- T_REMOV, 643
- T_SCANNER, 643
- T_SEQUENTIAL, 643
- T_STORARRAY, 643
- T_WORM, 643
- TEST_UNIT_READY, 643
- WRITE_10, 643
- WRITE_12, 643
- WRITE_16, 644
- WRITE_6, 644
- WRITE_BUFFER, 644
- scsi_calc_syncparam
 - scsi_all.c, 593
 - scsi_all.h, 646
- scsi_calc_syncrate
 - scsi_all.c, 593

- scsi_all.h, 646
- scsi_cd.c
 - CD_CCB_BUFFER_IO, 662
 - CD_CCB_PROBE, 662
 - CD_CCB_RETRY_UA, 663
 - CD_CCB_TYPE_MASK, 663
 - CD_CCB_WAITING, 663
 - CD_FLAG_ACTIVE, 663
 - CD_FLAG_CHANGER, 663
 - CD_FLAG_DISC_LOCKED, 663
 - CD_FLAG_DISC_REMOVABLE, 663
 - CD_FLAG_INVALID, 663
 - CD_FLAG_NEW_DISC, 663
 - CD_FLAG_RETRY_UA, 663
 - CD_FLAG_SCHED_ON_COMP, 663
 - CD_FLAG_SCTX_INIT, 663
 - CD_FLAG_TAGGED_QUEUEING, 663
 - CD_FLAG_VALID_MEDIA, 663
 - CD_FLAG_VALID_TOC, 663
 - CD_Q_10_BYTE_ONLY, 663
 - CD_Q_BCD_TRACKS, 663
 - CD_Q_CHANGER, 663
 - CD_Q_NO_CHANGER, 663
 - CD_Q_NO_TOUCH, 663
 - CD_Q_NONE, 663
 - CD_STATE_NORMAL, 664
 - CD_STATE_PROBE, 664
 - CHANGER_MANUAL_CALL, 663
 - CHANGER_NEED_TIMEOUT, 663
 - CHANGER_SHORT_TMOUT_SCHED, 663
 - CHANGER_TIMEOUT_SCHED, 663
- scsi_cd.c
 - __FBSDID, 664
 - ccb_bp, 662
 - ccb_state, 662
 - cd6byteworkaround, 664
 - cd_ccb_state, 662
 - cd_changer_flags, 663
 - cd_flags, 663
 - cd_page_size_table, 692
 - cd_quirk_table, 692
 - cd_quirks, 663
 - cd_state, 663
 - cdasync, 664
 - cdchangerschedule, 665
 - cdcheckmedia, 665
 - cdcleanup, 666, 693
 - cdcclose, 666, 693
 - cdcmdsizesysctl, 667
 - cddone, 667
 - cddriver, 693
 - cderror, 668
 - cdgetccb, 669
 - cdgetmode, 670
 - cdgetpage, 670
 - cdgetpagesize, 671
 - cdinit, 693
 - cdioctl, 671, 693
 - cdoninvalidate, 672, 693
 - cdopen, 673, 693
 - cdpause, 673
 - cdplay, 674
 - cdplaymsf, 675
 - cdplaytracks, 676
 - cdprevent, 677
 - cdreadvdstructure, 678
 - cdreadsubchannel, 678
 - cdreadtoc, 679
 - cdregister, 680, 694
 - cdreportkey, 681
 - cdrunccb, 682
 - cdrunchangerqueue, 682
 - cdschedule, 683
 - cdsendkey, 683
 - cdsetmode, 684
 - cdsetspeed, 684
 - cdshorttimeout, 685
 - cdsize, 686
 - cdstart, 686, 694
 - cdstartunit, 687
 - cdstopunit, 688
 - cdstrategy, 689, 694
 - cdsysctlinit, 690
 - CHANGER_MAX_BUSY_SECONDS, 662
 - changer_max_busy_seconds, 694
 - CHANGER_MIN_BUSY_SECONDS, 662
 - changer_min_busy_seconds, 694
 - LEADOUT, 662
 - num_changers, 694
 - PERIPHDRIVER_DECLARE, 690
 - scsi_read_dvd_structure, 690
 - scsi_report_key, 690
 - scsi_send_key, 691
 - STAILQ_HEAD, 691
 - SYSCTL_INT, 692
 - SYSCTL_NODE, 692
 - TUNABLE_INT, 692
- scsi_cd.h
 - _SCSI SCSI_CD_H, 700
 - AUDIO_PAGE, 700
 - CD_MSF, 700
 - CD_PA_APR_VALID, 700
 - CD_PA_FORMAT_LBA, 700
 - CD_PA_IMMED, 700
 - CD_PA_SOTC, 700
 - CD_PAGE_CODE, 700
 - CD_PAGE_PS, 700
 - CD_RELADDR, 700

- CHANNEL, 700
- CHANNEL_0, 701
- CHANNEL_1, 701
- CHANNEL_2, 701
- CHANNEL_3, 701
- LEFT_CHANNEL, 701
- LEFT_PORT, 701
- PA_PAUSE, 701
- PA_RESUME, 701
- PAUSE, 701
- PLAY_10, 701
- PLAY_12, 702
- PLAY_MSF, 702
- PLAY_TRACK, 702
- PLAY_TRACK_REL, 702
- PLAY_TRACK_REL_BIG, 702
- RDS_FORMAT_BCA, 702
- RDS_FORMAT_CMGS_CPM, 702
- RDS_FORMAT_COPYRIGHT, 702
- RDS_FORMAT_DCB, 702
- RDS_FORMAT_DDS, 702
- RDS_FORMAT_DISC_ID, 703
- RDS_FORMAT_DISC_KEY, 703
- RDS_FORMAT_DISC_KEY_BLOCK, 703
- RDS_FORMAT_DVDRAM_MEDIA_STAT, 703
- RDS_FORMAT_LEADIN, 703
- RDS_FORMAT_MANUFACTURER, 703
- RDS_FORMAT_PHYSICAL, 703
- RDS_FORMAT_PROT_DISCID, 703
- RDS_FORMAT_RMD, 703
- RDS_FORMAT_RMD_BORDEROUT, 703
- RDS_FORMAT_SPARE_AREA, 703
- RDS_FORMAT_STRUCTURE_LIST, 704
- RDS_FORMAT_WRITE_PROT, 704
- RDS_ACTION_FORMAT, 704
- RDS_ACTION_MODIFY_DCB, 704
- RDS_ACTION_READING, 704
- RDS_ACTION_RECORDING, 704
- RDS_BCA, 704
- RDS_BCA_MASK, 704
- RDS_BCA_SHIFT, 704
- RDS_BOOK_TYPE_DVD_PRW, 704
- RDS_BOOK_TYPE_DVD_R, 704
- RDS_BOOK_TYPE_DVD_RAM, 705
- RDS_BOOK_TYPE_DVD_ROM, 705
- RDS_BOOK_TYPE_DVD_RW, 705
- RDS_BOOK_TYPE_MASK, 705
- RDS_BOOK_TYPE_SHIFT, 705
- RDS_BOOK_VERSION_MASK, 705
- RDS_CMGS_COPY_ALLOWED, 705
- RDS_CMGS_MASK, 705
- RDS_CMGS_NO_COPIES, 705
- RDS_CMGS_ONE_COPY, 705
- RDS_CPM_HAS_COPYRIGHT, 706
- RDS_CPM_MASK, 706
- RDS_CPM_NO_COPYRIGHT, 706
- RDS_CPS_DATA_EXISTS, 706
- RDS_CPS_NOT_PRESENT, 706
- RDS_DISC_SIZE_120MM, 706
- RDS_DISC_SIZE_80MM, 706
- RDS_DISC_SIZE_MASK, 706
- RDS_DISC_SIZE_SHIFT, 706
- RDS_DT_NEED_CARTRIDGE, 706
- RDS_DT_NO_CART_NEEDED, 706
- RDS_LAYER_TYPE_MASK, 707
- RDS_LAYER_TYPE_RECORD, 707
- RDS_LAYER_TYPE_RO, 707
- RDS_LAYER_TYPE_RW, 707
- RDS_LIN_DENSITY_0267, 707
- RDS_LIN_DENSITY_0280_0291, 707
- RDS_LIN_DENSITY_0293, 707
- RDS_LIN_DENSITY_0353, 707
- RDS_LIN_DENSITY_0409_0435, 707
- RDS_LIN_DENSITY_MASK, 707
- RDS_LIN_DENSITY_SHIFT, 707
- RDS_MAIN_DATA_START_DVD_RO, 708
- RDS_MAIN_DATA_START_DVD_RW, 708
- RDS_MAX_RATE_0252, 708
- RDS_MAX_RATE_0504, 708
- RDS_MAX_RATE_1008, 708
- RDS_MAX_RATE_MASK, 708
- RDS_MAX_RATE_NOT_SPEC, 708
- RDS_MS_CARTRIDGE, 708
- RDS_MS_CWP, 708
- RDS_MS_MSWI, 708
- RDS_MS_OUT, 708
- RDS_MS_PWP, 709
- RDS_NL_ONE_LAYER, 709
- RDS_NL_TWO_LAYERS, 709
- RDS_NUM_LAYERS_MASK, 709
- RDS_NUM_LAYERS_SHIFT, 709
- RDS_RDS_MASK, 709
- RDS_RDS_NOT_READABLE, 709
- RDS_RDS_READABLE, 709
- RDS_SDS_MASK, 709
- RDS_SDS_NOT_WRITEABLE, 709
- RDS_SDS_WRITEABLE, 709
- RDS_SWI_NO_BARE, 710
- RDS_SWI_UNSPEC, 710
- RDS_TP_OTP, 710
- RDS_TP_PTP, 710
- RDS_TRACK_DENSITY_0615, 710
- RDS_TRACK_DENSITY_074, 710
- RDS_TRACK_DENSITY_080, 710
- RDS_TRACK_DENSITY_MASK, 710

- RDSD_TRACK_PATH_MASK, 710
- RDSD_TRACK_PATH_SHIFT, 710
- RDSD_WPS_CWP, 711
- RDSD_WPS_MSWI, 711
- RDSD_WPS_PWP, 711
- RDSD_WPS_SWPP, 711
- READ_CD_CAPACITY, 711
- READ_DVD_STRUCTURE, 711
- READ_HEADER, 711
- READ_SUBCHANNEL, 711
- READ_TOC, 711
- REPORT_KEY, 711
- RF_KF_ASF, 712
- RF_KF_INV_AGID, 712
- RF_KF_KEY1, 712
- RF_KF_RPC_REPORT, 712
- RF_KF_TITLE, 712
- RIGHT_CHANNEL, 712
- RIGHT_PORT, 712
- RK_KF_AGID, 712
- RK_KF_AGID_MASK, 712
- RK_KF_AGID_SHIFT, 712
- RK_KF_CHALLENGE, 713
- RK_KF_KEY2, 713
- RK_KF_KEYFORMAT_MASK, 713
- RK_KF_RPC_SET, 713
- RKD_AGID_MASK, 713
- RKD_AGID_SHIFT, 713
- RKD_ASF_SUCCESS, 713
- RKD_RPC_SCHEME_PHASE_II, 713
- RKD_RPC_SCHEME_UNKNOWN, 713
- RKD_RPC_TYPE_LAST_CHANCE, 713
- RKD_RPC_TYPE_MASK, 714
- RKD_RPC_TYPE_NONE, 714
- RKD_RPC_TYPE_PERM, 714
- RKD_RPC_TYPE_SET, 714
- RKD_RPC_TYPE_SHIFT, 714
- RKD_RPC_USER_RESET_MASK, 714
- RKD_RPC_USER_RESET_SHIFT, 714
- RKD_RPC_VENDOR_RESET_MASK, 714
- RKD_RPC_VENDOR_RESET_SHIFT, 714
- RKD_TITLE_CMGS_1_GEN, 714
- RKD_TITLE_CMGS_MASK, 715
- RKD_TITLE_CMGS_NO_COPY, 715
- RKD_TITLE_CMGS_NO_RST, 715
- RKD_TITLE_CMGS_RSVD, 715
- RKD_TITLE_CMGS_SHIFT, 715
- RKD_TITLE_CP_SEC, 715
- RKD_TITLE_CP_SEC_SHIFT, 715
- RKD_TITLE_CPM, 715
- RKD_TITLE_CPM_SHIFT, 715
- SCSI_RCB, 715
- scsi_read_dvd_structure, 716
- scsi_report_key, 716
- scsi_send_key, 717
- SEND_KEY, 716
- SET_CD_SPEED, 716
- SRS_SUBQ, 716
- SCSI_CDB6_LEN
 - scsi_all.h, 623
- scsi_cdb_string
 - scsi_all.c, 593
 - scsi_all.h, 646
- scsi_ch.c
 - CH_CCB_PROBE, 721
 - CH_CCB_WAITING, 721
 - CH_FLAG_INVALID, 722
 - CH_FLAG_OPEN, 722
 - CH_Q_NO_DBD, 722
 - CH_Q_NONE, 722
 - CH_STATE_NORMAL, 722
 - CH_STATE_PROBE, 722
- scsi_ch.c
 - __FBSIDID, 722
 - ccb_bp, 721
 - ccb_state, 721
 - ch_ccb_types, 721
 - ch_cdevsw, 738
 - ch_flags, 721
 - ch_quirks, 722
 - ch_state, 722
 - CH_TIMEOUT_EXCHANGE_MEDIUM, 738
 - CH_TIMEOUT_INITIALIZE_ELEMENT_STATUS, 738
 - CH_TIMEOUT_MODE_SENSE, 738
 - CH_TIMEOUT_MOVE_MEDIUM, 738
 - CH_TIMEOUT_POSITION_TO_ELEMENT, 738
 - CH_TIMEOUT_READ_ELEMENT_STATUS, 738
 - CH_TIMEOUT_SEND_VOLTAGE, 738
 - chasync, 722
 - chcleanup, 723, 739
 - chclose, 723, 739
 - chdone, 723
 - chdriver, 739
 - cherror, 724
 - chexchange, 725
 - chgetelemstatus, 726
 - chgetparams, 727
 - chielem, 728
 - chinit, 729, 739
 - chioctl, 730, 739
 - chmove, 730
 - choninvalidate, 731, 739
 - chopen, 732, 739
 - chposition, 732

- chregister, 733, 739
- chsetvtag, 734
- chstart, 734, 739
- CHUNIT, 721
- copy_element_status, 735
- copy_vtag, 735
- PERIPHDRIVER_DECLARE, 735
- PLURAL, 721
- scsi_exchange_medium, 736
- scsi_initialize_element_status, 736
- scsi_move_medium, 736
- scsi_position_to_element, 736
- scsi_read_element_status, 737
- scsi_send_volume_tag, 737
- scsi_ch.h
 - _SCSI_SCSI_CH_H, 743
 - CAN_ROTATE, 743
 - CH_DEVICE_CAP_PAGE, 743
 - CH_ELEMENT_ADDR_ASSIGN_PAGE, 743
 - CH_TRANS_GEOM_PARAMS_PAGE, 743
 - ELEMENT_TYPE_ALL, 743
 - ELEMENT_TYPE_DT, 743
 - ELEMENT_TYPE_IE, 744
 - ELEMENT_TYPE_MASK, 744
 - ELEMENT_TYPE_MT, 744
 - ELEMENT_TYPE_ST, 744
 - EXCHANGE_MEDIUM, 744
 - EXCHANGE_MEDIUM_INV1, 744
 - EXCHANGE_MEDIUM_INV2, 744
 - EXCHANGE_WITH_DT, 744
 - EXCHANGE_WITH_IE, 744
 - EXCHANGE_WITH_MT, 744
 - EXCHANGE_WITH_ST, 745
 - INITIALIZE_ELEMENT_STATUS, 745
 - MOVE_MEDIUM, 745
 - MOVE_MEDIUM_INVERT, 745
 - MOVE_TO_DT, 745
 - MOVE_TO_IE, 745
 - MOVE_TO_MT, 745
 - MOVE_TO_ST, 745
 - PGCODE_MASK, 745
 - PGCODE_PS, 745
 - POSITION_TO_ELEMENT, 745
 - POSITION_TO_ELEMENT_INVERT, 746
 - READ_ELEMENT_STATUS, 746
 - READ_ELEMENT_STATUS_ACCESS, 746
 - READ_ELEMENT_STATUS_AVOLTAG, 746
 - READ_ELEMENT_STATUS_DT_IDVALID, 746
 - READ_ELEMENT_STATUS_DT_-LUNMASK, 746
 - READ_ELEMENT_STATUS_DT_-LUVALID, 746
 - READ_ELEMENT_STATUS_DT_MASK1, 746
 - READ_ELEMENT_STATUS_DT_NOTBUS, 746
 - READ_ELEMENT_STATUS_EXCEPT, 746
 - READ_ELEMENT_STATUS_EXENAB, 746
 - READ_ELEMENT_STATUS_FULL, 747
 - READ_ELEMENT_STATUS_IE_MASK1, 747
 - READ_ELEMENT_STATUS_IMPEXP, 747
 - READ_ELEMENT_STATUS_INENAB, 747
 - READ_ELEMENT_STATUS_INVERT, 747
 - READ_ELEMENT_STATUS_MT_MASK1, 747
 - READ_ELEMENT_STATUS_PVOLTAG, 747
 - READ_ELEMENT_STATUS_ST_MASK1, 747
 - READ_ELEMENT_STATUS_SVALID, 747
 - READ_ELEMENT_STATUS_VOLTAG, 747
 - REQUEST_VOLUME_ELEMENT_ADDRESS, 747
 - REQUEST_VOLUME_ELEMENT_ADDRESS_VOLTAG, 748
 - scsi_exchange_medium, 749
 - scsi_initialize_element_status, 749
 - scsi_move_medium, 749
 - scsi_position_to_element, 750
 - scsi_read_element_status, 750
 - scsi_send_volume_tag, 751
 - SEND_VOLUME_TAG, 748
 - SEND_VOLUME_TAG_ASSERT_-ALTERNATE, 748
 - SEND_VOLUME_TAG_ASSERT_-PRIMARY, 748
 - SEND_VOLUME_TAG_REPLACE_-ALTERNATE, 748
 - SEND_VOLUME_TAG_REPLACE_-PRIMARY, 748
 - SEND_VOLUME_TAG_UNDEFINED_-ALTERNATE, 748
 - SEND_VOLUME_TAG_UNDEFINED_-PRIMARY, 748
 - STOR_DT, 748
 - STOR_IE, 748
 - STOR_MT, 749
 - STOR_ST, 749
 - scsi_changedef, 209
 - byte2, 209
 - control, 209
 - datalen, 209
 - how, 209

- opcode, 209
- unused, 209
- unused1, 209
- SCSI_CMD_LUN
 - scsi_all.h, 623
- SCSI_CMD_LUN_SHIFT
 - scsi_all.h, 623
- scsi_command_string
 - scsi_all.c, 593
 - scsi_all.h, 646
- scsi_control_page, 211
 - aen_holdoff_period, 211
 - eca_and_aen, 211
 - page_code, 211
 - page_length, 211
 - queue_flags, 211
 - reserved, 211
 - rlec, 211
- SCSI_CTL_FLAG
 - scsi_all.h, 623
- SCSI_CTL_LINK
 - scsi_all.h, 623
- SCSI_CTL_VENDOR
 - scsi_all.h, 623
- scsi_da.c
 - DA_CCB_BUFFER_IO, 756
 - DA_CCB_DUMP, 756
 - DA_CCB_PROBE, 756
 - DA_CCB_PROBE2, 756
 - DA_CCB_RETRY_UA, 756
 - DA_CCB_TYPE_MASK, 756
 - DA_CCB_WAITING, 756
 - DA_FLAG_NEED_OTAG, 756
 - DA_FLAG_NEW_PACK, 756
 - DA_FLAG_OPEN, 756
 - DA_FLAG_PACK_INVALID, 756
 - DA_FLAG_PACK_LOCKED, 756
 - DA_FLAG_PACK_REMOVABLE, 756
 - DA_FLAG_RETRY_UA, 756
 - DA_FLAG_SCTX_INIT, 756
 - DA_FLAG_TAGGED_QUEUEING, 756
 - DA_FLAG_WENT_IDLE, 756
 - DA_Q_NO_6_BYTE, 756
 - DA_Q_NO_PREVENT, 756
 - DA_Q_NO_SYNC_CACHE, 756
 - DA_Q_NONE, 756
 - DA_STATE_NORMAL, 757
 - DA_STATE_PROBE, 757
 - DA_STATE_PROBE2, 757
- scsi_da.c
 - __FBSDDID, 757
 - ccb_bp, 755
 - ccb_state, 755
 - cmd6workaround, 757
 - da_ccb_state, 756
 - DA_DEFAULT_RETRY, 755
 - DA_DEFAULT_SEND_ORDERED, 755
 - DA_DEFAULT_TIMEOUT, 755
 - da_default_timeout, 770
 - da_flags, 756
 - DA_ORDEREDTAG_INTERVAL, 755
 - da_quirk_table, 770
 - da_quirks, 756
 - da_retry_count, 770
 - da_send_ordered, 770
 - da_state, 756
 - daasync, 757
 - dacleanup, 758, 770
 - daclose, 758
 - dacmdsizesysctl, 759
 - dadone, 759
 - dadriver, 770
 - dadump, 760, 771
 - daerror, 761
 - dagetcapacity, 762
 - dainit, 763, 771
 - daoninvalidate, 764, 771
 - daprevent, 764
 - daregister, 765, 771
 - dasendorderedtag, 766, 771
 - dasetgeom, 766
 - dashutdown, 766
 - dastart, 767, 771
 - dastrategy, 767, 771
 - dasysctlinit, 768
 - microp, 771
 - PERIPHDRIVER_DECLARE, 768
 - quantum, 772
 - SLIST_HEAD, 768
 - SYSCTL_INT, 769, 770
 - SYSCTL_NODE, 770
 - TUNABLE_INT, 770
- scsi_da.h
 - _SCSI SCSI_DA_H, 775
 - DISK_FMT_HSEC, 775
 - DISK_FMT_RMB, 775
 - DISK_FMT_SSEC, 775
 - DISK_FMT_SURF, 776
 - FCD_CODE_MASK, 776
 - FCD_FORMATTED, 776
 - FCD_NOMEDIA, 776
 - FCD_UNFORMATTED, 776
 - FORMAT_UNIT, 776
 - FU_BFI_FORMAT, 776
 - FU_BLOCK_FORMAT, 776
 - FU_CMLPLST, 776
 - FU_DLH_DCRT, 776
 - FU_DLH_DPRY, 776

- FU_DLH_DSP, 777
- FU_DLH_FOV, 777
- FU_DLH_IMMED, 777
- FU_DLH_IP, 777
- FU_DLH_STPF, 777
- FU_DLH_VS, 777
- FU_FMT_DATA, 777
- FU_FORMAT_MASK, 777
- FU_INIT_LBA_EACH, 777
- FU_INIT_LBA_MSB, 777
- FU_INIT_NO_HDR, 777
- FU_INIT_PAT_DEFAULT, 778
- FU_INIT_PAT_REPEAT, 778
- FU_INIT_SI, 778
- FU_PHYS_FORMAT, 778
- MODE_SELECT, 778
- MODE_SENSE, 778
- READ_DEFECT_DATA_10, 778
- READ_DEFECT_DATA_12, 778
- READ_FORMAT_CAPACITIES, 778
- REASSIGN_BLOCKS, 778
- REZERO_UNIT, 778
- SMS_FLEXIBLE_GEOMETRY_PAGE, 779
- SMS_FLEXIBLE_GEOMETRY_PLEN, 779
- SMS_FORMAT_DEVICE_PAGE, 779
- SMS_FORMAT_DEVICE_PLEN, 779
- SMS_RIGID_GEOMETRY_PAGE, 779
- SMS_RIGID_GEOMETRY_PLEN, 779
- SMS_RW_ERROR_RECOVERY_PAGE, 779
- SMS_RWER_ARRE, 779
- SMS_RWER_AWRE, 779
- SMS_RWER_DCR, 779
- SMS_RWER_DTE, 779
- SMS_RWER_EER, 780
- SMS_RWER_PER, 780
- SMS_RWER_RC, 780
- SMS_RWER_TB, 780
- SRDD10_BLOCK_FORMAT, 780
- SRDD10_BYTES_FROM_INDEX_FORMAT, 780
- SRDD10_DLIST_FORMAT_MASK, 780
- SRDD10_GLIST, 780
- SRDD10_LUN_MASK, 780
- SRDD10_PHYSICAL_SECTOR_FORMAT, 780
- SRDD10_PLIST, 780
- SRDD12_BLOCK_FORMAT, 781
- SRDD12_BYTES_FROM_INDEX_FORMAT, 781
- SRDD12_DLIST_FORMAT_MASK, 781
- SRDD12_GLIST, 781
- SRDD12_LUN_MASK, 781
- SRDD12_PHYSICAL_SECTOR_FORMAT, 781
- SRDD12_PLIST, 781
- SRDDH10_BLOCK_FORMAT, 781
- SRDDH10_BYTES_FROM_INDEX_FORMAT, 781
- SRDDH10_DLIST_FORMAT_MASK, 781
- SRDDH10_GLIST, 781
- SRDDH10_PHYSICAL_SECTOR_FORMAT, 782
- SRDDH10_PLIST, 782
- SRDDH12_BLOCK_FORMAT, 782
- SRDDH12_BYTES_FROM_INDEX_FORMAT, 782
- SRDDH12_DLIST_FORMAT_MASK, 782
- SRDDH12_GLIST, 782
- SRDDH12_PHYSICAL_SECTOR_FORMAT, 782
- SRDDH12_PLIST, 782
- SRFC_LUN_MASK, 782
- SRZU_LUN_MASK, 782
- SVFY_BYTECHK, 782
- SVFY_DPO, 783
- SVFY_LUN_MASK, 783
- SVFY_RELADR, 783
- SWVY_BYTECHK, 783
- SWVY_DPO, 783
- SWVY_LUN_MASK, 783
- SWVY_RELADR, 783
- VERIFY, 783
- WRITE_AND_VERIFY, 783
- scsi_da_rw_recovery_page, 213
 - byte3, 213
 - correction_span, 213
 - data_strobe_offset_cnt, 213
 - head_offset_count, 213
 - page_code, 213
 - page_length, 213
 - read_retry_count, 213
 - recovery_time_limit, 214
 - reserved, 214
 - reserved2, 214
 - write_retry_count, 214
- scsi_data_comp_page
 - scsi_sa.h, 948
- scsi_data_compression_page, 215
 - comp_algorithm, 215
 - dce_and_dcc, 215
 - dde_and_red, 215
 - decomp_algorithm, 215
 - page_code, 215
 - page_length, 215
 - reserved, 216
- SCSI_DEFAULT_DENSITY
 - scsi_all.h, 623
- scsi_defect_desc_block, 217

- address, 217
- scsi_defect_desc_bytes_from_index, 218
 - bytes_from_index, 218
 - cylinder, 218
 - head, 218
- scsi_defect_desc_phys_sector, 219
 - cylinder, 219
 - head, 219
 - sector, 219
- SCSI_DELAY
 - scsi_all.c, 592
- scsi_delay
 - scsi_all.c, 605
 - scsi_all.h, 657
- SCSI_DENSITY_HALFINCH_1600
 - scsi_sa.h, 943
- SCSI_DENSITY_HALFINCH_6250
 - scsi_sa.h, 943
- SCSI_DENSITY_HALFINCH_6250C
 - scsi_sa.h, 943
- SCSI_DENSITY_HALFINCH_800
 - scsi_sa.h, 943
- SCSI_DENSITY_HALFINCH_PE
 - scsi_sa.h, 943
- SCSI_DENSITY_QIC_11_4TRK
 - scsi_sa.h, 943
- SCSI_DENSITY_QIC_11_9TRK
 - scsi_sa.h, 943
- SCSI_DENSITY_QIC_120
 - scsi_sa.h, 944
- SCSI_DENSITY_QIC_1320
 - scsi_sa.h, 944
- SCSI_DENSITY_QIC_150
 - scsi_sa.h, 944
- SCSI_DENSITY_QIC_24
 - scsi_sa.h, 944
- SCSI_DENSITY_QIC_2GB
 - scsi_sa.h, 944
- SCSI_DENSITY_QIC_3080
 - scsi_sa.h, 944
- SCSI_DENSITY_QIC_4GB
 - scsi_sa.h, 944
- SCSI_DENSITY_QIC_525_320
 - scsi_sa.h, 944
- scsi_dev_conf_page, 220
 - active_partition, 220
 - byte10, 220
 - byte2, 220
 - byte8, 220
 - ew_bufsize, 220
 - extra_wp, 220
 - gap_size, 221
 - pagecode, 221
 - pagelength, 221
 - rb_empty_ratio, 221
 - sel_comp_alg, 221
 - wb_full_ratio, 221
 - wrdelay_time, 221
- scsi_dvcfg.h
 - DVF SCSI BITS, 784
 - DVF SCSI DEFCFG, 784
 - DVF SCSI DISC, 784
 - DVF SCSI LINK, 784
 - DVF SCSI NOPARITY, 784
 - DVF SCSI OFFSET, 785
 - DVF SCSI PERIOD, 785
 - DVF SCSI QTAG, 785
 - DVF SCSI SAVESP, 785
 - DVF SCSI SP0, 785
 - DVF SCSI SP1, 785
 - DVF SCSI SYNC, 785
 - DVF SCSI SYNCMASK, 785
 - DVF SCSI WAIT, 785
- scsi_erase, 222
 - control, 222
 - lun_imm_long, 222
 - opcode, 222
 - reserved, 222
 - scsi_sa.c, 929
 - scsi_sa.h, 948
- scsi_error_action
 - scsi_all.c, 594
 - scsi_all.h, 647
- scsi_exchange_medium, 223
 - byte2, 223
 - control, 223
 - fdst, 223
 - invert, 223
 - opcode, 223
 - scsi_ch.c, 736
 - scsi_ch.h, 749
 - sdst, 223
 - src, 223
 - tea, 224
- scsi_extract_sense
 - scsi_all.h, 647
- scsi_format_unit, 225
 - byte2, 225
 - control, 225
 - interleave, 225
 - opcode, 225
 - vendor_specific, 225
- scsi_generic, 226
 - bytes, 226
 - opcode, 226
- scsi_initialize_element_status, 227
 - byte2, 227
 - control, 227

- opcode, 227
- reserved, 227
- scsi_ch.c, 736
- scsi_ch.h, 749
- scsi_inquiry, 228
 - byte2, 228
 - control, 228
 - length, 228
 - opcode, 228
 - page_code, 228
 - reserved, 228
 - scsi_all.c, 594
 - scsi_all.h, 647
- scsi_inquiry_data, 230
 - additional_length, 230
 - dev_qual2, 230
 - device, 230
 - flags, 231
 - product, 231
 - reserved, 231
 - reserved2, 231
 - reserved3, 231
 - response_format, 231
 - revision, 231
 - spc2_flags, 231
 - spi3data, 231
 - vendor, 231
 - vendor_specific0, 232
 - vendor_specific1, 232
 - version, 232
 - version1, 232
 - version2, 232
 - version3, 232
 - version4, 232
 - version5, 232
 - version6, 232
 - version7, 232
 - version8, 232
- scsi_inquiry_match
 - scsi_all.c, 595
 - scsi_all.h, 648
- scsi_inquiry_pattern, 234
 - media_type, 234
 - product, 234
 - revision, 234
 - type, 234
 - vendor, 234
- scsi_interpret_sense
 - scsi_all.h, 648
- scsi_iu.h
 - _SCSI_SCSI_IU_H, 786
 - SIU_PFC_CIU_FIELDS_INVALID, 786
 - SIU_PFC_ILLEGAL_REQUEST, 786
 - SIU_PFC_INVALID_TYPE_CODE, 786
- SIU_PFC_NONE, 786
- SIU_PFC_TMF_FAILED, 787
- SIU_PFC_TMF_NOT_SUPPORTED, 787
- SIU_PKTFAIL_CODE, 787
- SIU_PKTFAIL_OFFSET, 787
- SIU_RSPVALID, 787
- SIU_SENSE_OFFSET, 787
- SIU_SNSVALID, 787
- SIU_TASKMGMT_ABORT_TASK, 787
- SIU_TASKMGMT_ABORT_TASK_SET, 787
- SIU_TASKMGMT_CLEAR_ACA, 787
- SIU_TASKMGMT_CLEAR_TASK_SET, 788
- SIU_TASKMGMT_LUN_RESET, 788
- SIU_TASKMGMT_NONE, 788
- SIU_TASKMGMT_TARGET_RESET, 788
- scsi_load_unload, 235
 - control, 235
 - eot_reten_load, 235
 - immediate, 235
 - opcode, 235
 - reserved, 235
 - scsi_sa.c, 930
 - scsi_sa.h, 948
- scsi_log_header, 236
 - datalen, 236
 - page, 236
 - reserved, 236
- scsi_log_param_header, 237
 - param_code, 237
 - param_control, 237
 - param_len, 237
- scsi_log_select, 238
 - byte2, 238
 - control, 238
 - length, 238
 - opcode, 238
 - page, 238
 - reserved, 238
 - scsi_all.c, 595
 - scsi_all.h, 648
- scsi_log_sense, 239
 - byte2, 239
 - control, 239
 - length, 239
 - opcode, 239
 - page, 239
 - paramptr, 239
 - reserved, 239
 - scsi_all.c, 595
 - scsi_all.h, 648
- scsi_low.c
 - __FBSDID, 800

- inq_cmd, 827
- MALLOC_DEFINE, 800
- MSG_RELEASE_ATN, 794
- MSGCMD_LUN, 794
- MSGIN_DATA_LAST, 794
- MSGIN_OFFSET, 794
- MSGIN_PERIOD, 794
- MSGIN_WIDTHP, 794
- MSGINPTR_CLR, 795
- phase, 827
- scsi_low_abort_ccb, 800
- SCSI_LOW_ABORT_CHECK, 795
- scsi_low_activate, 800
- scsi_low_activate_qtag, 801
- scsi_low_alloc_li, 801
- scsi_low_alloc_qtag, 801
- scsi_low_alloc_ti, 802
- scsi_low_arbit_fail, 802
- scsi_low_attach, 802
- SCSI_LOW_ATTEN_CHECK, 795
- scsi_low_bus_idle, 803
- scsi_low_bus_release, 803
- scsi_low_bus_reset, 803
- scsi_low_calcf_lun, 804
- scsi_low_calcf_show, 804
- scsi_low_calcf_target, 804
- scsi_low_ccb_message_assert, 804
- scsi_low_ccb_message_clear, 805
- scsi_low_ccb_message_exec, 805
- scsi_low_ccb_message_retry, 805
- scsi_low_cmd, 805
- SCSI_LOW_CMD_ABORT_WARNING, 795
- scsi_low_cmd_flags, 827
- SCSI_LOW_CMD_ORDERED_QTAG, 795
- SCSI_LOW_CMD_RESIDUAL_CHK, 795
- SCSI_LOW_CMDLNK_CHECK, 795
- SCSI_LOW_CMDLNK_NOK, 795
- scsi_low_data, 805
- scsi_low_deactivate, 806
- scsi_low_deactivate_qtag, 806
- scsi_low_dealloc_qtag, 806
- SCSI_LOW_DEBUG, 795
- scsi_low_debug, 827
- SCSI_LOW_DEBUG_ACTION, 796
- SCSI_LOW_DEBUG_CALCF, 796
- SCSI_LOW_DEBUG_DISC, 796
- SCSI_LOW_DEBUG_DONE, 796
- SCSI_LOW_DEBUG_GO, 796
- SCSI_LOW_DEBUG_SENSE, 796
- SCSI_LOW_DEBUG_TEST_GO, 796
- scsi_low_dettach, 806
- scsi_low_disconnected, 806
- SCSI_LOW_DISK_DISC, 796
- SCSI_LOW_DISK_LFLAGS, 796
- SCSI_LOW_DISK_LINK, 797
- SCSI_LOW_DISK_PARITY, 797
- SCSI_LOW_DISK_QTAG, 797
- SCSI_LOW_DISK_SYNC, 797
- SCSI_LOW_DISK_TFLAGS, 797
- SCSI_LOW_DISK_WIDE, 797
- SCSI_LOW_DISK_WIDE_16, 797
- SCSI_LOW_DISK_WIDE_32, 797
- scsi_low_done, 807
- SCSI_LOW_DONE_COMPLETE, 798
- SCSI_LOW_DONE_RETRY, 798
- scsi_low_engage, 808
- scsi_low_enqueue, 808
- scsi_low_errfunc_identify, 809
- scsi_low_errfunc_qtag, 809
- scsi_low_errfunc_synch, 810
- scsi_low_errfunc_wide, 810
- scsi_low_establish_ccb, 810
- scsi_low_find_ccb, 810
- SCSI_LOW_FLAGS_QUIRKS_OK, 798
- scsi_low_free_ti, 811
- SCSI_LOW_INFO, 798
- scsi_low_info, 811
- scsi_low_init, 811
- scsi_low_init_msgsys, 811
- SCSI_LOW_INLINE, 798
- scsi_low_is_busy, 812
- SCSI_LOW_MAX_ATTEN_CHECK, 798
- scsi_low_message_enqueue, 812
- SCSI_LOW_MSG_ABORT_OK, 798
- scsi_low_msgfunc_abort, 812
- scsi_low_msgfunc_identify, 812
- scsi_low_msgfunc_qabort, 813
- scsi_low_msgfunc_qtag, 813
- scsi_low_msgfunc_reset, 813
- scsi_low_msgfunc_synch, 813
- scsi_low_msgfunc_wide, 813
- scsi_low_msgin, 813
- scsi_low_msgin_data, 827
- scsi_low_msginfunc_cc, 814
- scsi_low_msginfunc_disc, 814
- scsi_low_msginfunc_ext, 814
- scsi_low_msginfunc_i_wide_residue, 815
- scsi_low_msginfunc_lcc, 815
- scsi_low_msginfunc_msg_reject, 815
- scsi_low_msginfunc_noop, 815
- scsi_low_msginfunc_parity, 815
- scsi_low_msginfunc_rejop, 816
- scsi_low_msginfunc_rp, 816
- scsi_low_msginfunc_sdp, 816
- scsi_low_msginfunc_simple_qtag, 816
- scsi_low_msgout, 817
- scsi_low_msgout_data, 828

- SCSI_LOW_NEGOTIATE_BEFORE_-SENSE, 798
- SCSI_LOW_NEXUS_CHECK, 798
- scsi_low_poll, 817
- SCSI_LOW_POLL_HZ, 799
- scsi_low_print, 818
- SCSI_LOW_QTAG_OK, 799
- scsi_low_reselected, 818
- scsi_low_reset_nexus, 819
- scsi_low_reset_nexus_lun, 819
- scsi_low_reset_nexus_target, 820
- scsi_low_restart, 820
- scsi_low_resume, 821
- scsi_low_revoke_ccb, 821
- scsi_low_sense_abort_start, 822
- scsi_low_setup_done, 822
- scsi_low_setup_start, 823
- scsi_low_start, 823
- SCSI_LOW_START_NO_QTAG, 799
- SCSI_LOW_START_QTAG, 799
- scsi_low_start_up, 824
- SCSI_LOW_START_UP_CHECK, 799
- SCSI_LOW_STATICS, 799
- scsi_low_synch, 824
- scsi_low_test, 828
- scsi_low_test_abort, 824
- scsi_low_test_atten, 824
- scsi_low_test_cmdlnk, 825
- scsi_low_test_id, 828
- scsi_low_timeout, 825
- scsi_low_timeout_check, 825
- scsi_low_translate_error_code, 826
- scsi_low_twiddle_wait, 826
- scsi_low_unit_ready_cmd, 826
- scsi_low_version_major, 828
- scsi_low_version_minor, 828
- scsi_low_wide, 827
- sl_tab, 828
- SLSC_MODE_SENSE_SHORT, 799
- sms_cmd, 828
- ss_cmd, 829
- tw_chars, 829
- tw_pos, 829
- TWIDDLEWAIT, 799
- unit_ready_cmd, 829
- scsi_low.h
 - ABORTIO, 836
 - BUSYERR, 836
 - c, 865
 - CAM, 836
 - CCB_AUTOSENSE, 836
 - CCB_CLEARQ, 836
 - CCB_DISCQ, 836
 - CCB_INTERNAL, 836
 - CCB_NORETRY, 837
 - CCB_NOSDONE, 837
 - CCB_POLLED, 837
 - CCB_SCSIIO, 837
 - CCB_SENSE, 837
 - CCB_SILENT, 837
 - CCB_STARTQ, 837
 - CCB_URGENT, 837
 - CFG_ASYNC, 837
 - CFG_NOATTEN, 838
 - CFG_NODISC, 838
 - CFG_NOPARITY, 838
 - CFG_NOQTAG, 838
 - FATALIO, 838
 - HW_INACTIVE, 838
 - HW_INITIALIZING, 838
 - HW_PDMASRT, 838
 - HW_POWDOWN, 839
 - HW_POWERCTRL, 839
 - HW_READ_PADDING, 839
 - HW_RESUME, 839
 - HW_WRITE_PADDING, 839
 - LIST_HEAD, 858
 - mask, 865
 - MKMSG_EXTEND, 839
 - msg, 865
 - MSG_ABORT, 839
 - MSG_ABORT_QTAG, 839
 - MSG_CLEAR_QTAG, 839
 - MSG_COMP, 839
 - MSG_DISCON, 839
 - MSG_EXTEND, 839
 - MSG_EXTEND_MDPCODE, 839
 - MSG_EXTEND_MDPLEN, 839
 - MSG_EXTEND_SYNCHCODE, 840
 - MSG_EXTEND_SYNCHLEN, 840
 - MSG_EXTEND_WIDECODE, 840
 - MSG_EXTEND_WIDELLEN, 840
 - MSG_HEAD_QTAG, 840
 - MSG_I_ERROR, 840
 - MSG_IDENTIFY, 840
 - MSG_IDENTIFY_DISCPRIV, 840
 - MSG_LCOMP, 840
 - MSG_LCOMP_F, 840
 - MSG_NOOP, 840
 - MSG_ORDERED_QTAG, 840
 - MSG_PARITY, 841
 - MSG_REJECT, 841
 - MSG_RESET, 841
 - MSG_RESTORESP, 841
 - MSG_SAVESP, 841
 - MSG_SIMPLE_QTAG, 841
 - MSG_TERM_IO, 841
 - MSGERR, 841

- MSGPH_ABORT, 841
- MSGPH_CMDC, 841
- MSGPH_DISC, 841
- MSGPH_LCTERM, 841
- MSGPH_NULL, 841
- MSGPH_RESET, 841
- MSGPH_TERM, 842
- now, 865
- PARITYERR, 842
- PDMAERR, 842
- PENDINGIO, 842
- PH_ARBSTART, 842
- PH_CMD, 842
- PH_DATA, 842
- PH_DISC, 842
- PH_MSGIN, 842
- PH_MSGOUT, 842
- PH_NULL, 843
- PH_RESEL, 843
- PH_SELECTED, 843
- PH_SELSTART, 843
- PH_STAT, 843
- SC_LOW_ATTEN_T, 843
- SC_LOW_BUSRST_T, 843
- SC_LOW_INIT_T, 843
- SC_LOW_LUN_INIT_T, 843
- SC_LOW_MSG_T, 843
- SC_LOW_NEXUS_T, 844
- SC_LOW_POLL_T, 844
- SC_LOW_POWER_T, 844
- SC_LOW_SELECT_T, 844
- sc_low_t, 857
- SC_LOW_TARG_INIT_T, 844
- SC_LOW_TIMEOUT_T, 844
- SCSI2_RESET_DELAY, 844
- scsi_low_activate, 858
- SCSI_LOW_ALT_QTAG_ALLOCATE, 844
- scsi_low_arbit_fail, 858
- scsi_low_arbit_win, 859
- SCSI_LOW_ASSERT_ATN, 844
- scsi_low_assert_msg, 859
- scsi_low_attach, 859
- scsi_low_attention, 859
- SCSI_LOW_BITS, 844
- scsi_low_bus_idle, 859
- scsi_low_bus_reset, 860
- SCSI_LOW_BUS_WIDTH_16, 845
- SCSI_LOW_BUS_WIDTH_32, 845
- SCSI_LOW_BUS_WIDTH_8, 845
- SCSI_LOW_BZERO, 845
- scsi_low_cmd, 860
- scsi_low_data, 860
- scsi_low_data_finish, 861
- SCSI_LOW_DATA_PE, 845
- scsi_low_deactivate, 861
- SCSI_LOW_DEASSERT_ATN, 845
- SCSI_LOW_DEBUGGER, 845
- SCSI_LOW_DEFCFG, 845
- SCSI_LOW_DELAY, 846
- scsi_low_dettach, 861
- SCSI_LOW_DIAGNOSTIC, 846
- SCSI_LOW_DISC, 846
- scsi_low_disconnected, 861
- SCSI_LOW_ENGAGE, 846
- SCSI_LOW_ERRORBITS, 846
- SCSI_LOW_INFO_ALLOC, 846
- SCSI_LOW_INFO_DEALLOC, 846
- SCSI_LOW_INFO_REVOKE, 846
- SCSI_LOW_INTERFACE_CAM, 846
- scsi_low_is_busy, 862
- scsi_low_is_disconnect_ok, 862
- scsi_low_is_msgout_continue, 862
- SCSI_LOW_LINK, 847
- SCSI_LOW_LUN_FLAGS_ALL_VALID, 847
- SCSI_LOW_LUN_FLAGS_DISK_VALID, 847
- SCSI_LOW_LUN_FLAGS_QUIRKS_VALID, 847
- SCSI_LOW_LUN_FLAGS_USER_VALID, 847
- SCSI_LOW_LUN_INQ, 847
- SCSI_LOW_LUN_MODEQ, 847
- SCSI_LOW_LUN_OK, 847
- SCSI_LOW_LUN_SLEEP, 848
- SCSI_LOW_LUN_START, 848
- SCSI_LOW_MAX_MSGLEN, 848
- SCSI_LOW_MAX_PHCHANGES, 848
- SCSI_LOW_MAX_RETRY, 848
- SCSI_LOW_MAX_SELECTION_RETRY, 848
- SCSI_LOW_MAXNEXUS, 848
- SCSI_LOW_MIN_TOUT, 848
- SCSI_LOW_MSG_ABORT, 849
- SCSI_LOW_MSG_ABORT_QTAG, 849
- SCSI_LOW_MSG_ALL, 849
- SCSI_LOW_MSG_CLEAR_QTAG, 849
- SCSI_LOW_MSG_ERROR, 849
- SCSI_LOW_MSG_HEAD_QTAG, 849
- SCSI_LOW_MSG_IDENTIFY, 849
- SCSI_LOW_MSG_LAST, 849
- SCSI_LOW_MSG_LOG_DATALEN, 849
- SCSI_LOW_MSG_NOOP, 849
- SCSI_LOW_MSG_ORDERED_QTAG, 850
- SCSI_LOW_MSG_PARITY, 850
- SCSI_LOW_MSG_REJECT, 850
- SCSI_LOW_MSG_RESET, 850
- SCSI_LOW_MSG_SIMPLE_QTAG, 850

- SCSI_LOW_MSG_SYNC, 850
- SCSI_LOW_MSG_TERMIO, 850
- SCSI_LOW_MSG_WIDE, 850
- scsi_low_msgin, 862
- scsi_low_msgout, 863
- SCSI_LOW_MSGOUT_INIT, 850
- SCSI_LOW_MSGOUT_UNIFY, 851
- SCSI_LOW_NCCB, 851
- SCSI_LOW_NOPARITY, 851
- SCSI_LOW_NTARGETS, 851
- SCSI_LOW_OFFSET, 851
- scsi_low_osdep_sense_data_t, 857
- SCSI_LOW_PERIOD, 851
- SCSI_LOW_POWDOWN, 851
- SCSI_LOW_POWDOWN_TC, 851
- scsi_low_print, 863
- SCSI_LOW_QFLAG_CA_QCLEAR, 851
- SCSI_LOW_QTAG, 852
- SCSI_LOW_READ, 852
- scsi_low_reselected, 864
- scsi_low_restart, 864
- SCSI_LOW_RESTART_HARD, 852
- SCSI_LOW_RESTART_SOFT, 852
- SCSI_LOW_RWUNK, 852
- SCSI_LOW_SAVESP, 852
- SCSI_LOW_SETUP_MSGPHASE, 852
- SCSI_LOW_SETUP_PHASE, 853
- SCSI_LOW_SPLSCSI, 853
- SCSI_LOW_START_FAIL, 853
- SCSI_LOW_START_OK, 853
- scsi_low_statusin, 865
- SCSI_LOW_SYNC, 853
- scsi_low_tag_t, 858
- SCSI_LOW_TARG_FLAGS_ALL_VALID, 853
- SCSI_LOW_TARG_FLAGS_DISK_VALID, 853
- SCSI_LOW_TARG_FLAGS_QUIRKS_-VALID, 854
- SCSI_LOW_TARG_FLAGS_USER_VALID, 854
- SCSI_LOW_TIMEOUT_CH_ENGAGE, 854
- SCSI_LOW_TIMEOUT_CH_IO, 854
- SCSI_LOW_TIMEOUT_CH_RECOVER, 854
- SCSI_LOW_TIMEOUT_CHECK_-INTERVAL, 854
- SCSI_LOW_TIMEOUT_HZ, 854
- SCSI_LOW_TIMEOUT_START, 854
- SCSI_LOW_TIMEOUT_STOP, 854
- SCSI_LOW_UNKLUN, 855
- SCSI_LOW_UNKTAG, 855
- SCSI_LOW_WAIT, 855
- SCSI_LOW_WRITE, 855
- SELTIMEOUTIO, 855
- SENSEERR, 855
- SENSEIO, 855
- SHOW_ALL_NEG, 855
- SHOW_CALCF_RES, 855
- SHOW_PROBE_RES, 856
- SHOW_SYNC_NEG, 856
- SHOW_WIDE_NEG, 856
- sl_dev, 856
- ST_BUSY, 856
- ST_CHKCOND, 856
- ST_CMDTERM, 856
- ST_CONFLICT, 856
- ST_GOOD, 856
- ST_INTERGOOD, 856
- ST_INTERMET, 857
- ST_MET, 857
- ST_QUEFULL, 857
- ST_UNKNOWN, 857
- STATERR, 857
- TAILQ_HEAD, 865
- ti, 865, 866
- TIMEOUTIO, 857
- UACAERR, 857
- UBFERR, 857
- scsi_low_abort_ccb
- scsi_low.c, 800
- SCSI_LOW_ABORT_CHECK
- scsi_low.c, 795
- scsi_low_activate
- scsi_low.c, 800
- scsi_low.h, 858
- scsi_low_activate_pisa
- scsi_low_pisa.c, 868
- scsi_low_pisa.h, 869
- scsi_low_activate_qtag
- scsi_low.c, 801
- scsi_low_alloc_li
- scsi_low.c, 801
- scsi_low_alloc_qtag
- scsi_low.c, 801
- scsi_low_alloc_ti
- scsi_low.c, 802
- SCSI_LOW_ALT_QTAG_ALLOCATE
- scsi_low.h, 844
- scsi_low_arbit_fail
- scsi_low.c, 802
- scsi_low.h, 858
- scsi_low_arbit_win
- scsi_low.h, 859
- SCSI_LOW_ASSERT_ATN
- scsi_low.h, 844
- scsi_low_assert_msg
- scsi_low.h, 859

- scsi_low_attach
 - scsi_low.c, 802
 - scsi_low.h, 859
- SCSI_LOW_ATTEN_CHECK
 - scsi_low.c, 795
- scsi_low_attention
 - scsi_low.h, 859
 - scsi_low_funcs, 242
- SCSI_LOW_BITS
 - scsi_low.h, 844
- scsi_low_bus_idle
 - scsi_low.c, 803
 - scsi_low.h, 859
- scsi_low_bus_release
 - scsi_low.c, 803
- scsi_low_bus_reset
 - scsi_low.c, 803
 - scsi_low.h, 860
 - scsi_low_funcs, 242
- SCSI_LOW_BUS_WIDTH_16
 - scsi_low.h, 845
- SCSI_LOW_BUS_WIDTH_32
 - scsi_low.h, 845
- SCSI_LOW_BUS_WIDTH_8
 - scsi_low.h, 845
- SCSI_LOW_BZERO
 - scsi_low.h, 845
- scsi_low_calcf_lun
 - scsi_low.c, 804
- scsi_low_calcf_show
 - scsi_low.c, 804
- scsi_low_calcf_target
 - scsi_low.c, 804
- scsi_low_ccb_message_assert
 - scsi_low.c, 804
- scsi_low_ccb_message_clear
 - scsi_low.c, 805
- scsi_low_ccb_message_exec
 - scsi_low.c, 805
- scsi_low_ccb_message_retry
 - scsi_low.c, 805
- scsi_low_cmd
 - scsi_low.c, 805
 - scsi_low.h, 860
- SCSI_LOW_CMD_ABORT_WARNING
 - scsi_low.c, 795
- scsi_low_cmd_flags
 - scsi_low.c, 827
- SCSI_LOW_CMD_ORDERED_QTAG
 - scsi_low.c, 795
- SCSI_LOW_CMD_RESIDUAL_CHK
 - scsi_low.c, 795
- SCSI_LOW_CMDLNK_CHECK
 - scsi_low.c, 795
- SCSI_LOW_CMDLNK_NOK
 - scsi_low.c, 795
- scsi_low_data
 - scsi_low.c, 805
 - scsi_low.h, 860
- scsi_low_data_finish
 - scsi_low.h, 861
- SCSI_LOW_DATA_PE
 - scsi_low.h, 845
- scsi_low_deactivate
 - scsi_low.c, 806
 - scsi_low.h, 861
- scsi_low_deactivate_pisa
 - scsi_low_pisa.c, 868
 - scsi_low_pisa.h, 869
- scsi_low_deactivate_qtag
 - scsi_low.c, 806
- scsi_low_dealloc_qtag
 - scsi_low.c, 806
- SCSI_LOW_DEASSERT_ATN
 - scsi_low.h, 845
- SCSI_LOW_DEBUG
 - scsi_low.c, 795
- scsi_low_debug
 - scsi_low.c, 827
- SCSI_LOW_DEBUG_ACTION
 - scsi_low.c, 796
- SCSI_LOW_DEBUG_CALCF
 - scsi_low.c, 796
- SCSI_LOW_DEBUG_DISC
 - scsi_low.c, 796
- SCSI_LOW_DEBUG_DONE
 - scsi_low.c, 796
- SCSI_LOW_DEBUG_GO
 - scsi_low.c, 796
- SCSI_LOW_DEBUG_SENSE
 - scsi_low.c, 796
- SCSI_LOW_DEBUG_TEST_GO
 - scsi_low.c, 796
- SCSI_LOW_DEBUGGER
 - scsi_low.h, 845
- SCSI_LOW_DEFCFG
 - scsi_low.h, 845
- SCSI_LOW_DELAY
 - scsi_low.h, 846
- scsi_low_dettach
 - scsi_low.c, 806
 - scsi_low.h, 861
- SCSI_LOW_DIAGNOSTIC
 - scsi_low.h, 846
- SCSI_LOW_DISC
 - scsi_low.h, 846
- scsi_low_disconnected
 - scsi_low.c, 806

- scsi_low.h, [861](#)
- SCSI_LOW_DISK_DISC
 - scsi_low.c, [796](#)
- SCSI_LOW_DISK_LFLAGS
 - scsi_low.c, [796](#)
- SCSI_LOW_DISK_LINK
 - scsi_low.c, [797](#)
- SCSI_LOW_DISK_PARITY
 - scsi_low.c, [797](#)
- SCSI_LOW_DISK_QTAG
 - scsi_low.c, [797](#)
- SCSI_LOW_DISK_SYNC
 - scsi_low.c, [797](#)
- SCSI_LOW_DISK_TFLAGS
 - scsi_low.c, [797](#)
- SCSI_LOW_DISK_WIDE
 - scsi_low.c, [797](#)
- SCSI_LOW_DISK_WIDE_16
 - scsi_low.c, [797](#)
- SCSI_LOW_DISK_WIDE_32
 - scsi_low.c, [797](#)
- scsi_low_done
 - scsi_low.c, [807](#)
- SCSI_LOW_DONE_COMPLETE
 - scsi_low.c, [798](#)
- SCSI_LOW_DONE_RETRY
 - scsi_low.c, [798](#)
- SCSI_LOW_ENGAGE
 - scsi_low.h, [846](#)
- scsi_low_engage
 - scsi_low.c, [808](#)
- scsi_low_enqueue
 - scsi_low.c, [808](#)
- scsi_low_errfunc_identify
 - scsi_low.c, [809](#)
- scsi_low_errfunc_qtag
 - scsi_low.c, [809](#)
- scsi_low_errfunc_synch
 - scsi_low.c, [810](#)
- scsi_low_errfunc_wide
 - scsi_low.c, [810](#)
- scsi_low_error_code, [241](#)
 - error_bits, [241](#)
 - error_code, [241](#)
- SCSI_LOW_ERRORBITS
 - scsi_low.h, [846](#)
- scsi_low_establish_ccb
 - scsi_low.c, [810](#)
- scsi_low_establish_ccb_nexus
 - scsi_low_funcs, [242](#)
- scsi_low_establish_lun_nexus
 - scsi_low_funcs, [242](#)
- scsi_low_find_ccb
 - scsi_low.c, [810](#)
- SCSI_LOW_FLAGS_QUIRKS_OK
 - scsi_low.c, [798](#)
- scsi_low_free_ti
 - scsi_low.c, [811](#)
- scsi_low_funcs, [242](#)
 - scsi_low_attention, [242](#)
 - scsi_low_bus_reset, [242](#)
 - scsi_low_establish_ccb_nexus, [242](#)
 - scsi_low_establish_lun_nexus, [242](#)
 - scsi_low_init, [242](#)
 - scsi_low_ioctl, [242](#)
 - scsi_low_lun_init, [242](#)
 - scsi_low_msg, [242](#)
 - scsi_low_poll, [243](#)
 - scsi_low_power, [243](#)
 - scsi_low_start_bus, [243](#)
 - scsi_low_targ_init, [243](#)
 - scsi_low_timeout, [243](#)
- SCSI_LOW_INFO
 - scsi_low.c, [798](#)
- scsi_low_info
 - scsi_low.c, [811](#)
- SCSI_LOW_INFO_ALLOC
 - scsi_low.h, [846](#)
- SCSI_LOW_INFO_DEALLOC
 - scsi_low.h, [846](#)
- SCSI_LOW_INFO_REVOKE
 - scsi_low.h, [846](#)
- scsi_low_init
 - scsi_low.c, [811](#)
 - scsi_low_funcs, [242](#)
- scsi_low_init_msgsys
 - scsi_low.c, [811](#)
- SCSI_LOW_INLINE
 - scsi_low.c, [798](#)
- SCSI_LOW_INTERFACE_CAM
 - scsi_low.h, [846](#)
- scsi_low_ioctl
 - scsi_low_funcs, [242](#)
- scsi_low_is_busy
 - scsi_low.c, [812](#)
 - scsi_low.h, [862](#)
- scsi_low_is_disconnect_ok
 - scsi_low.h, [862](#)
- scsi_low_is_msgout_continue
 - scsi_low.h, [862](#)
- SCSI_LOW_LINK
 - scsi_low.h, [847](#)
- SCSI_LOW_LUN_FLAGS_ALL_VALID
 - scsi_low.h, [847](#)
- SCSI_LOW_LUN_FLAGS_DISK_VALID
 - scsi_low.h, [847](#)
- SCSI_LOW_LUN_FLAGS_QUIRKS_VALID
 - scsi_low.h, [847](#)

- SCSI_LOW_LUN_FLAGS_USER_VALID
 - scsi_low.h, 847
- scsi_low_lun_init
 - scsi_low_funcs, 242
- SCSI_LOW_LUN_INQ
 - scsi_low.h, 847
- SCSI_LOW_LUN_MODEQ
 - scsi_low.h, 847
- SCSI_LOW_LUN_OK
 - scsi_low.h, 847
- SCSI_LOW_LUN_SLEEP
 - scsi_low.h, 848
- SCSI_LOW_LUN_START
 - scsi_low.h, 848
- SCSI_LOW_MAX_ATTEN_CHECK
 - scsi_low.c, 798
- SCSI_LOW_MAX_MSGLEN
 - scsi_low.h, 848
- SCSI_LOW_MAX_PHCHANGES
 - scsi_low.h, 848
- SCSI_LOW_MAX_RETRY
 - scsi_low.h, 848
- SCSI_LOW_MAX_SELECTION_RETRY
 - scsi_low.h, 848
- SCSI_LOW_MAXNEXUS
 - scsi_low.h, 848
- scsi_low_message_enqueue
 - scsi_low.c, 812
- SCSI_LOW_MIN_TOUT
 - scsi_low.h, 848
- scsi_low_moduledata
 - scsi_low_pisa.c, 868
- scsi_low_msg
 - scsi_low_funcs, 242
- SCSI_LOW_MSG_ABORT
 - scsi_low.h, 849
- SCSI_LOW_MSG_ABORT_OK
 - scsi_low.c, 798
- SCSI_LOW_MSG_ABORT_QTAG
 - scsi_low.h, 849
- SCSI_LOW_MSG_ALL
 - scsi_low.h, 849
- SCSI_LOW_MSG_CLEAR_QTAG
 - scsi_low.h, 849
- SCSI_LOW_MSG_ERROR
 - scsi_low.h, 849
- SCSI_LOW_MSG_HEAD_QTAG
 - scsi_low.h, 849
- SCSI_LOW_MSG_IDENTIFY
 - scsi_low.h, 849
- SCSI_LOW_MSG_LAST
 - scsi_low.h, 849
- scsi_low_msg_log, 244
 - msg, 244
- slml_msg, 244
- slml_ptr, 244
- SCSI_LOW_MSG_LOG_DATALEN
 - scsi_low.h, 849
- SCSI_LOW_MSG_NOOP
 - scsi_low.h, 849
- SCSI_LOW_MSG_ORDERED_QTAG
 - scsi_low.h, 850
- SCSI_LOW_MSG_PARITY
 - scsi_low.h, 850
- SCSI_LOW_MSG_REJECT
 - scsi_low.h, 850
- SCSI_LOW_MSG_RESET
 - scsi_low.h, 850
- SCSI_LOW_MSG_SIMPLE_QTAG
 - scsi_low.h, 850
- SCSI_LOW_MSG_SYNCH
 - scsi_low.h, 850
- SCSI_LOW_MSG_TERMIO
 - scsi_low.h, 850
- SCSI_LOW_MSG_WIDE
 - scsi_low.h, 850
- scsi_low_msgfunc_abort
 - scsi_low.c, 812
- scsi_low_msgfunc_identify
 - scsi_low.c, 812
- scsi_low_msgfunc_qabort
 - scsi_low.c, 813
- scsi_low_msgfunc_qtag
 - scsi_low.c, 813
- scsi_low_msgfunc_reset
 - scsi_low.c, 813
- scsi_low_msgfunc_synch
 - scsi_low.c, 813
- scsi_low_msgfunc_wide
 - scsi_low.c, 813
- scsi_low_msgin
 - scsi_low.c, 813
 - scsi_low.h, 862
- scsi_low_msgin_data, 245
 - md_len, 245
 - md_msgfunc, 245
 - scsi_low.c, 827
- scsi_low_msginfunc_cc
 - scsi_low.c, 814
- scsi_low_msginfunc_disc
 - scsi_low.c, 814
- scsi_low_msginfunc_ext
 - scsi_low.c, 814
- scsi_low_msginfunc_i_wide_residue
 - scsi_low.c, 815
- scsi_low_msginfunc_lcc
 - scsi_low.c, 815
- scsi_low_msginfunc_msg_reject

- scsi_low.c, 815
- scsi_low_msginfunc_noop
 - scsi_low.c, 815
- scsi_low_msginfunc_parity
 - scsi_low.c, 815
- scsi_low_msginfunc_rejop
 - scsi_low.c, 816
- scsi_low_msginfunc_rp
 - scsi_low.c, 816
- scsi_low_msginfunc_sdp
 - scsi_low.c, 816
- scsi_low_msginfunc_simple_qtag
 - scsi_low.c, 816
- scsi_low_msgout
 - scsi_low.c, 817
 - scsi_low.h, 863
- scsi_low_msgout_data, 246
 - md_condition, 246
 - md_errfunc, 246
 - md_flags, 246
 - md_msg, 246
 - md_msgfunc, 246
 - scsi_low.c, 828
- SCSI_LOW_MSGOUT_INIT
 - scsi_low.h, 850
- SCSI_LOW_MSGOUT_UNIFY
 - scsi_low.h, 851
- SCSI_LOW_NCCB
 - scsi_low.h, 851
- SCSI_LOW_NEGOTIATE_BEFORE_SENSE
 - scsi_low.c, 798
- SCSI_LOW_NEXUS_CHECK
 - scsi_low.c, 798
- SCSI_LOW_NOPARITY
 - scsi_low.h, 851
- SCSI_LOW_NTARGETS
 - scsi_low.h, 851
- SCSI_LOW_OFFSET
 - scsi_low.h, 851
- scsi_low_osdep_attach
 - scsi_low_osdep_funcs, 247
- scsi_low_osdep_ccb_setup
 - scsi_low_osdep_funcs, 247
- scsi_low_osdep_dettach
 - scsi_low_osdep_funcs, 247
- scsi_low_osdep_done
 - scsi_low_osdep_funcs, 247
- scsi_low_osdep_funcs, 247
 - scsi_low_osdep_attach, 247
 - scsi_low_osdep_ccb_setup, 247
 - scsi_low_osdep_dettach, 247
 - scsi_low_osdep_done, 247
 - scsi_low_osdep_timeout, 247
 - scsi_low_osdep_world_start, 247
- scsi_low_osdep_interface, 248
 - engage_ch, 248
 - path, 248
 - si_dev, 248
 - si_poll_count, 248
 - sim, 249
 - timeout_ch, 249
- scsi_low_osdep_lun_interface, 250
- scsi_low_osdep_sense_data_t
 - scsi_low.h, 857
- scsi_low_osdep_targ_interface, 251
- scsi_low_osdep_timeout
 - scsi_low_osdep_funcs, 247
- scsi_low_osdep_world_start
 - scsi_low_osdep_funcs, 247
- SCSI_LOW_PERIOD
 - scsi_low.h, 851
- scsi_low_pisa.c
 - __FBSDID, 868
 - DECLARE_MODULE, 868
 - MODULE_DEPEND, 868
 - MODULE_VERSION, 868
 - scsi_low_activate_pisa, 868
 - scsi_low_deactivate_pisa, 868
 - scsi_low_moduledata, 868
- scsi_low_pisa.h
 - scsi_low_activate_pisa, 869
 - scsi_low_deactivate_pisa, 869
- scsi_low_poll
 - scsi_low.c, 817
 - scsi_low_funcs, 243
- SCSI_LOW_POLL_HZ
 - scsi_low.c, 799
- SCSI_LOW_POWDOWN
 - scsi_low.h, 851
- SCSI_LOW_POWDOWN_TC
 - scsi_low.h, 851
- scsi_low_power
 - scsi_low_funcs, 243
- scsi_low_print
 - scsi_low.c, 818
 - scsi_low.h, 863
- SCSI_LOW_QFLAG_CA_QCLEAR
 - scsi_low.h, 851
- SCSI_LOW_QTAG
 - scsi_low.h, 852
- SCSI_LOW_QTAG_OK
 - scsi_low.c, 799
- SCSI_LOW_READ
 - scsi_low.h, 852
- scsi_low_reselected
 - scsi_low.c, 818
 - scsi_low.h, 864
- scsi_low_reset_nexus

- scsi_low.c, 819
- scsi_low_reset_nexus_lun
 - scsi_low.c, 819
- scsi_low_reset_nexus_target
 - scsi_low.c, 820
- scsi_low_restart
 - scsi_low.c, 820
 - scsi_low.h, 864
- SCSI_LOW_RESTART_HARD
 - scsi_low.h, 852
- SCSI_LOW_RESTART_SOFT
 - scsi_low.h, 852
- scsi_low_resume
 - scsi_low.c, 821
- scsi_low_revoke_ccb
 - scsi_low.c, 821
- SCSI_LOW_RWUNK
 - scsi_low.h, 852
- SCSI_LOW_SAVESP
 - scsi_low.h, 852
- scsi_low_sense_abort_start
 - scsi_low.c, 822
- scsi_low_setup_done
 - scsi_low.c, 822
- SCSI_LOW_SETUP_MSGPHASE
 - scsi_low.h, 852
- SCSI_LOW_SETUP_PHASE
 - scsi_low.h, 853
- scsi_low_setup_start
 - scsi_low.c, 823
- scsi_low_softc, 252
 - LIST_ENTRY, 253
 - sl_active, 253
 - sl_atten, 253
 - sl_cfgflags, 253
 - sl_clear_atten, 253
 - sl_disc, 253
 - sl_error, 253
 - sl_flags, 254
 - sl_funcs, 254
 - sl_hostid, 254
 - sl_Lnexus, 254
 - sl_max_retry, 254
 - sl_msgphase, 254
 - sl_nexus_call, 254
 - sl_nio, 254
 - sl_nluns, 254
 - sl_ntargs, 254
 - sl_openings, 255
 - sl_osdep_fp, 255
 - sl_ph_count, 255
 - sl_powc, 255
 - sl_Qnexus, 255
 - sl_retry_sel, 255
 - sl_rstep, 255
 - sl_scp, 255
 - sl_selid, 255
 - sl_show_result, 255
 - sl_si, 256
 - sl_start, 256
 - sl_targsize, 256
 - sl_ti, 256
 - sl_timeout_count, 256
 - sl_titab, 256
 - sl_Tnexus, 256
 - sl_xname, 256
- SCSI_LOW_SPLSCSI
 - scsi_low.h, 853
- scsi_low_start
 - scsi_low.c, 823
- scsi_low_start_bus
 - scsi_low_funcs, 243
- SCSI_LOW_START_FAIL
 - scsi_low.h, 853
- SCSI_LOW_START_NO_QTAG
 - scsi_low.c, 799
- SCSI_LOW_START_OK
 - scsi_low.h, 853
- SCSI_LOW_START_QTAG
 - scsi_low.c, 799
- scsi_low_start_up
 - scsi_low.c, 824
- SCSI_LOW_START_UP_CHECK
 - scsi_low.c, 799
- SCSI_LOW_STATICS
 - scsi_low.c, 799
- scsi_low_statics, 257
 - nexus_conflict, 257
 - nexus_disconnected, 257
 - nexus_fail, 257
 - nexus_reselected, 257
 - nexus_win, 257
- scsi_low_statusin
 - scsi_low.h, 865
- SCSI_LOW_SYNC
 - scsi_low.h, 853
- scsi_low_synch
 - scsi_low.c, 824
- scsi_low_tag_t
 - scsi_low.h, 858
- SCSI_LOW_TARG_FLAGS_ALL_VALID
 - scsi_low.h, 853
- SCSI_LOW_TARG_FLAGS_DISK_VALID
 - scsi_low.h, 853
- SCSI_LOW_TARG_FLAGS_QUIRKS_VALID
 - scsi_low.h, 854
- SCSI_LOW_TARG_FLAGS_USER_VALID
 - scsi_low.h, 854

- scsi_low_targ_init
 - scsi_low_funcs, 243
- scsi_low_test
 - scsi_low.c, 828
- scsi_low_test_abort
 - scsi_low.c, 824
- scsi_low_test_atten
 - scsi_low.c, 824
- scsi_low_test_cmdlnk
 - scsi_low.c, 825
- scsi_low_test_id
 - scsi_low.c, 828
- scsi_low_timeout
 - scsi_low.c, 825
 - scsi_low_funcs, 243
- SCSI_LOW_TIMEOUT_CH_ENGAGE
 - scsi_low.h, 854
- SCSI_LOW_TIMEOUT_CH_IO
 - scsi_low.h, 854
- SCSI_LOW_TIMEOUT_CH_RECOVER
 - scsi_low.h, 854
- scsi_low_timeout_check
 - scsi_low.c, 825
- SCSI_LOW_TIMEOUT_CHECK_INTERVAL
 - scsi_low.h, 854
- SCSI_LOW_TIMEOUT_HZ
 - scsi_low.h, 854
- SCSI_LOW_TIMEOUT_START
 - scsi_low.h, 854
- SCSI_LOW_TIMEOUT_STOP
 - scsi_low.h, 854
- scsi_low_translate_error_code
 - scsi_low.c, 826
- scsi_low_twiddle_wait
 - scsi_low.c, 826
- scsi_low_unit_ready_cmd
 - scsi_low.c, 826
- SCSI_LOW_UNKLNK
 - scsi_low.h, 855
- SCSI_LOW_UNKTAG
 - scsi_low.h, 855
- scsi_low_version_major
 - scsi_low.c, 828
- scsi_low_version_minor
 - scsi_low.c, 828
- SCSI_LOW_WAIT
 - scsi_low.h, 855
- scsi_low_wide
 - scsi_low.c, 827
- SCSI_LOW_WRITE
 - scsi_low.h, 855
- SCSI_MAX_CDBLEN
 - scsi_all.h, 623
- scsi_message.h
 - MSG_ABORT, 871
 - MSG_ABORT_TAG, 871
 - MSG_ABORT_TASK, 871
 - MSG_ABORT_TASK_SET, 871
 - MSG_ACA_TASK, 871
 - MSG_BUS_DEV_RESET, 872
 - MSG_CLEAR_ACA, 872
 - MSG_CLEAR_QUEUE, 872
 - MSG_CLEAR_TASK_SET, 872
 - MSG_CMDCOMPLETE, 872
 - MSG_DISCONNECT, 872
 - MSG_EXT_PPR, 872
 - MSG_EXT_PPR_DT_REQ, 872
 - MSG_EXT_PPR_HOLD_MCS, 872
 - MSG_EXT_PPR_IU_REQ, 872
 - MSG_EXT_PPR_LEN, 872
 - MSG_EXT_PPR_PCOMP_EN, 873
 - MSG_EXT_PPR_QAS_REQ, 873
 - MSG_EXT_PPR_RD_STRM, 873
 - MSG_EXT_PPR_RTI, 873
 - MSG_EXT_PPR_WR_FLOW, 873
 - MSG_EXT_SDTR, 873
 - MSG_EXT_SDTR_LEN, 873
 - MSG_EXT_WDTR, 873
 - MSG_EXT_WDTR_BUS_16_BIT, 873
 - MSG_EXT_WDTR_BUS_32_BIT, 873
 - MSG_EXT_WDTR_BUS_8_BIT, 874
 - MSG_EXT_WDTR_LEN, 874
 - MSG_EXTENDED, 874
 - MSG_HEAD_OF_Q_TAG, 874
 - MSG_HEAD_OF_QUEUE_TASK, 874
 - MSG_IDENTIFY, 874
 - MSG_IDENTIFY_DISCFLAG, 874
 - MSG_IDENTIFY_LUNMASK, 874
 - MSG_IDENTIFYFLAG, 874
 - MSG_IGN_WIDE_RESIDUE, 874
 - MSG_INIT_RECOVERY, 874
 - MSG_INITIATOR_DET_ERR, 875
 - MSG_ISIDENTIFY, 875
 - MSG_LINK_CMD_COMPLETE, 875
 - MSG_LINK_CMD_COMPLETEF, 875
 - MSG_LOGICAL_UNIT_RESET, 875
 - MSG_MESSAGE_REJECT, 875
 - MSG_NOOP, 875
 - MSG_ORDERED_Q_TAG, 875
 - MSG_ORDERED_TASK, 875
 - MSG_PARITY_ERROR, 875
 - MSG_QAS_REQUEST, 875
 - MSG_REL_RECOVERY, 876
 - MSG_RESTOREPOINTERS, 876
 - MSG_SAVEDATAPOINTER, 876
 - MSG_SIMPLE_Q_TAG, 876
 - MSG_SIMPLE_TASK, 876
 - MSG_TARGET_RESET, 876

- MSG_TASK_COMPLETE, 876
- MSG_TERM_IO_PROC, 876
- SCSI_MIN_DELAY
 - scsi_all.c, 592
- scsi_mode_blk_desc, 258
 - blklen, 258
 - density, 258
 - nblocks, 258
 - reserved, 258
- scsi_mode_block_descr, 259
 - block_len, 259
 - density_code, 259
 - num_blocks, 259
 - reserved, 259
- scsi_mode_hdr_10, 260
 - block_descr_len, 260
 - datalen, 260
 - dev_specific, 260
 - medium_type, 260
 - reserved, 260
- scsi_mode_hdr_6, 261
 - block_descr_len, 261
 - datalen, 261
 - dev_specific, 261
 - medium_type, 261
- scsi_mode_header_10, 262
 - blk_desc_len, 262
 - data_length, 262
 - dev_spec, 262
 - medium_type, 262
 - unused, 262
- scsi_mode_header_6, 263
 - blk_desc_len, 263
 - data_length, 263
 - dev_spec, 263
 - medium_type, 263
- scsi_mode_page_header, 264
 - page_code, 264
 - page_length, 264
- scsi_mode_select
 - scsi_all.c, 596
 - scsi_all.h, 649
- scsi_mode_select_10, 265
 - byte2, 265
 - control, 265
 - length, 265
 - opcode, 265
 - unused, 265
- scsi_mode_select_6, 266
 - byte2, 266
 - control, 266
 - length, 266
 - opcode, 266
 - unused, 266
- scsi_mode_select_len
 - scsi_all.c, 596
 - scsi_all.h, 649
- scsi_mode_sense
 - scsi_all.c, 596
 - scsi_all.h, 649
- scsi_mode_sense_10, 267
 - byte2, 267
 - control, 267
 - length, 267
 - opcode, 267
 - page, 267
 - unused, 267
- scsi_mode_sense_6, 268
 - byte2, 268
 - control, 268
 - length, 268
 - opcode, 268
 - page, 268
 - unused, 268
- scsi_mode_sense_data, 270
 - blk_desc, 270
 - cap, 270
 - ea, 270
 - header, 270
 - pages, 270
 - tg, 271
- scsi_mode_sense_len
 - scsi_all.c, 597
 - scsi_all.h, 650
- scsi_move_medium, 272
 - byte2, 272
 - control, 272
 - dst, 272
 - invert, 272
 - opcode, 272
 - reserved, 272
 - scsi_ch.c, 736
 - scsi_ch.h, 749
 - src, 272
 - tea, 273
- scsi_op_codes
 - scsi_all.c, 605
- scsi_op_desc
 - scsi_all.c, 597
 - scsi_all.h, 650
- scsi_op_quirk_entry, 274
 - inq_pat, 274
 - num_ops, 274
 - op_table, 274
- scsi_op_quirk_table
 - scsi_all.c, 605
- scsi_pass.c
 - PASS_CCB_BUFFER_IO, 879

- PASS_CCB_WAITING, 879
- PASS_FLAG_INVALID, 880
- PASS_FLAG_LOCKED, 880
- PASS_FLAG_OPEN, 880
- PASS_STATE_NORMAL, 880
- scsi_pass.c
 - __FBSDID, 880
 - ccb_bp, 879
 - ccb_type, 879
 - pass_ccb_types, 879
 - pass_cdevsw, 886
 - pass_flags, 880
 - pass_state, 880
 - passasync, 880
 - passcleanup, 881, 886
 - passclose, 881, 886
 - passdone, 881
 - passdriver, 887
 - passerror, 882
 - passinit, 883, 887
 - passioctl, 883, 887
 - passoninvalidate, 884, 887
 - passopen, 884, 887
 - passregister, 885, 887
 - passsendccb, 885
 - passstart, 886, 887
 - PERIPHDRIVER_DECLARE, 886
- scsi_pass.h
 - _SCSI_PASS_H, 888
 - CAMGETPASSTHRU, 888
 - CAMIOCOMMAND, 888
- scsi_pause, 275
 - byte2, 275
 - control, 275
 - op_code, 275
 - resume, 275
 - unused, 275
- scsi_play_10, 276
 - blk_addr, 276
 - byte2, 276
 - control, 276
 - op_code, 276
 - unused, 276
 - xfer_len, 276
- scsi_play_12, 277
 - blk_addr, 277
 - byte2, 277
 - control, 277
 - op_code, 277
 - unused, 277
 - xfer_len, 277
- scsi_play_msf, 278
 - byte2, 278
 - control, 278
 - end_f, 278
 - end_m, 278
 - end_s, 278
 - op_code, 278
 - start_f, 278
 - start_m, 279
 - start_s, 279
 - unused, 279
- scsi_play_rel_12, 280
 - blk_addr, 280
 - byte2, 280
 - control, 280
 - op_code, 280
 - track, 280
 - xfer_len, 280
- scsi_play_track, 281
 - byte2, 281
 - control, 281
 - end_index, 281
 - end_track, 281
 - op_code, 281
 - start_index, 281
 - start_track, 281
 - unused, 282
 - unused1, 282
- scsi_position_to_element, 283
 - byte2, 283
 - control, 283
 - dst, 283
 - invert, 283
 - opcode, 283
 - reserved, 283
 - scsi_ch.c, 736
 - scsi_ch.h, 750
 - tea, 283
- scsi_prevent, 285
 - byte2, 285
 - control, 285
 - how, 285
 - opcode, 285
 - scsi_all.c, 597
 - scsi_all.h, 650
 - unused, 285
- scsi_print_inquiry
 - scsi_all.c, 598
 - scsi_all.h, 651
- scsi_pt.c
 - PT_CCB_BUFFER_IO, 892
 - PT_CCB_BUFFER_IO_UA, 892
 - PT_CCB_RETRY_UA, 892
 - PT_CCB_WAITING, 892
 - PT_FLAG_DEVICE_INVALID, 892
 - PT_FLAG_NONE, 892
 - PT_FLAG_OPEN, 892

- PT_FLAG_RETRY_UA, 892
- PT_STATE_NORMAL, 892
- PT_STATE_PROBE, 892
- scsi_pt.c
 - __FBSDID, 892
 - ccb_bp, 891
 - ccb_state, 891
 - PERIPHDRIVER_DECLARE, 892
 - pt_ccb_state, 892
 - pt_cdevsw, 899
 - pt_flags, 892
 - pt_state, 892
 - ptasync, 892
 - ptclose, 893, 899
 - ptctor, 893, 899
 - ptdone, 894
 - ptdriver, 899
 - ptdtor, 894, 899
 - pterror, 895
 - ptinit, 895, 899
 - ptioctl, 896, 899
 - ptoninvalidate, 896, 899
 - ptopen, 897, 900
 - ptstart, 897, 900
 - ptstrategy, 898, 900
 - SCSI_PT_DEFAULT_TIMEOUT, 891
 - scsi_send_receive, 898
- scsi_pt.h
 - _SCSI_SCSI_PT_H, 901
 - RECEIVE, 901
 - SEND, 901
- SCSI_PT_DEFAULT_TIMEOUT
 - scsi_pt.c, 891
- SCSI_RCB
 - scsi_cd.h, 715
- scsi_read_block_limits, 286
 - byte2, 286
 - control, 286
 - opcode, 286
 - scsi_sa.c, 930
 - scsi_sa.h, 948
 - unused, 286
- scsi_read_block_limits_data, 287
 - gran, 287
 - maximum, 287
 - minimum, 287
- scsi_read_buffer, 288
 - buffer_id, 288
 - byte2, 288
 - control, 288
 - length, 288
 - offset, 288
 - opcode, 288
- scsi_read_capacity, 289
 - addr, 289
 - byte2, 289
 - control, 289
 - opcode, 289
 - scsi_all.c, 598
 - scsi_all.h, 651
 - unused, 289
- scsi_read_capacity_16, 290
 - addr, 290
 - alloc_len, 290
 - control, 290
 - opcode, 290
 - reladr, 290
 - scsi_all.c, 598
 - scsi_all.h, 651
 - service_action, 290
- scsi_read_capacity_data, 291
 - addr, 291
 - length, 291
- scsi_read_capacity_data_long, 292
 - addr, 292
 - length, 292
- scsi_read_cd_cap_data, 293
 - addr_0, 293
 - addr_1, 293
 - addr_2, 293
 - addr_3, 293
 - length_0, 293
 - length_1, 293
 - length_2, 293
 - length_3, 294
- scsi_read_cd_capacity, 295
 - addr_0, 295
 - addr_1, 295
 - addr_2, 295
 - addr_3, 295
 - byte2, 295
 - control, 295
 - op_code, 295
 - unused, 296
- scsi_read_defect_data_10, 297
 - alloc_length, 297
 - byte2, 297
 - control, 297
 - format, 297
 - opcode, 297
 - reserved, 297
- scsi_read_defect_data_12, 298
 - alloc_length, 298
 - byte2, 298
 - control, 298
 - format, 298
 - opcode, 298
 - reserved, 298

- scsi_read_defect_data_hdr_10, 299
 - format, 299
 - length, 299
 - reserved, 299
- scsi_read_defect_data_hdr_12, 300
 - format, 300
 - length, 300
 - reserved, 300
- scsi_read_dvd_struct_data_bca, 301
 - bca_info, 301
 - data_len, 301
 - reserved, 301
- scsi_read_dvd_struct_data_copy_manage, 302
 - byte4, 302
 - data_len, 302
 - reserved0, 302
 - reserved1, 302
- scsi_read_dvd_struct_data_copyright, 303
 - cps_type, 303
 - data_len, 303
 - region_info, 303
 - reserved0, 303
 - reserved1, 303
- scsi_read_dvd_struct_data_dcb, 304
 - data_len, 304
 - dcb, 304
 - reserved, 304
- scsi_read_dvd_struct_data_dds, 305
 - data_len, 305
 - dds_info, 305
 - reserved, 305
- scsi_read_dvd_struct_data_disc_id, 306
 - data_len, 306
 - day, 306
 - hour, 306
 - minute, 306
 - month, 306
 - random_num, 306
 - reserved, 306
 - second, 307
 - year, 307
- scsi_read_dvd_struct_data_disc_key, 308
 - data_len, 308
 - disc_key, 308
 - reserved, 308
- scsi_read_dvd_struct_data_disc_key_blk, 309
 - data_len, 309
 - disc_key_pack_data, 309
 - reserved, 309
 - total_packs, 309
- scsi_read_dvd_struct_data_generic_dcb, 310
 - content_desc, 310
 - dcb_data, 310
 - unknown_desc_actions, 310
- vendor_id, 310
- scsi_read_dvd_struct_data_header, 311
 - data_len, 311
 - reserved, 311
- scsi_read_dvd_struct_data_layer_desc, 312
 - bca, 312
 - book_type_version, 312
 - density, 312
 - disc_size_max_rate, 312
 - end_sector_layer0, 312
 - layer_info, 312
 - main_data_end, 313
 - main_data_start, 313
 - media_specific, 313
 - zeros0, 313
 - zeros1, 313
 - zeros2, 313
- scsi_read_dvd_struct_data_lead_in, 314
 - app_code, 314
 - data_len, 314
 - disc_physical_data, 314
 - field_id_1, 314
 - field_id_2, 315
 - field_id_3, 315
 - field_id_4, 315
 - field_id_5, 315
 - last_addr, 315
 - manuf_id_11_6, 315
 - manuf_id_17_12, 315
 - manuf_id_5_0, 315
 - optimum_write_strategy, 315
 - reserved0, 315
 - reserved1, 315
 - reserved2, 316
 - reserved3, 316
 - reserved4, 316
 - reserved5, 316
 - rwp, 316
 - rwp_wavelength, 316
- scsi_read_dvd_struct_data_manufacturer, 317
 - data_len, 317
 - manuf_info, 317
 - reserved, 317
- scsi_read_dvd_struct_data_medium_status, 318
 - byte4, 318
 - data_len, 318
 - disc_type_id, 318
 - ram_swi_info, 318
 - reserved0, 318
 - reserved1, 318
- scsi_read_dvd_struct_data_physical, 319
 - data_len, 319
 - layer_desc, 319
 - reserved, 319

- scsi_read_dvd_struct_data_prot_discid, 320
 - data_len, 320
 - prot_discid_data, 320
 - reserved, 320
- scsi_read_dvd_struct_data_rmd, 321
 - data_len, 321
 - last_sector_num, 321
 - reserved, 321
 - rmd_bytes, 321
- scsi_read_dvd_struct_data_rmd_borderout, 322
 - data_len, 322
 - reserved, 322
 - rmd, 322
- scsi_read_dvd_struct_data_spare_area, 323
 - allocated_supl, 323
 - data_len, 323
 - reserved, 323
 - unused_primary, 323
 - unused_supl, 323
- scsi_read_dvd_structure, 324
 - address, 324
 - agid, 324
 - alloc_len, 324
 - control, 324
 - format, 324
 - layer_number, 324
 - opcode, 324
 - reserved, 325
 - scsi_cd.c, 690
 - scsi_cd.h, 716
- scsi_read_element_status, 326
 - byte2, 326
 - control, 326
 - count, 326
 - len, 326
 - opcode, 326
 - reserved0, 326
 - reserved1, 326
 - scsi_ch.c, 737
 - scsi_ch.h, 750
 - sea, 327
- scsi_read_format_capacities, 328
 - alloc_length, 328
 - byte2, 328
 - opcode, 328
 - reserved0, 328
 - reserved1, 328
- scsi_read_header, 329
 - blk_addr, 329
 - byte2, 329
 - control, 329
 - data_len, 329
 - op_code, 329
 - unused, 329
- scsi_read_position
 - scsi_sa.c, 930
 - scsi_sa.h, 948
- scsi_read_subchannel, 330
 - byte1, 330
 - byte2, 330
 - control, 330
 - data_len, 330
 - op_code, 330
 - subchan_format, 330
 - track, 330
 - unused, 331
- scsi_read_toc, 332
 - byte2, 332
 - control, 332
 - data_len, 332
 - from_track, 332
 - op_code, 332
 - unused, 332
- scsi_read_write
 - scsi_all.c, 599
 - scsi_all.h, 652
- scsi_reassign_blocks, 333
 - byte2, 333
 - control, 333
 - opcode, 333
 - unused, 333
- scsi_reassign_blocks_data, 334
 - defect_descriptor, 334
 - dlbaddr, 334
 - length, 334
 - reserved, 334
- scsi_release, 335
 - byte2, 335
 - control, 335
 - length, 335
 - opcode, 335
 - unused, 335
- scsi_report_key, 336
 - agid_keyformat, 336
 - alloc_len, 336
 - control, 336
 - lba, 336
 - opcode, 336
 - reserved0, 336
 - reserved1, 336
 - scsi_cd.c, 690
 - scsi_cd.h, 716
- scsi_report_key_data_agid, 338
 - agid, 338
 - data_len, 338
 - reserved, 338
- scsi_report_key_data_asf, 339
 - data_len, 339

- reserved, 339
- success, 339
- scsi_report_key_data_challenge, 340
 - challenge_key, 340
 - data_len, 340
 - reserved0, 340
 - reserved1, 340
- scsi_report_key_data_header, 341
 - data_len, 341
 - reserved, 341
- scsi_report_key_data_key1_key2, 342
 - data_len, 342
 - key1, 342
 - reserved0, 342
 - reserved1, 342
- scsi_report_key_data_rpc, 343
 - byte4, 343
 - data_len, 343
 - region_mask, 343
 - reserved0, 343
 - reserved1, 343
 - rpc_scheme0, 343
 - rpc_scheme1, 343
- scsi_report_key_data_title, 345
 - byte0, 345
 - data_len, 345
 - reserved0, 345
 - reserved1, 345
 - title_key, 345
- scsi_report_luns, 346
 - control, 346
 - length, 346
 - opcode, 346
 - reserved1, 346
 - reserved2, 346
 - reserved3, 346
 - scsi_all.c, 599
 - scsi_all.h, 652
 - select_report, 346
- scsi_report_luns_data, 348
 - length, 348
 - lundata, 348
 - luns, 348
 - reserved, 348
- scsi_request_sense, 349
 - byte2, 349
 - control, 349
 - length, 349
 - opcode, 349
 - scsi_all.c, 600
 - scsi_all.h, 653
 - unused, 349
- scsi_request_volume_element_address, 350
 - byte2, 350
 - control, 350
 - count, 350
 - eaddr, 350
 - len, 350
 - opcode, 350
 - reserved0, 350
 - reserved1, 351
- scsi_reserve, 352
 - byte2, 352
 - control, 352
 - length, 352
 - opcode, 352
 - unused, 352
- scsi_reserve_release_unit, 353
 - control, 353
 - lun_thirdparty, 353
 - opcode, 353
 - reserved, 353
 - scsi_sa.c, 931
 - scsi_sa.h, 949
- scsi_rev
 - sa_softc, 204
- SCSI_REV_0
 - scsi_all.h, 624
- SCSI_REV_2
 - scsi_all.h, 624
- SCSI_REV_CCS
 - scsi_all.h, 624
- SCSI_REV_SPC
 - scsi_all.h, 624
- SCSI_REV_SPC2
 - scsi_all.h, 624
- scsi_rewind, 354
 - control, 354
 - immediate, 354
 - opcode, 354
 - reserved, 354
 - scsi_sa.c, 931
 - scsi_sa.h, 949
- scsi_rezero_unit, 355
 - byte2, 355
 - control, 355
 - opcode, 355
 - reserved, 355
- scsi_rw_10, 356
 - addr, 356
 - byte2, 356
 - control, 356
 - length, 356
 - opcode, 356
 - reserved, 356
- scsi_rw_12, 358
 - addr, 358
 - byte2, 358

- control, 358
- length, 358
- opcode, 358
- reserved, 358
- scsi_rw_16, 360
 - addr, 360
 - byte2, 360
 - control, 360
 - length, 360
 - opcode, 360
 - reserved, 360
- scsi_rw_6, 362
 - addr, 362
 - control, 362
 - length, 362
 - opcode, 362
- scsi_sa.c
 - SA_FLAG_COMP_ENABLED, 911
 - SA_FLAG_COMP_SUPP, 911
 - SA_FLAG_COMP_UNSUPP, 911
 - SA_FLAG_EIO_PENDING, 911
 - SA_FLAG_EOF_PENDING, 911
 - SA_FLAG_EOM_PENDING, 911
 - SA_FLAG_ERR_PENDING, 911
 - SA_FLAG_FIXED, 911
 - SA_FLAG_INVALID, 911
 - SA_FLAG_OPEN, 911
 - SA_FLAG_TAPE_FROZEN, 911
 - SA_FLAG_TAPE_LOCKED, 911
 - SA_FLAG_TAPE_MOUNTED, 911
 - SA_FLAG_TAPE_WP, 911
 - SA_FLAG_TAPE_WRITTEN, 911
 - SA_MODE_NOREWIND, 912
 - SA_MODE_OFFLINE, 912
 - SA_MODE_REWIND, 912
 - SA_PARAM_ALL, 912
 - SA_PARAM_BLOCKSIZE, 912
 - SA_PARAM_BUFF_MODE, 912
 - SA_PARAM_COMPRESSION, 912
 - SA_PARAM_DENSITY, 912
 - SA_PARAM_NONE, 912
 - SA_PARAM_NUMBLOCKS, 912
 - SA_PARAM_SPEED, 912
 - SA_PARAM_WP, 912
 - SA_QUIRK_1FM, 912
 - SA_QUIRK_2FM, 912
 - SA_QUIRK_FIXED, 912
 - SA_QUIRK_NO_CPAGE, 912
 - SA_QUIRK_NO_MODESEL, 912
 - SA_QUIRK_NOCOMP, 912
 - SA_QUIRK_NODREAD, 912
 - SA_QUIRK_NONE, 912
 - SA_QUIRK_VARIABLE, 912
 - SA_STATE_ABNORMAL, 912
 - SA_STATE_NORMAL, 912
- scsi_sa.c
 - __FBSDID, 913
 - ccb_bp, 906
 - ccb_pflags, 906
 - CCB_Type, 907
 - D_TAPE, 907
 - ERASE_TIMEOUT, 907
 - IO_TIMEOUT, 907
 - last_ctl_cdb, 907
 - last_ctl_resid, 907
 - last_ctl_sense, 907
 - last_io_cdb, 907
 - last_io_resid, 907
 - last_io_sense, 907
 - MALLOC_DEFINE, 913
 - PENDING_MOUNT_CHECK, 908
 - PERIPHDRIVER_DECLARE, 913
 - QFRLS, 908
 - REWIND_TIMEOUT, 908
 - SA_ATYPE_ER, 908
 - SA_ATYPE_NR, 908
 - SA_ATYPE_R, 908
 - SA_CCB_BUFFER_IO, 909
 - SA_CCB_TYPEMASK, 909
 - SA_CCB_WAITING, 909
 - sa_cdevsw, 933
 - SA_CTLDEV, 909
 - SA_ERASE_TIMEOUT, 909
 - sa_flags, 911
 - SA_IO_TIMEOUT, 909
 - SA_IS_CTRL, 909
 - sa_mode, 911
 - SA_NOT_CTLDEV, 909
 - SA_NUM_MODES, 909
 - sa_params, 912
 - SA_POSITION_UPDATED, 910
 - sa_quirk_table, 933
 - sa_quirks, 912
 - SA_REWIND_TIMEOUT, 910
 - SA_SPACE_TIMEOUT, 910
 - sa_state, 912
 - saasync, 913
 - sachecked, 913
 - sacleanup, 914, 933
 - saclose, 914, 933
 - SADENSITY, 910
 - sadone, 915
 - sadriver, 933
 - saerase, 915
 - saerror, 916
 - sagetparams, 917
 - sainit, 918, 933
 - saiocctl, 918, 933

- saloadunload, 920
- samarkswanted, 920
- SAMINOR, 910
- SAMODE, 910
- samount, 920
- saoninvalidate, 921, 934
- saopen, 922, 934
- saprevent, 922
- sardpos, 923
- saregister, 923, 934
- sareservereleaseunit, 924
- saretension, 924
- sarewind, 925
- sasetparams, 925
- sasetpos, 926
- saspace, 927
- sastart, 928, 934
- sastrategy, 928, 934
- SAUNIT, 910
- sawritefilemarks, 929
- scsi_erase, 929
- scsi_load_unload, 930
- scsi_read_block_limits, 930
- scsi_read_position, 930
- scsi_reserve_release_unit, 931
- scsi_rewind, 931
- scsi_sa_read_write, 931
- scsi_set_position, 932
- scsi_space, 932
- scsi_write_filemarks, 932
- SCSIOP_TIMEOUT, 910
- Set_CCB_Type, 911
- SPACE_TIMEOUT, 911
- UNUSED_PARAMETER, 911
- scsi_sa.h
 - SS_BLOCKS, 947
 - SS_EOD, 947
 - SS_FILEMARKS, 947
 - SS_SEQFILEMARKS, 947
 - SS_SEQSETMARKS, 947
 - SS_SETMARKS, 947
- scsi_sa.h
 - _SCSI SCSI_SA_H, 938
 - ERASE, 938
 - LOAD_UNLOAD, 938
 - LOCATE, 938
 - RBL_GRAN, 938
 - RBL_GRAN_MASK, 938
 - READ_BLOCK_LIMITS, 938
 - READ_POSITION, 939
 - RELEASE_UNIT, 939
 - RESERVE_UNIT, 939
 - REWIND, 939
 - SA_ASOC_WP, 939
 - SA_AVC, 939
 - SA_BIS, 939
 - SA_COMP_DEFAULT, 939
 - SA_COMP_NONE, 939
 - SA_DATA_COMPRESSION_PAGE, 939
 - SA_DBR, 940
 - SA_DCP_DCC, 940
 - SA_DCP_DCE, 940
 - SA_DCP_DDE, 940
 - SA_DCP_RED_0, 940
 - SA_DCP_RED_1, 940
 - SA_DCP_RED_2, 940
 - SA_DCP_RED_MASK, 940
 - SA_DCP_RED_SHAMT, 940
 - SA_DEVICE_CONFIGURATION_PAGE, 940
 - SA_MEDIUM_PARTITION_PAGE_1, 941
 - SA_MEDIUM_PARTITION_PAGE_2, 941
 - SA_MEDIUM_PARTITION_PAGE_3, 941
 - SA_MEDIUM_PARTITION_PAGE_4, 941
 - SA_PERM_WP, 941
 - SA_PERS_WP, 941
 - SA_RBO, 941
 - SA_READ, 941
 - SA_REW, 941
 - SA_RPOS_BCU, 941
 - SA_RPOS_BOP, 941
 - SA_RPOS_BPU, 942
 - SA_RPOS_BYCU, 942
 - SA_RPOS_EOP, 942
 - SA_RPOS_PERR, 942
 - SA_RPOS_UNCERTAIN, 942
 - SA_RSMK, 942
 - SA_SOCF_MASK, 942
 - SA_SPOS_BT, 942
 - SA_SPOS_CP, 942
 - SA_SPOS_IMMED, 942
 - SA_WRITE, 942
 - SAR_SLI, 943
 - SARW_FIXED, 943
 - scsi_data_comp_page, 948
 - SCSI_DENSITY_HALFINCH_1600, 943
 - SCSI_DENSITY_HALFINCH_6250, 943
 - SCSI_DENSITY_HALFINCH_6250C, 943
 - SCSI_DENSITY_HALFINCH_800, 943
 - SCSI_DENSITY_HALFINCH_PE, 943
 - SCSI_DENSITY_QIC_11_4TRK, 943
 - SCSI_DENSITY_QIC_11_9TRK, 943
 - SCSI_DENSITY_QIC_120, 944
 - SCSI_DENSITY_QIC_1320, 944
 - SCSI_DENSITY_QIC_150, 944
 - SCSI_DENSITY_QIC_24, 944
 - SCSI_DENSITY_QIC_2GB, 944
 - SCSI_DENSITY_QIC_3080, 944

- SCSI_DENSITY_QIC_4GB, 944
- SCSI_DENSITY_QIC_525_320, 944
- scsi_erase, 948
- scsi_load_unload, 948
- scsi_read_block_limits, 948
- scsi_read_position, 948
- scsi_reserve_release_unit, 949
- scsi_rewind, 949
- scsi_sa_read_write, 949
- scsi_set_position, 950
- scsi_space, 950
- scsi_space_code, 947
- scsi_write_filemarks, 951
- SE_IMMED, 944
- SE_LONG, 945
- SE_LUN_MASK, 945
- SLU_EOT, 945
- SLU_IMMED, 945
- SLU_LOAD, 945
- SLU_RETEN, 945
- SMH_SA_BUF_MODE_MASK, 945
- SMH_SA_BUF_MODE_MIBUF, 945
- SMH_SA_BUF_MODE_NOBUF, 945
- SMH_SA_BUF_MODE_SIBUF, 946
- SMH_SA_SPEED_DEFAULT, 946
- SMH_SA_SPEED_MASK, 946
- SMH_SA_WP, 946
- SPACE, 946
- SREW_IMMED, 946
- SRRU_3RD_MASK, 946
- SRRU_3RD_PARTY, 946
- SRRU_3RD_SHAMT, 947
- SRRU_LUN_MASK, 947
- SWFMRK_IMMED, 947
- SWFMRK_WSMK, 947
- WRITE_FILEMARKS, 947
- scsi_sa_read_write
 - scsi_sa.c, 931
 - scsi_sa.h, 949
- scsi_sa_rw, 363
 - control, 363
 - length, 363
 - opcode, 363
 - sli_fixed, 363
- SCSI_SAME_DENSITY
 - scsi_all.h, 624
- scsi_send_diag, 364
 - byte2, 364
 - control, 364
 - opcode, 364
 - paramlen, 364
 - unused, 364
- scsi_send_key, 365
 - agid_keyformat, 365
 - control, 365
 - opcode, 365
 - param_len, 365
 - reserved, 365
 - scsi_cd.c, 691
 - scsi_cd.h, 717
- scsi_send_key_data_rpc, 366
 - data_len, 366
 - region_code, 366
 - reserved0, 366
 - reserved1, 366
- scsi_send_receive, 367
 - byte2, 367
 - control, 367
 - opcode, 367
 - scsi_pt.c, 898
 - xfer_len, 367
- scsi_send_volume_tag, 368
 - byte2, 368
 - control, 368
 - ea, 368
 - opcode, 368
 - pll, 368
 - reserved2, 368
 - reserved4, 368
 - reserved5, 369
 - sac, 369
 - scsi_ch.c, 737
 - scsi_ch.h, 751
- scsi_send_volume_tag_parameters, 370
 - maxvsn, 370
 - minvsn, 370
 - reserved1, 370
 - reserved2, 370
 - vitf, 370
- scsi_sense, 371
 - byte2, 371
 - control, 371
 - length, 371
 - opcode, 371
 - unused, 371
- scsi_sense_action
 - scsi_all.h, 644
- scsi_sense_action_qualifier
 - scsi_all.h, 644
- scsi_sense_data, 372
 - add_sense_code, 372
 - add_sense_code_qual, 372
 - cmd_spec_info, 372
 - error_code, 372
 - extra_bytes, 372
 - extra_len, 373
 - flags, 373
 - fru, 373

- info, 373
- segment, 373
- sense_key_spec, 373
- scsi_sense_desc
 - scsi_all.c, 600
 - scsi_all.h, 653
- scsi_sense_key_text
 - scsi_all.h, 657
- scsi_sense_print
 - scsi_all.c, 600
 - scsi_all.h, 653
- scsi_sense_quirk_entry, 374
 - asc_info, 374
 - inq_pat, 374
 - num_asc, 374
 - num_sense_keys, 374
 - sense_key_info, 374
- scsi_sense_sbuf
 - scsi_all.c, 601
 - scsi_all.h, 654
- scsi_sense_string
 - scsi_all.c, 601
 - scsi_all.h, 654
- scsi_sense_string_flags
 - scsi_all.h, 644
- scsi_ses.c
 - SES_NONE, 966
 - SES_SAFTE, 966
 - SES_SEN, 966
 - SES_SES, 966
 - SES_SES_PASSTHROUGH, 966
 - SES_SES_SCSI2, 966
 - SesArrayControl, 966
 - SesConfigPage, 966
 - SesControlPage, 966
 - SesElementDescriptor, 966
 - SesHelpTxt, 966
 - SesShortStatus, 966
 - SesStringOut, 966
 - SesThresholdOut, 966
- scsi_ses.c
 - __FBSDDID, 966
 - ALL_ENC_STAT, 957
 - ccb_bp, 957
 - ccb_state, 957
 - CFLEN, 957
 - ENCI_SVALID, 957
 - enctyp, 966
 - gbit, 957
 - gbyte, 957
 - gget16, 957
 - gget24, 957
 - gget32, 958
 - gget8, 958
 - MALLOC_DEFINE, 966
 - MEMCPY, 958
 - MEMZERO, 958
 - NPSEUDO_ALARM, 958
 - NPSEUDO_THERM, 958
 - perf_slotop, 966
 - PERIPHDRIVER_DECLARE, 967
 - PRINTF, 958
 - SAFT_ALARM_OFFSET, 959
 - SAFT_BAIL, 959
 - SAFT_FLG1_ALARM, 959
 - SAFT_FLG1_ENCDRVFAIL, 959
 - SAFT_FLG1_ENCDRVWARN, 959
 - SAFT_FLG1_ENCFANFAIL, 959
 - SAFT_FLG1_ENCPWRFAIL, 959
 - SAFT_FLG1_ENCPWROFF, 959
 - SAFT_FLG1_GLOBFAIL, 960
 - SAFT_FLG1_GLOBWARN, 960
 - SAFT_FLG2_LOCKDOOR, 960
 - SAFT_PRIVATE, 960
 - SAFT_SCRATCH, 960
 - saft_2little, 983
 - SAFTE_END, 960
 - saft_get_encstat, 967
 - saft_get_objstat, 967
 - saft_getconfig, 968
 - saft_init_enc, 968
 - SAFTE_LEN, 960
 - SAFTE_RD_RDCFG, 960
 - SAFTE_RD_RDDSTS, 960
 - SAFTE_RD_RDESTS, 961
 - saft_rdstat, 969
 - saft_set_encstat, 969
 - saft_set_objstat, 970
 - saft_softc_init, 970
 - SAFTE_START, 961
 - SAFTE_WT_ACTPWS, 961
 - SAFTE_WT_DSTAT, 961
 - SAFTE_WT_FANSPD, 961
 - SAFTE_WT_GLOBAL, 961
 - SAFTE_WT_SLTOP, 961
 - sbit, 961
 - sbyte, 961
 - SCSZ, 962
 - SEN_ID, 962
 - SEN_ID_LEN, 962
 - ses_cdevsw, 983
 - ses_cfghdr, 971
 - SES_CFGHDR_MINLEN, 962
 - SES_CFLAGS, 962
 - ses_decode, 971
 - SES_DLOG, 962
 - ses_endedesc, 971
 - ses_enchdr, 971

- SES_ENCHDR_MINLEN, 962
- SES_ENCHDR_VMIN, 962
- ses_encode, 971
- SES_FLAG_INITIALIZED, 962
- SES_FLAG_INVALID, 963
- SES_FLAG_OPEN, 963
- SES_FLAGS, 963
- SES_FREE, 963
- ses_get_encstat, 971
- ses_get_objstat, 972
- ses_getconfig, 972
- ses_getputstat, 973
- ses_getthdr, 973
- ses_init_enc, 974
- SES_LOG, 963
- ses_log, 974
- SES_MALLOC, 963
- ses_runcmd, 974
- ses_set_encstat, 974
- ses_set_objstat, 975
- ses_softc_init, 975
- ses_softc_t, 965
- ses_type, 976
- SES_VLOG, 963
- SesArrayStatus, 963
- sesasync, 976
- sescleanup, 977, 983
- sesclose, 977, 983
- SesDiagPageCodes, 966
- sesdone, 977
- sesdriver, 983
- seserror, 978
- sesinit, 978, 984
- sesioctl, 979, 984
- sesoninvalidate, 979, 984
- sesopen, 980, 984
- sesregister, 980, 984
- sesstart, 981, 984
- SesStatusPage, 964
- SesStringIn, 964
- SesThresholdIn, 964
- SESUNIT, 964
- set_objstat_sel, 981
- sget16, 964
- sget24, 964
- sget32, 964
- sget8, 964
- sset16, 964
- sset24, 965
- sset32, 965
- sset8, 965
- STRNCMP, 965
- wrbuf16, 982
- wrslot_stat, 982
- scsi_ses.h
 - ses_encstat, 992
 - SES_ENCSTAT_CRITICAL, 986
 - SES_ENCSTAT_INFO, 986
 - SES_ENCSTAT_NONCRITICAL, 986
 - SES_ENCSTAT_UNRECOV, 986
 - SES_OBJSTAT_CRIT, 986
 - SES_OBJSTAT_NONCRIT, 987
 - SES_OBJSTAT_NOTAVAIL, 987
 - SES_OBJSTAT_NOTINSTALLED, 987
 - SES_OBJSTAT_OK, 987
 - SES_OBJSTAT_UNKNOWN, 987
 - SES_OBJSTAT_UNRECOV, 987
 - SES_OBJSTAT_UNSUPPORTED, 987
 - SESCTL_CSEL, 987
 - SESCTL_DEVOFF, 987
 - SESCTL_DISABLE, 987
 - SESCTL_DRVLCK, 988
 - SESCTL_PRDFAIL, 988
 - SESCTL_RQSFLT, 988
 - SESCTL_RQSID, 988
 - SESCTL_RQSINS, 988
 - SESCTL_RQSRMV, 988
 - SESCTL_RQSTFAIL, 988
 - SESCTL_RQSTON, 988
 - SESCTL_RSTSWAP, 988
 - SESIOC, 989
 - SESIOC_GETENCSTAT, 989
 - SESIOC_GETNOBJ, 989
 - SESIOC_GETOBJMAP, 989
 - SESIOC_GETOBJSTAT, 989
 - SESIOC_GETTEXT, 989
 - SESIOC_INIT, 989
 - SESIOC_SETENCSTAT, 989
 - SESIOC_SETOBJSTAT, 989
 - SESTYP_ALARM, 990
 - SESTYP_AMMETER, 990
 - SESTYP_COMPORT, 990
 - SESTYP_DEVICE, 990
 - SESTYP_DISPLAY, 990
 - SESTYP_DOORLOCK, 990
 - SESTYP_ESCC, 990
 - SESTYP_FAN, 990
 - SESTYP_KEYPAD, 990
 - SESTYP_LANGUAGE, 990
 - SESTYP_NVRAM, 991
 - SESTYP_POWER, 991
 - SESTYP_SCC, 991
 - SESTYP_SCSI_INI, 991
 - SESTYP_SCSI_TGT, 991
 - SESTYP_SCSIXVR, 991
 - SESTYP_SUBENC, 991
 - SESTYP_THERM, 991
 - SESTYP_UNSPECIFIED, 991

- SESTYP_UPS, 991
- SESTYP_VOM, 991
- scsi_set_position
 - scsi_sa.c, 932
 - scsi_sa.h, 950
- scsi_set_speed, 376
 - byte2, 376
 - control, 376
 - opcode, 376
 - readspeed, 376
 - reserved, 376
 - writespeed, 376
- scsi_space, 377
 - code, 377
 - control, 377
 - count, 377
 - opcode, 377
 - scsi_sa.c, 932
 - scsi_sa.h, 950
- scsi_space_code
 - scsi_sa.h, 947
- scsi_start_stop
 - scsi_all.c, 602
 - scsi_all.h, 655
- scsi_start_stop_unit, 378
 - byte2, 378
 - control, 378
 - how, 378
 - opcode, 378
 - reserved, 378
- scsi_static_inquiry_match
 - scsi_all.c, 602
 - scsi_all.h, 655
- scsi_static_inquiry_pattern, 379
 - media_type, 379
 - product, 379
 - revision, 379
 - type, 379
 - vendor, 379
- scsi_status
 - ccb_scsiio, 101
- SCSI_STATUS_ACA_ACTIVE
 - scsi_all.h, 624
- SCSI_STATUS_BUSY
 - scsi_all.h, 624
- SCSI_STATUS_CHECK_COND
 - scsi_all.h, 624
- SCSI_STATUS_CMD_TERMINATED
 - scsi_all.h, 625
- SCSI_STATUS_COND_MET
 - scsi_all.h, 625
- SCSI_STATUS_INTERMED
 - scsi_all.h, 625
- SCSI_STATUS_INTERMED_COND_MET
 - scsi_all.h, 625
- scsi_status_iu_header, 380
 - flags, 380
 - pkt_failures, 380
 - pkt_failures_length, 380
 - reserved, 380
 - sense_length, 380
 - status, 380
- SCSI_STATUS_OK
 - scsi_all.h, 625
- SCSI_STATUS_QUEUE_FULL
 - scsi_all.h, 625
- SCSI_STATUS_RESERV_CONFLICT
 - scsi_all.h, 625
- scsi_status_string
 - scsi_all.c, 602
 - scsi_all.h, 655
- SCSI_STATUS_TASK_ABORTED
 - scsi_all.h, 625
- scsi_sync_cache, 381
 - begin_lba, 381
 - byte2, 381
 - control, 381
 - lb_count, 381
 - opcode, 381
 - reserved, 381
- scsi_synchronize_cache
 - scsi_all.c, 603
 - scsi_all.h, 656
- scsi_synchrates
 - scsi_all.c, 605
- scsi_tape_locate, 382
 - blkaddr, 382
 - byte1, 382
 - control, 382
 - opcode, 382
 - partition, 382
 - reserved1, 382
 - reserved2, 382
- scsi_tape_position_data, 384
 - firstblk, 384
 - flags, 384
 - lastblk, 384
 - nbuflk, 384
 - nbuflkbyte, 384
 - partition, 384
 - reserved, 384
 - reserved2, 385
- scsi_tape_read_position, 386
 - byte1, 386
 - opcode, 386
 - reserved, 386
- scsi_targ_bh.c
 - TARGBH_CCB_WAITING, 996

- TARGBH_CCB_WORKQ, 996
- TARGBH_FLAG_LUN_ENABLED, 996
- TARGBH_FLAG_NONE, 996
- TARGBH_STATE_EXCEPTION, 996
- TARGBH_STATE_NORMAL, 996
- TARGBH_STATE_TEARDOWN, 996
- scsi_targ_bh.c
 - __FBSDID, 997
 - ccb_atio, 995
 - ccb_descr, 995
 - ccb_type, 995
 - MALLOC_DEFINE, 997
 - MAX_ACCEPT, 996
 - MAX_BUF_SIZE, 996
 - MAX_IMMEDIATE, 996
 - no_lun_inq_data, 1002
 - no_lun_sense_data, 1002
 - PERIPHDRIIVER_DECLARE, 997
 - request_sense_size, 1002
 - TAILQ_HEAD, 997
 - targbh_ccb_types, 996
 - targbh_flags, 996
 - targbh_state, 996
 - targbhallocdescr, 997
 - targbhasync, 997
 - targbhctor, 997, 1002
 - targbhdislun, 998
 - targbhdone, 998
 - targbhdriver, 1002
 - targbhdtor, 999, 1002
 - targbhenlun, 999
 - targbhfreedescr, 1000
 - targbhinit, 1000, 1003
 - targbhstart, 1001, 1003
- scsi_target.c
 - TARG_STATE_LUN_ENABLED, 1007
 - TARG_STATE_OPENED, 1007
 - TARG_STATE_RESV, 1007
- scsi_target.c
 - __FBSDID, 1007
 - abort_all_pending, 1007
 - MALLOC_DEFINE, 1007
 - notify_user, 1007
 - PERIPHDRIIVER_DECLARE, 1008
 - TAILQ_HEAD, 1008
 - targ_cdevsw, 1016
 - targ_descr, 1007
 - targ_state, 1007
 - targasync, 1008
 - targcamstatus, 1008
 - targccbten, 1008
 - targclone, 1008
 - targclose, 1008, 1017
 - targctor, 1009, 1017
 - targdisable, 1009
 - targdone, 1009
 - targdriver, 1017
 - targdtor, 1010, 1017
 - targenable, 1010
 - targendislun, 1011
 - targfreeccb, 1011
 - targgetccb, 1012
 - targgetdescr, 1012
 - targinit, 1012, 1017
 - targioctl, 1012, 1017
 - targkqfilter, 1013, 1017
 - targopen, 1013, 1017
 - targpoll, 1013, 1017
 - targread, 1014, 1018
 - targread_filtops, 1018
 - targreadfilt, 1014
 - targreadfiltdetach, 1014
 - targreturnccb, 1014
 - targsendccb, 1015
 - targstart, 1015, 1018
 - targusermerge, 1015
 - targwrite, 1016, 1018
- scsi_targetio.h
 - TAILQ_HEAD, 1020
 - TARGIOCDEBUG, 1019
 - TARGIOCDISABLE, 1019
 - TARGIOCENABLE, 1020
- scsi_test_unit_ready, 387
 - byte2, 387
 - control, 387
 - opcode, 387
 - scsi_all.c, 603
 - scsi_all.h, 656
 - unused, 387
- scsi_u64to8b
 - scsi_all.h, 656
- scsi_ulto2b
 - scsi_all.h, 656
- scsi_ulto3b
 - scsi_all.h, 657
- scsi_ulto4b
 - scsi_all.h, 657
- scsi_verify, 388
 - addr, 388
 - byte2, 388
 - len, 388
 - opcode, 388
 - reserved0, 388
 - reserved1, 388
- scsi_vpd_unit_serial_number, 389
 - device, 389
 - length, 389
 - page_code, 389

- reserved, 389
- serial_num, 389
- scsi_write_and_verify, 390
 - addr, 390
 - byte2, 390
 - len, 390
 - opcode, 390
 - reserved0, 390
 - reserved1, 390
- scsi_write_buffer, 391
 - buffer_id, 391
 - byte2, 391
 - control, 391
 - length, 391
 - offset, 391
 - opcode, 391
- scsi_write_filemarks, 392
 - byte2, 392
 - control, 392
 - num_marks, 392
 - opcode, 392
 - scsi_sa.c, 932
 - scsi_sa.h, 951
- SCSIOP_TIMEOUT
 - scsi_sa.c, 910
- SCSZ
 - scsi_ses.c, 962
- sd_len
 - lun_info::scsi_low_inq_data, 168
- sd_resp
 - lun_info::scsi_low_inq_data, 168
- sd_sp1
 - lun_info::scsi_low_inq_data, 168
- sd_sp2
 - lun_info::scsi_low_inq_data, 168
- sd_support
 - lun_info::scsi_low_inq_data, 168
- sd_type
 - lun_info::scsi_low_inq_data, 168
- sd_version
 - lun_info::scsi_low_inq_data, 168
- sds_rds
 - read_dvd_struct_list_entry, 188
- sdst
 - scsi_exchange_medium, 223
- SE_IMMED
 - scsi_sa.h, 944
- SE_LONG
 - scsi_sa.h, 945
- SE_LUN_MASK
 - scsi_sa.h, 945
- sea
 - scsi_read_element_status, 327
- seagate
 - cam_xpt.c, 568
- sec_per_track
 - disk_pages::flexible_disk_page, 148
- second
 - scsi_read_dvd_struct_data_disc_id, 307
- secs_per_track
 - ccb_calc_geometry, 56
 - disk_params, 156
- secsize
 - disk_params, 156
- sector
 - scsi_defect_desc_phys_sector, 219
- sectors
 - disk_params, 156
- segment
 - scsi_sense_data, 373
- sel_comp_alg
 - scsi_dev_conf_page, 221
- select_report
 - scsi_report_luns, 346
- SELTIMEOUTIO
 - scsi_low.h, 855
- SEN_ID
 - scsi_ses.c, 962
- SEN_ID_LEN
 - scsi_ses.c, 962
- SEND
 - scsi_pt.h, 901
- send_active
 - cam_devq, 27
- SEND_DIAGNOSTIC
 - scsi_all.h, 626
- SEND_KEY
 - scsi_cd.h, 716
- send_openings
 - cam_devq, 27
- send_queue
 - cam_devq, 27
- SEND_VOLUME_TAG
 - scsi_ch.h, 748
- SEND_VOLUME_TAG_ASSERT_ALTERNATE
 - scsi_ch.h, 748
- SEND_VOLUME_TAG_ASSERT_PRIMARY
 - scsi_ch.h, 748
- SEND_VOLUME_TAG_REPLACE_-ALTERNATE
 - scsi_ch.h, 748
- SEND_VOLUME_TAG_REPLACE_PRIMARY
 - scsi_ch.h, 748
- SEND_VOLUME_TAG_UNDEFINED_-ALTERNATE
 - scsi_ch.h, 748
- SEND_VOLUME_TAG_UNDEFINED_-PRIMARY

- scsi_ch.h, 748
- sense_buf
 - sense_t, 394
- sense_code
 - read_element_status_descriptor, 191
- sense_data
 - ccb_accept_tio, 54
 - ccb_immed_notify, 80
 - ccb_scsiio, 101
- sense_key
 - sense_key_table_entry, 393
- sense_key_info
 - scsi_sense_quirk_entry, 374
- sense_key_spec
 - scsi_sense_data, 373
- sense_key_table
 - scsi_all.c, 605
- sense_key_table_entry, 393
 - action, 393
 - desc, 393
 - sense_key, 393
- sense_key_table_size
 - scsi_all.c, 606
- sense_len
 - ccb_accept_tio, 54
 - ccb_immed_notify, 81
 - ccb_scsiio, 101
- sense_length
 - scsi_status_iu_header, 380
- sense_ptr
 - sense_t, 394
- sense_qual
 - read_element_status_descriptor, 191
- sense_quirk_table
 - scsi_all.c, 606
- sense_quirk_table_size
 - scsi_all.c, 606
- sense_resid
 - ccb_scsiio, 101
- sense_t, 394
 - sense_buf, 394
 - sense_ptr, 394
- senseentrycomp
 - scsi_all.c, 603
- SENSEERR
 - scsi_low.h, 855
- SENSEIO
 - scsi_low.h, 855
- seq_id
 - ccb_notify_ack, 82
- serial_num
 - ccb_getdev, 71
 - scsi_vpd_unit_serial_number, 389
- serial_num_len
 - ccb_getdev, 71
- service_action
 - scsi_read_capacity_16, 290
- SERVICE_ACTION_IN
 - scsi_all.h, 626
- ses_cdevsw
 - scsi_ses.c, 983
- ses_cfghdr
 - scsi_ses.c, 971
- SES_CFGHDR_MINLEN
 - scsi_ses.c, 962
- SES_CFLAGS
 - scsi_ses.c, 962
- ses_decode
 - scsi_ses.c, 971
- ses_dev
 - ses_softc, 398
- SES_DLOG
 - scsi_ses.c, 962
- ses_eltmap
 - sscfcg, 409
- ses_encdesc
 - scsi_ses.c, 971
- ses_enchdr
 - scsi_ses.c, 971
- SES_ENCHDR_MINLEN
 - scsi_ses.c, 962
- SES_ENCHDR_VMIN
 - scsi_ses.c, 962
- ses_encode
 - scsi_ses.c, 971
- ses_encstat
 - scsi_ses.h, 992
 - ses_softc, 398
- SES_ENCSTAT_CRITICAL
 - scsi_ses.h, 986
- SES_ENCSTAT_INFO
 - scsi_ses.h, 986
- SES_ENCSTAT_NONCRITICAL
 - scsi_ses.h, 986
- SES_ENCSTAT_UNRECOV
 - scsi_ses.h, 986
- SES_FLAG_INITIALIZED
 - scsi_ses.c, 962
- SES_FLAG_INVALID
 - scsi_ses.c, 963
- SES_FLAG_OPEN
 - scsi_ses.c, 963
- SES_FLAGS
 - scsi_ses.c, 963
- ses_flags
 - ses_softc, 398
- SES_FREE
 - scsi_ses.c, 963

- ses_get_encstat
 - scsi_ses.c, 971
- ses_get_objstat
 - scsi_ses.c, 972
- ses_getconfig
 - scsi_ses.c, 972
- ses_getputstat
 - scsi_ses.c, 973
- ses_getthdr
 - scsi_ses.c, 973
- ses_hlptxt, 395
 - obj_id, 395
 - obj_text, 395
- ses_init_enc
 - scsi_ses.c, 974
- SES_LOG
 - scsi_ses.c, 963
- ses_log
 - scsi_ses.c, 974
- SES_MALLOC
 - scsi_ses.c, 963
- ses_nobjects
 - ses_softc, 399
- SES_NONE
 - scsi_ses.c, 966
- ses_ntypes
 - sscfg, 409
- ses_object, 396
 - obj_id, 396
 - object_type, 396
 - subencid, 396
- ses_objmap
 - ses_softc, 399
- ses_objstat, 397
 - cstat, 397
 - obj_id, 397
- SES_OBJSTAT_CRIT
 - scsi_ses.h, 986
- SES_OBJSTAT_NONCRIT
 - scsi_ses.h, 987
- SES_OBJSTAT_NOTAVAIL
 - scsi_ses.h, 987
- SES_OBJSTAT_NOTINSTALLED
 - scsi_ses.h, 987
- SES_OBJSTAT_OK
 - scsi_ses.h, 987
- SES_OBJSTAT_UNKNOWN
 - scsi_ses.h, 987
- SES_OBJSTAT_UNRECOV
 - scsi_ses.h, 987
- SES_OBJSTAT_UNSUPPORTED
 - scsi_ses.h, 987
- ses_oidx
 - typidx, 423
- ses_private
 - ses_softc, 399
- ses_runcmd
 - scsi_ses.c, 974
- SES_SAFT
 - scsi_ses.c, 966
- ses_saved_ccb
 - ses_softc, 399
- SES_SEN
 - scsi_ses.c, 966
- SES_SES
 - scsi_ses.c, 966
- SES_SES_PASSTHROUGH
 - scsi_ses.c, 966
- SES_SES_SCSI2
 - scsi_ses.c, 966
- ses_set_encstat
 - scsi_ses.c, 974
- ses_set_objstat
 - scsi_ses.c, 975
- ses_softc, 398
 - periph, 398
 - ses_dev, 398
 - ses_encstat, 398
 - ses_flags, 398
 - ses_nobjects, 399
 - ses_objmap, 399
 - ses_private, 399
 - ses_saved_ccb, 399
 - ses_type, 399
 - ses_vec, 399
- ses_softc_init
 - scsi_ses.c, 975
- ses_softc_t
 - scsi_ses.c, 965
- ses_tidx
 - typidx, 423
- ses_type
 - scsi_ses.c, 976
 - ses_softc, 399
- ses_typidx
 - sscfg, 409
- ses_vec
 - ses_softc, 399
- SES_VLOG
 - scsi_ses.c, 963
- SesArrayControl
 - scsi_ses.c, 966
- SesArrayStatus
 - scsi_ses.c, 963
- sesasync
 - scsi_ses.c, 976
- SesCfgHdr, 400
- SesCfgHdr

- GenCode, 400
 - Nsubenc, 400
 - sescleanup
 - scsi_ses.c, 977, 983
 - sesclose
 - scsi_ses.c, 977, 983
 - SesComStat, 401
 - SesComStat
 - comstat, 401
 - comstatus, 401
 - SesConfigPage
 - scsi_ses.c, 966
 - SesControlPage
 - scsi_ses.c, 966
 - SESCTL_CSEL
 - scsi_ses.h, 987
 - SESCTL_DEVOFF
 - scsi_ses.h, 987
 - SESCTL_DISABLE
 - scsi_ses.h, 987
 - SESCTL_DRVLCK
 - scsi_ses.h, 988
 - SESCTL_PRDFAIL
 - scsi_ses.h, 988
 - SESCTL_RQSFLT
 - scsi_ses.h, 988
 - SESCTL_RQSID
 - scsi_ses.h, 988
 - SESCTL_RQSINS
 - scsi_ses.h, 988
 - SESCTL_RQSRMV
 - scsi_ses.h, 988
 - SESCTL_RQSTFAIL
 - scsi_ses.h, 988
 - SESCTL_RQSTON
 - scsi_ses.h, 988
 - SESCTL_RSTSWAP
 - scsi_ses.h, 988
 - SesDiagPageCodes
 - scsi_ses.c, 966
 - sesdone
 - scsi_ses.c, 977
 - sesdriver
 - scsi_ses.c, 983
 - SesElementDescriptor
 - scsi_ses.c, 966
 - SesEncDesc, 402
 - SesEncDesc
 - encPid, 402
 - encRev, 402
 - encVen, 402
 - encVid, 402
 - encWWN, 402
 - SesEncHdr, 403
 - SesEncHdr
 - Ntypes, 403
 - Subencid, 403
 - VEncLen, 403
 - seserror
 - scsi_ses.c, 978
 - SesHelpTxt
 - scsi_ses.c, 966
 - sesinit
 - scsi_ses.c, 978, 984
 - SESIOC
 - scsi_ses.h, 989
 - SESIOC_GETENCSTAT
 - scsi_ses.h, 989
 - SESIOC_GETNOBJ
 - scsi_ses.h, 989
 - SESIOC_GETOBJMAP
 - scsi_ses.h, 989
 - SESIOC_GETOBJSTAT
 - scsi_ses.h, 989
 - SESIOC_GETTEXT
 - scsi_ses.h, 989
 - SESIOC_INIT
 - scsi_ses.h, 989
 - SESIOC_SETENCSTAT
 - scsi_ses.h, 989
 - SESIOC_SETOBJSTAT
 - scsi_ses.h, 989
- sesioctl
 - scsi_ses.c, 979, 984
 - sesoninvalidate
 - scsi_ses.c, 979, 984
 - sesopen
 - scsi_ses.c, 980, 984
 - sesregister
 - scsi_ses.c, 980, 984
 - SesShortStatus
 - scsi_ses.c, 966
 - sesstart
 - scsi_ses.c, 981, 984
 - SesStatusPage
 - scsi_ses.c, 964
 - SesStringIn
 - scsi_ses.c, 964
 - SesStringOut
 - scsi_ses.c, 966
 - SesThdr, 404
 - SesThdr
 - enc_maxelt, 404
 - enc_subenc, 404
 - enc_tlen, 404
 - enc_type, 404
 - SesThresholdIn
 - scsi_ses.c, 964

- SesThresholdOut
 - scsi_ses.c, 966
- SESTYP_ALARM
 - scsi_ses.h, 990
- SESTYP_AMMETER
 - scsi_ses.h, 990
- SESTYP_COMPORT
 - scsi_ses.h, 990
- SESTYP_DEVICE
 - scsi_ses.h, 990
- SESTYP_DISPLAY
 - scsi_ses.h, 990
- SESTYP_DOORLOCK
 - scsi_ses.h, 990
- SESTYP_ESCC
 - scsi_ses.h, 990
- SESTYP_FAN
 - scsi_ses.h, 990
- SESTYP_KEYPAD
 - scsi_ses.h, 990
- SESTYP_LANGUAGE
 - scsi_ses.h, 990
- SESTYP_NVRAM
 - scsi_ses.h, 991
- SESTYP_POWER
 - scsi_ses.h, 991
- SESTYP_SCC
 - scsi_ses.h, 991
- SESTYP_SCSI_INI
 - scsi_ses.h, 991
- SESTYP_SCSI_TGT
 - scsi_ses.h, 991
- SESTYP_SCSIXVR
 - scsi_ses.h, 991
- SESTYP_SUBENC
 - scsi_ses.h, 991
- SESTYP_THERM
 - scsi_ses.h, 991
- SESTYP_UNSPECIFIED
 - scsi_ses.h, 991
- SESTYP_UPS
 - scsi_ses.h, 991
- SESTYP_VOM
 - scsi_ses.h, 991
- SESUNIT
 - scsi_ses.c, 964
- Set_CCB_Type
 - scsi_sa.c, 911
- SET_CD_SPEED
 - scsi_cd.h, 716
- set_encstat
 - encvec, 158
- set_objstat
 - encvec, 158
- set_objstat_sel
 - scsi_ses.c, 981
- set_scsi_delay
 - scsi_all.c, 603
- SF_NO_PRINT
 - scsi_all.h, 626
- SF_PRINT_ALWAYS
 - scsi_all.h, 626
- SF_QUIET_IR
 - scsi_all.h, 626
- SF_RETRY_UA
 - scsi_all.h, 626
- sget16
 - scsi_ses.c, 964
- sget24
 - scsi_ses.c, 964
- sget32
 - scsi_ses.c, 964
- sget8
 - scsi_ses.c, 964
- sglist_cnt
 - ccb_eng_exec, 66
 - ccb_scsiio, 101
- short_handle
 - cdchanger, 133
- SHORT_INQUIRY_LENGTH
 - scsi_all.h, 626
- SHOW_ALL_NEG
 - scsi_low.h, 855
- SHOW_CALCF_RES
 - scsi_low.h, 855
- SHOW_PROBE_RES
 - scsi_low.h, 856
- SHOW_SYNCH_NEG
 - scsi_low.h, 856
- SHOW_WIDE_NEG
 - scsi_low.h, 856
- si_dev
 - scsi_low_osdep_interface, 248
- SI_EVPD
 - scsi_all.h, 626
- si_poll_count
 - scsi_low_osdep_interface, 248
- SID_ADDITIONAL_LENGTH
 - scsi_all.h, 627
- SID_AENC
 - scsi_all.h, 627
- SID_ANSI_REV
 - scsi_all.h, 627
- SID_CmdQue
 - scsi_all.h, 627
- SID_ECMA
 - scsi_all.h, 627
- SID_IS_REMOVABLE

- scsi_all.h, 627
- SID_ISO
 - scsi_all.h, 627
- SID_Linked
 - scsi_all.h, 627
- SID_PRODUCT_SIZE
 - scsi_all.h, 628
- SID_QUAL
 - scsi_all.h, 628
- SID_QUAL2
 - scsi_all.h, 628
- SID_QUAL_BAD_LU
 - scsi_all.h, 628
- SID_QUAL_IS_VENDOR_UNIQUE
 - scsi_all.h, 628
- SID_QUAL_LU_CONNECTED
 - scsi_all.h, 628
- SID_QUAL_LU_OFFLINE
 - scsi_all.h, 628
- SID_QUAL_RSVD
 - scsi_all.h, 628
- SID_RelAdr
 - scsi_all.h, 628
- SID_REVISION_SIZE
 - scsi_all.h, 629
- SID_SftRe
 - scsi_all.h, 629
- SID_SPI_CLOCK_DT
 - scsi_all.h, 629
- SID_SPI_CLOCK_DT_ST
 - scsi_all.h, 629
- SID_SPI_CLOCK_ST
 - scsi_all.h, 629
- SID_SPI_IUS
 - scsi_all.h, 629
- SID_SPI_MASK
 - scsi_all.h, 629
- SID_SPI_QAS
 - scsi_all.h, 629
- SID_Sync
 - scsi_all.h, 629
- SID_TrmIOP
 - scsi_all.h, 629
- SID_TYPE
 - scsi_all.h, 630
- SID_VENDOR_SIZE
 - scsi_all.h, 630
- SID_VENDOR_SPECIFIC_0_SIZE
 - scsi_all.h, 630
- SID_VENDOR_SPECIFIC_1_SIZE
 - scsi_all.h, 630
- SID_WBus16
 - scsi_all.h, 630
- SID_WBus32
 - scsi_all.h, 630
- scsi_all.h, 630
- sim
 - scsi_low_osdep_interface, 249
- sim_action
 - cam_sim, 40
- sim_action_func
 - cam_sim.h, 512
- SIM_DEAD
 - cam_xpt.c, 524
- SIM_IDLEN
 - cam_ccb.h, 453
- sim_links
 - ccb_hdr, 78
- sim_name
 - cam_sim, 40
- sim_poll
 - cam_sim, 40
- sim_poll_func
 - cam_sim.h, 512
- sim_priv
 - ccb_hdr, 78
- sim_vid
 - ccb_pathinq, 86
- SIP_MEDIA_FIXED
 - scsi_all.h, 630
- SIP_MEDIA_REMOVABLE
 - scsi_all.h, 630
- SIU_PFC_CIU_FIELDS_INVALID
 - scsi_iu.h, 786
- SIU_PFC_ILLEGAL_REQUEST
 - scsi_iu.h, 786
- SIU_PFC_INVALID_TYPE_CODE
 - scsi_iu.h, 786
- SIU_PFC_NONE
 - scsi_iu.h, 786
- SIU_PFC_TMF_FAILED
 - scsi_iu.h, 787
- SIU_PFC_TMF_NOT_SUPPORTED
 - scsi_iu.h, 787
- SIU_PKTFAIL_CODE
 - scsi_iu.h, 787
- SIU_PKTFAIL_OFFSET
 - scsi_iu.h, 787
- SIU_RSPVALID
 - scsi_iu.h, 787
- SIU_SENSE_OFFSET
 - scsi_iu.h, 787
- SIU_SNSVALID
 - scsi_iu.h, 787
- SIU_TASKMGMT_ABORT_TASK
 - scsi_iu.h, 787
- SIU_TASKMGMT_ABORT_TASK_SET
 - scsi_iu.h, 787
- SIU_TASKMGMT_CLEAR_ACA

- scsi_iu.h, 787
- SIU_TASKMGMT_CLEAR_TASK_SET
 - scsi_iu.h, 788
- SIU_TASKMGMT_LUN_RESET
 - scsi_iu.h, 788
- SIU_TASKMGMT_NONE
 - scsi_iu.h, 788
- SIU_TASKMGMT_TARGET_RESET
 - scsi_iu.h, 788
- sl_active
 - scsi_low_softc, 253
- sl_atten
 - scsi_low_softc, 253
- sl_cfgflags
 - scsi_low_softc, 253
- sl_clear_atten
 - scsi_low_softc, 253
- sl_dev
 - scsi_low.h, 856
- sl_disc
 - scsi_low_softc, 253
- sl_error
 - scsi_low_softc, 253
- sl_flags
 - scsi_low_softc, 254
- sl_funcs
 - scsi_low_softc, 254
- sl_hostid
 - scsi_low_softc, 254
- sl_Lnexus
 - scsi_low_softc, 254
- sl_max_retry
 - scsi_low_softc, 254
- sl_msgphase
 - scsi_low_softc, 254
- sl_nexus_call
 - scsi_low_softc, 254
- sl_nio
 - scsi_low_softc, 254
- sl_nluns
 - scsi_low_softc, 254
- sl_ntargs
 - scsi_low_softc, 254
- sl_openings
 - scsi_low_softc, 255
- sl_osdep_fp
 - scsi_low_softc, 255
- sl_ph_count
 - scsi_low_softc, 255
- sl_powc
 - scsi_low_softc, 255
- sl_Qnexus
 - scsi_low_softc, 255
- sl_retry_sel
 - scsi_low_softc, 255
- sl_rstep
 - scsi_low_softc, 255
- sl_scp
 - scsi_low_softc, 255
- sl_selid
 - scsi_low_softc, 255
- sl_show_result
 - scsi_low_softc, 255
- sl_si
 - scsi_low_softc, 256
- sl_start
 - scsi_low_softc, 256
- sl_tab
 - scsi_low.c, 828
- sl_targsize
 - scsi_low_softc, 256
- sl_ti
 - scsi_low_softc, 256
- sl_timeout_count
 - scsi_low_softc, 256
- sl_titab
 - scsi_low_softc, 256
- sl_Tnexus
 - scsi_low_softc, 256
- sl_xname
 - scsi_low_softc, 256
- slccb, 405
 - bp, 406
 - ccb_datalen, 406
 - ccb_error, 406
 - ccb_flags, 406
 - ccb_msgoutflag, 406
 - ccb_omsgoutflag, 406
 - ccb_otag, 406
 - ccb_rcnt, 407
 - ccb_scp, 407
 - ccb_scsi_cmd, 407
 - ccb_selrcnt, 407
 - ccb_sense, 407
 - ccb_sscp, 407
 - ccb_tag, 407
 - ccb_tc, 407
 - ccb_tcmx, 408
 - li, 408
 - osdep, 408
 - TAILQ_ENTRY, 406
 - ti, 408
- sli_fixed
 - scsi_sa_rw, 363
- SLIST_ENTRY
 - cam_periph, 34
 - camq_entry, 45
- SLIST_HEAD

- cam_periph, 34
- cam_queue.h, 507
- cam_xpt.c, 531
- scsi_da.c, 768
- sml_msg
 - scsi_low_msg_log, 244
- sml_ptr
 - scsi_low_msg_log, 244
- slotoff
 - scfg, 208
- SLP_DS
 - scsi_all.h, 630
- SLP_DU
 - scsi_all.h, 631
- SLP_ETC
 - scsi_all.h, 631
- SLP_LBIN
 - scsi_all.h, 631
- SLP_LP
 - scsi_all.h, 631
- SLP_TMC_ALWAYS
 - scsi_all.h, 631
- SLP_TMC_EQUAL
 - scsi_all.h, 631
- SLP_TMC_GREATER
 - scsi_all.h, 631
- SLP_TMC_MASK
 - scsi_all.h, 631
- SLP_TMC_NOTEQUAL
 - scsi_all.h, 631
- SLP_TSD
 - scsi_all.h, 631
- SLS_ALL_PAGES_PAGE
 - scsi_all.h, 631
- SLS_ERROR_LASTN_PAGE
 - scsi_all.h, 632
- SLS_ERROR_NONMEDIUM_PAGE
 - scsi_all.h, 632
- SLS_ERROR_READ_PAGE
 - scsi_all.h, 632
- SLS_ERROR_READREVERSE_PAGE
 - scsi_all.h, 632
- SLS_ERROR_VERIFY_PAGE
 - scsi_all.h, 632
- SLS_ERROR_WRITE_PAGE
 - scsi_all.h, 632
- SLS_OVERRUN_PAGE
 - scsi_all.h, 632
- SLS_PAGE_CODE
 - scsi_all.h, 632
- SLS_PAGE_CTRL_CUMUL_DEFAULT
 - scsi_all.h, 632
- SLS_PAGE_CTRL_CUMULATIVE
 - scsi_all.h, 632
- SLS_PAGE_CTRL_MASK
 - scsi_all.h, 632
- SLS_PAGE_CTRL_THRESH_DEFAULT
 - scsi_all.h, 633
- SLS_PAGE_CTRL_THRESHOLD
 - scsi_all.h, 633
- SLS_PCR
 - scsi_all.h, 633
- SLS_PPC
 - scsi_all.h, 633
- SLS_SP
 - scsi_all.h, 633
- SLSC_MODE_SENSE_SHORT
 - scsi_low.c, 799
- SLU_EOT
 - scsi_sa.h, 945
- SLU_IMMED
 - scsi_sa.h, 945
- SLU_LOAD
 - scsi_sa.h, 945
- SLU_RETEN
 - scsi_sa.h, 945
- SMH_SA_BUF_MODE_MASK
 - scsi_sa.h, 945
- SMH_SA_BUF_MODE_MIBUF
 - scsi_sa.h, 945
- SMH_SA_BUF_MODE_NOBUF
 - scsi_sa.h, 945
- SMH_SA_BUF_MODE_SIBUF
 - scsi_sa.h, 946
- SMH_SA_SPEED_DEFAULT
 - scsi_sa.h, 946
- SMH_SA_SPEED_MASK
 - scsi_sa.h, 946
- SMH_SA_WP
 - scsi_sa.h, 946
- SMS_ALL_PAGES_PAGE
 - scsi_all.h, 633
- sms_cmd
 - scsi_low.c, 828
- sms_cmp
 - lun_info::scsi_low_mode_sense_data, 171
- SMS_CONTROL_MODE_PAGE
 - scsi_all.h, 633
- SMS_DBD
 - scsi_all.h, 633
- SMS_DISCONNECT_RECONNECT_PAGE
 - scsi_all.h, 633
- SMS_FLEXIBLE_GEOMETRY_PAGE
 - scsi_da.h, 779
- SMS_FLEXIBLE_GEOMETRY_PLEN
 - scsi_da.h, 779
- SMS_FORMAT_DEVICE_PAGE
 - scsi_da.h, 779

- SMS_FORMAT_DEVICE_PLEN
 - scsi_da.h, 779
- sms_header
 - lun_info::scsi_low_mode_sense_data, 171
- SMS_PAGE_CODE
 - scsi_all.h, 633
- SMS_PAGE_CTRL_CHANGEABLE
 - scsi_all.h, 634
- SMS_PAGE_CTRL_CURRENT
 - scsi_all.h, 634
- SMS_PAGE_CTRL_DEFAULT
 - scsi_all.h, 634
- SMS_PAGE_CTRL_MASK
 - scsi_all.h, 634
- SMS_PAGE_CTRL_SAVED
 - scsi_all.h, 634
- SMS_PERIPHERAL_DEVICE_PAGE
 - scsi_all.h, 634
- SMS_PF
 - scsi_all.h, 634
- SMS_RIGID_GEOMETRY_PAGE
 - scsi_da.h, 779
- SMS_RIGID_GEOMETRY_PLEN
 - scsi_da.h, 779
- SMS_RW_ERROR_RECOVERY_PAGE
 - scsi_da.h, 779
- SMS_RWER_ARRE
 - scsi_da.h, 779
- SMS_RWER_AWRE
 - scsi_da.h, 779
- SMS_RWER_DCR
 - scsi_da.h, 779
- SMS_RWER_DTE
 - scsi_da.h, 779
- SMS_RWER_EER
 - scsi_da.h, 780
- SMS_RWER_PER
 - scsi_da.h, 780
- SMS_RWER_RC
 - scsi_da.h, 780
- SMS_RWER_TB
 - scsi_da.h, 780
- SMS_SP
 - scsi_all.h, 634
- SMS_VENDOR_SPECIFIC_PAGE
 - scsi_all.h, 634
- softc
 - cam_periph, 35
 - cam_sim, 40
- softc_init
 - encvec, 158
- sony
 - cam_xpt.c, 568
- sony_mo_entries
 - scsi_all.c, 606
- SPACE
 - scsi_sa.h, 946
- SPACE_TIMEOUT
 - scsi_sa.c, 911
- spc
 - disk_pages::flexible_disk_page, 148
- spc2_flags
 - scsi_inquiry_data, 231
- SPC2_SID_BQueue
 - scsi_all.h, 634
- SPC2_SID_EncServ
 - scsi_all.h, 635
- SPC2_SID_MChngr
 - scsi_all.h, 635
- SPC2_SID_MultiP
 - scsi_all.h, 635
- speed
 - sa_softc, 204
- spi
 - ccb_pathinq, 86
 - ccb_trans_settings, 109
- spi3data
 - scsi_inquiry_data, 231
- spriv_field0
 - cam_sim.h, 512
- spriv_field1
 - cam_sim.h, 512
- spriv_ptr0
 - cam_sim.h, 512
- spriv_ptr1
 - cam_sim.h, 512
- src
 - scsi_exchange_medium, 223
 - scsi_move_medium, 272
- SRC16_PMI
 - scsi_all.h, 635
- SRC16_RELADR
 - scsi_all.h, 635
- SRC16_SERVICE_ACTION
 - scsi_all.h, 635
- src_resid
 - ccb_eng_exec, 66
- SRDD10_BLOCK_FORMAT
 - scsi_da.h, 780
- SRDD10_BYTES_FROM_INDEX_FORMAT
 - scsi_da.h, 780
- SRDD10_DLIST_FORMAT_MASK
 - scsi_da.h, 780
- SRDD10_GLIST
 - scsi_da.h, 780
- SRDD10_LUN_MASK
 - scsi_da.h, 780
- SRDD10_PHYSICAL_SECTOR_FORMAT

- scsi_da.h, 780
- SRDD10_PLIST
 - scsi_da.h, 780
- SRDD12_BLOCK_FORMAT
 - scsi_da.h, 781
- SRDD12_BYTES_FROM_INDEX_FORMAT
 - scsi_da.h, 781
- SRDD12_DLIST_FORMAT_MASK
 - scsi_da.h, 781
- SRDD12_GLIST
 - scsi_da.h, 781
- SRDD12_LUN_MASK
 - scsi_da.h, 781
- SRDD12_PHYSICAL_SECTOR_FORMAT
 - scsi_da.h, 781
- SRDD12_PLIST
 - scsi_da.h, 781
- SRDDH10_BLOCK_FORMAT
 - scsi_da.h, 781
- SRDDH10_BYTES_FROM_INDEX_FORMAT
 - scsi_da.h, 781
- SRDDH10_DLIST_FORMAT_MASK
 - scsi_da.h, 781
- SRDDH10_GLIST
 - scsi_da.h, 781
- SRDDH10_PHYSICAL_SECTOR_FORMAT
 - scsi_da.h, 782
- SRDDH10_PLIST
 - scsi_da.h, 782
- SRDDH12_BLOCK_FORMAT
 - scsi_da.h, 782
- SRDDH12_BYTES_FROM_INDEX_FORMAT
 - scsi_da.h, 782
- SRDDH12_DLIST_FORMAT_MASK
 - scsi_da.h, 782
- SRDDH12_GLIST
 - scsi_da.h, 782
- SRDDH12_PHYSICAL_SECTOR_FORMAT
 - scsi_da.h, 782
- SRDDH12_PLIST
 - scsi_da.h, 782
- SREW_IMMED
 - scsi_sa.h, 946
- SRFC_LUN_MASK
 - scsi_da.h, 782
- SRRU_3RD_MASK
 - scsi_sa.h, 946
- SRRU_3RD_PARTY
 - scsi_sa.h, 946
- SRRU_3RD_SHAMT
 - scsi_sa.h, 947
- SRRU_LUN_MASK
 - scsi_sa.h, 947
- SRS_SUBQ
 - scsi_cd.h, 716
- SRW10_DPO
 - scsi_all.h, 635
- SRW10_EBP
 - scsi_all.h, 635
- SRW10_FUA
 - scsi_all.h, 635
- SRW10_RELADDR
 - scsi_all.h, 635
- SRW12_DPO
 - scsi_all.h, 636
- SRW12_FUA
 - scsi_all.h, 636
- SRW12_RELADDR
 - scsi_all.h, 636
- SRW16_DPO
 - scsi_all.h, 636
- SRW16_FUA
 - scsi_all.h, 636
- SRW16_RELADDR
 - scsi_all.h, 636
- SRW_TOPADDR
 - scsi_all.h, 636
- SRZU_LUN_MASK
 - scsi_da.h, 782
- SS_BLOCKS
 - scsi_sa.h, 947
- ss_cmd
 - scsi_low.c, 829
- SS_EOD
 - scsi_sa.h, 947
- SS_ERRMASK
 - scsi_all.h, 636
- SS_FAIL
 - scsi_all.h, 644
- SS_FATAL
 - scsi_all.h, 636
- SS_FILEMARKS
 - scsi_sa.h, 947
- SS_MASK
 - scsi_all.h, 644
- SS_NOP
 - scsi_all.h, 644
- SS_RDEF
 - scsi_all.h, 636
- SS_REQSENSE
 - scsi_all.h, 644
- SS_RET
 - scsi_all.h, 636
- SS_RETRY
 - scsi_all.h, 644
- SS_SEQFILEMARKS
 - scsi_sa.h, 947
- SS_SEQSETMARKS

- scsi_sa.h, 947
- SS_SETMARKS
 - scsi_sa.h, 947
- SS_START
 - scsi_all.h, 644
- SS_TUR
 - scsi_all.h, 644
- sscfg, 409
 - ses_eltmap, 409
 - ses_ntypes, 409
 - ses_typedix, 409
- SSD_BITPTR_VALID
 - scsi_all.h, 637
- SSD_BITPTR_VALUE
 - scsi_all.h, 637
- SSD_CURRENT_ERROR
 - scsi_all.h, 637
- SSD_DEFERRED_ERROR
 - scsi_all.h, 637
- SSD_DOL
 - scsi_all.h, 637
- SSD_EOM
 - scsi_all.h, 637
- SSD_ERRCODE
 - scsi_all.h, 637
- SSD_ERRCODE_VALID
 - scsi_all.h, 637
- SSD_FIELDPTR_CMD
 - scsi_all.h, 637
- SSD_FILEMARK
 - scsi_all.h, 637
- SSD_FULL_SIZE
 - scsi_all.h, 638
- SSD_ILI
 - scsi_all.h, 638
- SSD_KEY
 - scsi_all.h, 638
- SSD_KEY_ABORTED_COMMAND
 - scsi_all.h, 638
- SSD_KEY_BLANK_CHECK
 - scsi_all.h, 638
- SSD_KEY_COPY_ABORTED
 - scsi_all.h, 638
- SSD_KEY_DATA_PROTECT
 - scsi_all.h, 638
- SSD_KEY_EQUAL
 - scsi_all.h, 638
- SSD_KEY_HARDWARE_ERROR
 - scsi_all.h, 639
- SSD_KEY_ILLEGAL_REQUEST
 - scsi_all.h, 639
- SSD_KEY_MEDIUM_ERROR
 - scsi_all.h, 639
- SSD_KEY_MISCOMPARE
 - scsi_all.h, 639
- SSD_KEY_NO_SENSE
 - scsi_all.h, 639
- SSD_KEY_NOT_READY
 - scsi_all.h, 639
- SSD_KEY_RECOVERED_ERROR
 - scsi_all.h, 639
- SSD_KEY_RESERVED
 - scsi_all.h, 639
- SSD_KEY_UNIT_ATTENTION
 - scsi_all.h, 639
- SSD_KEY_Vendor_Specific
 - scsi_all.h, 640
- SSD_KEY_VOLUME_OVERFLOW
 - scsi_all.h, 640
- SSD_MIN_SIZE
 - scsi_all.h, 640
- SSD_PF
 - scsi_all.h, 640
- SSD_SCS_VALID
 - scsi_all.h, 640
- SSD_SELFTEST
 - scsi_all.h, 640
- SSD_UOL
 - scsi_all.h, 640
- ssea
 - read_element_status_descriptor, 191
- sset16
 - scsi_ses.c, 964
- sset24
 - scsi_ses.c, 965
- sset32
 - scsi_ses.c, 965
- sset8
 - scsi_ses.c, 965
- SSQ_DECREMENT_COUNT
 - scsi_all.h, 644
- SSQ_MANY
 - scsi_all.h, 644
- SSQ_MASK
 - scsi_all.h, 644
- SSQ_NONE
 - scsi_all.h, 644
- SSQ_PRINT_SENSE
 - scsi_all.h, 644
- SSQ_RANGE
 - scsi_all.h, 644
- SSS_FLAG_NONE
 - scsi_all.h, 645
- SSS_FLAG_PRINT_COMMAND
 - scsi_all.h, 645
- SSS_IMMED
 - scsi_all.h, 640
- SSS_LOEJ

- scsi_all.h, 640
- SSS_START
 - scsi_all.h, 640
- SST
 - scsi_all.c, 592
- ST_BUSY
 - scsi_low.h, 856
- ST_CHKCOND
 - scsi_low.h, 856
- ST_CMDTERM
 - scsi_low.h, 856
- ST_CONFLICT
 - scsi_low.h, 856
- st_cyl_rwc_0
 - disk_pages::flexible_disk_page, 149
 - disk_pages::rigid_geometry_page, 155
- st_cyl_rwc_1
 - disk_pages::flexible_disk_page, 149
 - disk_pages::rigid_geometry_page, 155
- st_cyl_rwc_2
 - disk_pages::rigid_geometry_page, 155
- st_cyl_wp_0
 - disk_pages::flexible_disk_page, 149
 - disk_pages::rigid_geometry_page, 155
- st_cyl_wp_1
 - disk_pages::flexible_disk_page, 149
 - disk_pages::rigid_geometry_page, 155
- st_cyl_wp_2
 - disk_pages::rigid_geometry_page, 155
- ST_GOOD
 - scsi_low.h, 856
- ST_INTERGOOD
 - scsi_low.h, 856
- ST_INTERMET
 - scsi_low.h, 857
- ST_MET
 - scsi_low.h, 857
- ST_QUEFULL
 - scsi_low.h, 857
- ST_UNKNOWN
 - scsi_low.h, 857
- STAILQ_ENTRY
 - camq_entry, 45
 - cd_mode_params, 122
- STAILQ_HEAD
 - cam_xpt.c, 531
 - scsi_cd.c, 691
- start_f
 - scsi_play_msf, 278
- start_index
 - scsi_play_track, 281
- start_m
 - scsi_play_msf, 279
- start_s
 - scsi_play_msf, 279
- START_STOP
 - scsi_all.h, 641
- START_STOP_UNIT
 - scsi_all.h, 641
- start_time
 - cdchanger, 133
- start_track
 - scsi_play_track, 281
- state
 - cd_softc, 127
 - ch_softc, 135
 - pass_softc, 181
 - sa_softc, 204
 - targ_softc, 418
 - targbh_softc, 422
- STATERR
 - scsi_low.h, 857
- status
 - ccb_dev_match, 59
 - ccb_getdevlist, 73
 - ccb_hdr, 78
 - scsi_status_iu_header, 380
 - targbh_cmd_desc, 420
- status_code
 - cam_status_entry, 42
- status_text
 - cam_status_entry, 42
- stor
 - page_device_capabilities, 175
- STOR_DT
 - scsi_ch.h, 748
- STOR_IE
 - scsi_ch.h, 748
- STOR_MT
 - scsi_ch.h, 749
- STOR_ST
 - scsi_ch.h, 749
- STRNCMP
 - scsi_ses.c, 965
- struct_len
 - read_dvd_struct_list_entry, 188
- subchan_format
 - scsi_read_subchannel, 330
- Subencid
 - SesEncHdr, 403
- subencid
 - ses_object, 396
- subenclosure
 - encobj, 157
- success
 - scsi_report_key_data_asf, 339
- svalid
 - encobj, 157

- SVFY_BYTECHK
 - scsi_da.h, 782
- SVFY_DPO
 - scsi_da.h, 783
- SVFY_LUN_MASK
 - scsi_da.h, 783
- SVFY_RELADR
 - scsi_da.h, 783
- SVPD_SERIAL_NUM_SIZE
 - scsi_all.h, 641
- SVPD_UNIT_SERIAL_NUMBER
 - scsi_all.h, 641
- swap
 - cam_queue.c, 499
- SWFMRK_IMMED
 - scsi_sa.h, 947
- SWFMRK_WSMK
 - scsi_sa.h, 947
- SWVY_BYTECHK
 - scsi_da.h, 783
- SWVY_DPO
 - scsi_da.h, 783
- SWVY_LUN_MASK
 - scsi_da.h, 783
- SWVY_RELADR
 - scsi_da.h, 783
- sync_offset
 - ccb_trans_settings_spi, 114
- sync_period
 - ccb_trans_settings_spi, 114
- SYNCHRONIZE_CACHE
 - scsi_all.h, 641
- sysctl_cam_search_luns
 - cam_xpt.c, 531
- SYSCTL_INT
 - scsi_cd.c, 692
 - scsi_da.c, 769, 770
- SYSCTL_NODE
 - cam.c, 433
 - scsi_cd.c, 692
 - scsi_da.c, 770
- SYSCTL_PROC
 - cam_xpt.c, 531
 - scsi_all.c, 604
- sysctl_scsi_delay
 - scsi_all.c, 604
- SYSINIT
 - scsi_all.c, 604
- T
 - scsi_all.c, 592
- T_ANY
 - scsi_all.h, 641
- T_ASC0
 - scsi_all.h, 641
- T_ASC1
 - scsi_all.h, 641
- T_CDROM
 - scsi_all.h, 641
- T_CHANGER
 - scsi_all.h, 641
- T_COMM
 - scsi_all.h, 642
- T_DIRECT
 - scsi_all.h, 642
- T_ENCLOSURE
 - scsi_all.h, 642
- T_FIXED
 - scsi_all.h, 642
- T_NODEVICE
 - scsi_all.h, 642
- T_OCRW
 - scsi_all.h, 642
- T_OPTICAL
 - scsi_all.h, 642
- T_PRINTER
 - scsi_all.h, 642
- T_PROCESSOR
 - scsi_all.h, 642
- T_RBC
 - scsi_all.h, 643
- T_REMOV
 - scsi_all.h, 643
- T_SCANNER
 - scsi_all.h, 643
- T_SEQUENTIAL
 - scsi_all.h, 643
- T_STORARRAY
 - scsi_all.h, 643
- T_WORM
 - scsi_all.h, 643
- tag_action
 - ccb_accept_tio, 54
 - ccb_scsiio, 101
- tag_id
 - ccb_accept_tio, 54
 - ccb_scsiio, 101
- TAILQ_ENTRY
 - cam_periph, 34
 - camq_entry, 45
 - slccb, 406
 - targ_info, 412
- TAILQ_HEAD
 - cam_queue.h, 507
 - cam_xpt.c, 532
 - periph_driver, 182
 - scsi_low.h, 865
 - scsi_targ_bh.c, 997

- scsi_target.c, 1008
- scsi_targetio.h, 1020
- targ_cdevsw
 - scsi_target.c, 1016
- targ_cmd_descr, 410
 - mapinfo, 410
- targ_descr
 - scsi_target.c, 1007
- targ_info, 411
 - TAILQ_ENTRY, 412
 - ti_disc, 412
 - ti_diskflags, 412
 - ti_emsgflags, 412
 - ti_flags_valid, 412
 - ti_id, 412
 - ti_litab, 413
 - ti_log_msgin, 413
 - ti_log_msgout, 413
 - ti_lunsize, 413
 - ti_maxsynch, 413
 - ti_msgflags, 413
 - ti_msgin, 413
 - ti_msgin_parity_error, 413
 - ti_msginlen, 414
 - ti_msginptr, 414
 - ti_msgoutlen, 414
 - ti_msgoutstr, 414
 - ti_omsgflags, 414
 - ti_ophase, 414
 - ti_osynch, 414
 - ti_owidth, 414
 - ti_phase, 415
 - ti_quirks, 415
 - ti_sc, 415
 - ti_setup_msg, 415
 - ti_setup_msg_done, 415
 - ti_slti, 415
 - ti_width, 415
- targ_info::synch, 416
 - offset, 416
 - period, 416
- targ_softc, 417
 - abort_queue, 417
 - device_stats, 417
 - path, 417
 - pending_ccb_queue, 417
 - periph, 418
 - read_select, 418
 - state, 418
 - user_ccb_queue, 418
 - work_queue, 418
- targ_state
 - scsi_target.c, 1007
- TARG_STATE_LUN_ENABLED
 - scsi_target.c, 1007
- TARG_STATE_OPENED
 - scsi_target.c, 1007
- TARG_STATE_RESV
 - scsi_target.c, 1007
- targasync
 - scsi_target.c, 1008
- targbh_ccb_types
 - scsi_targ_bh.c, 996
- TARGBH_CCB_WAITING
 - scsi_targ_bh.c, 996
- TARGBH_CCB_WORKQ
 - scsi_targ_bh.c, 996
- targbh_cmd_descr, 419
 - atio_link, 419
 - backing_store, 419
 - data, 419
 - data_increment, 420
 - data_resid, 420
 - max_size, 420
 - status, 420
 - timeout, 420
- TARGBH_FLAG_LUN_ENABLED
 - scsi_targ_bh.c, 996
- TARGBH_FLAG_NONE
 - scsi_targ_bh.c, 996
- targbh_flags
 - scsi_targ_bh.c, 996
- targbh_softc, 421
 - accept_tio_list, 421
 - device_stats, 421
 - flags, 422
 - immed_notify_slist, 422
 - init_level, 422
 - inq_data_len, 422
 - pending_queue, 422
 - state, 422
 - unknown_atio_queue, 422
 - work_queue, 422
- targbh_state
 - scsi_targ_bh.c, 996
- TARGBH_STATE_EXCEPTION
 - scsi_targ_bh.c, 996
- TARGBH_STATE_NORMAL
 - scsi_targ_bh.c, 996
- TARGBH_STATE_TEARDOWN
 - scsi_targ_bh.c, 996
- targbhallocdescr
 - scsi_targ_bh.c, 997
- targbhasync
 - scsi_targ_bh.c, 997
- targbhctor
 - scsi_targ_bh.c, 997, 1002
- targbhdislun

- scsi_targ_bh.c, 998
- targbhdone
 - scsi_targ_bh.c, 998
- targbhdriver
 - scsi_targ_bh.c, 1002
- targbhdtdor
 - scsi_targ_bh.c, 999, 1002
- targbhenlun
 - scsi_targ_bh.c, 999
- targbhfreedescri
 - scsi_targ_bh.c, 1000
- targbhinit
 - scsi_targ_bh.c, 1000, 1003
- targbhstart
 - scsi_targ_bh.c, 1001, 1003
- targcamstatus
 - scsi_target.c, 1008
- targccblen
 - scsi_target.c, 1008
- targclone
 - scsi_target.c, 1008
- targclose
 - scsi_target.c, 1008, 1017
- targctor
 - scsi_target.c, 1009, 1017
- targdisable
 - scsi_target.c, 1009
- targdone
 - scsi_target.c, 1009
- targdriver
 - scsi_target.c, 1017
- targdtor
 - scsi_target.c, 1010, 1017
- targenable
 - scsi_target.c, 1010
- targendislun
 - scsi_target.c, 1011
- target
 - cam_path, 32
 - ccb_dm_cookie, 61
- target_id
 - ccb_hdr, 78
 - cdchanger, 133
 - device_match_pattern, 141
 - device_match_result, 143
 - ioc_enable_lun, 163
 - periph_match_pattern, 183
 - periph_match_result, 184
- target_id_t
 - cam.h, 438
- target_lun
 - ccb_hdr, 78
 - device_match_pattern, 141
 - device_match_result, 143
 - periph_match_pattern, 183
 - periph_match_result, 184
- target_sprt
 - ccb_pathinq, 86
- targfreeccb
 - scsi_target.c, 1011
- targetccb
 - scsi_target.c, 1012
- targetdescr
 - scsi_target.c, 1012
- targinit
 - scsi_target.c, 1012, 1017
- TARGIOCDEBUG
 - scsi_targetio.h, 1019
- TARGIOCDISABLE
 - scsi_targetio.h, 1019
- TARGIOCENABLE
 - scsi_targetio.h, 1020
- targioctl
 - scsi_target.c, 1012, 1017
- targkqfilter
 - scsi_target.c, 1013, 1017
- targopen
 - scsi_target.c, 1013, 1017
- targpoll
 - scsi_target.c, 1013, 1017
- targread
 - scsi_target.c, 1014, 1018
- targread_filtops
 - scsi_target.c, 1018
- targreadfilt
 - scsi_target.c, 1014
- targreadfiltdetach
 - scsi_target.c, 1014
- targreturnccb
 - scsi_target.c, 1014
- targsendccb
 - scsi_target.c, 1015
- targstart
 - scsi_target.c, 1015, 1018
- targusermerge
 - scsi_target.c, 1015
- targwrite
 - scsi_target.c, 1016, 1018
- tea
 - scsi_exchange_medium, 224
 - scsi_move_medium, 273
 - scsi_position_to_element, 283
- termio_ccb
 - ccb_termio, 107
- TEST_UNIT_READY
 - scsi_all.h, 643
- tg
 - scsi_mode_sense_data, 271

- ti
 - scsi_low.h, 865, 866
 - slccb, 408
 - ti_disc
 - targ_info, 412
 - ti_diskflags
 - targ_info, 412
 - ti_emsgflags
 - targ_info, 412
 - ti_flags_valid
 - targ_info, 412
 - ti_id
 - targ_info, 412
 - ti_litab
 - targ_info, 413
 - ti_log_msgin
 - targ_info, 413
 - ti_log_msgout
 - targ_info, 413
 - ti_lunsize
 - targ_info, 413
 - ti_maxsynch
 - targ_info, 413
 - ti_msgflags
 - targ_info, 413
 - ti_msgin
 - targ_info, 413
 - ti_msgin_parity_error
 - targ_info, 413
 - ti_msginlen
 - targ_info, 414
 - ti_msginptr
 - targ_info, 414
 - ti_msgoutlen
 - targ_info, 414
 - ti_msgoutstr
 - targ_info, 414
 - ti_omsgflags
 - targ_info, 414
 - ti_ophase
 - targ_info, 414
 - ti_osynch
 - targ_info, 414
 - ti_owidth
 - targ_info, 414
 - ti_phase
 - targ_info, 415
 - ti_quirks
 - targ_info, 415
 - ti_sc
 - targ_info, 415
 - ti_setup_msg
 - targ_info, 415
 - ti_setup_msg_done
 - targ_info, 415
 - ti_slti
 - targ_info, 415
 - ti_width
 - targ_info, 415
- timeout
 - ccb_eng_exec, 66
 - ccb_hdr, 79
 - targbh_cmd_desc, 420
 - timeout_ch
 - ccb_hdr, 79
 - scsi_low_osdep_interface, 249
 - TIMEOUTIO
 - scsi_low.h, 857
 - tio
 - ccb, 50
 - title_key
 - scsi_report_key_data_title, 345
 - total_packs
 - scsi_read_dvd_struct_data_disc_key_blk, 309
 - tr_arg
 - xpt_traverse_config, 428
 - tr_func
 - xpt_traverse_config, 428
 - track
 - scsi_play_rel_12, 280
 - scsi_read_subchannel, 330
 - transport
 - ccb_pathinq, 86
 - ccb_trans_settings, 109
 - transport_version
 - ccb_pathinq, 86
 - ccb_trans_settings, 109
 - trdy_ssn_mo
 - disk_pages::flexible_disk_page, 149
 - trk_skew_0
 - disk_pages::format_device_page, 152
 - trk_skew_1
 - disk_pages::format_device_page, 152
 - trk_z_0
 - disk_pages::format_device_page, 152
 - trk_z_1
 - disk_pages::format_device_page, 152
 - TUNABLE_INT
 - cam_periph.c, 478, 479
 - cam_xpt.c, 532
 - scsi_cd.c, 692
 - scsi_da.c, 770
 - tw_chars
 - scsi_low.c, 829
 - tw_pos
 - scsi_low.c, 829
 - TWIDDLEWAIT
 - scsi_low.c, 799

- type
 - cam_periph, 36
 - ccb_trans_settings, 110
 - dev_match_pattern, 139
 - dev_match_result, 140
 - read_element_status_page_header, 193
 - scsi_inquiry_pattern, 234
 - scsi_static_inquiry_pattern, 379
- typeid, 423
 - ses_oidx, 423
 - ses_tid, 423
- UACAERR
 - scsi_low.h, 857
- UBFERR
 - scsi_low.h, 857
- unit_number
 - bus_match_pattern, 22
 - bus_match_result, 23
 - cam_periph, 36
 - cam_sim, 41
 - ccb_getdevlist, 73
 - ccb_pathinq, 86
 - periph_match_pattern, 183
 - periph_match_result, 184
- unit_ready_cmd
 - scsi_low.c, 829
- unknown_atio_queue
 - targbh_softc, 422
- unknown_desc_actions
 - scsi_read_dvd_struct_data_generic_dcb, 310
- unused
 - cd_audio_page, 117
 - format_capacity_list_header, 160
 - scsi_changedef, 209
 - scsi_mode_header_10, 262
 - scsi_mode_select_10, 265
 - scsi_mode_select_6, 266
 - scsi_mode_sense_10, 267
 - scsi_mode_sense_6, 268
 - scsi_pause, 275
 - scsi_play_10, 276
 - scsi_play_12, 277
 - scsi_play_msf, 279
 - scsi_play_track, 282
 - scsi_prevent, 285
 - scsi_read_block_limits, 286
 - scsi_read_capacity, 289
 - scsi_read_cd_capacity, 296
 - scsi_read_header, 329
 - scsi_read_subchannel, 331
 - scsi_read_toc, 332
 - scsi_reassign_blocks, 333
 - scsi_release, 335
 - scsi_request_sense, 349
 - scsi_reserve, 352
 - scsi_send_diag, 364
 - scsi_sense, 371
 - scsi_test_unit_ready, 387
- unused1
 - scsi_changedef, 209
 - scsi_play_track, 282
- UNUSED_PARAMETER
 - scsi_sa.c, 911
- unused_primary
 - scsi_read_dvd_struct_data_spare_area, 323
- unused_supl
 - scsi_read_dvd_struct_data_spare_area, 323
- user_ccb_queue
 - targ_softc, 418
- valid
 - ccb_trans_settings, 110
 - ccb_trans_settings_fc, 111
 - ccb_trans_settings_sas, 112
 - ccb_trans_settings_scsi, 113
 - ccb_trans_settings_spi, 114
- VEncLen
 - SesEncHdr, 403
- vendor
 - scsi_inquiry_data, 231
 - scsi_inquiry_pattern, 234
 - scsi_static_inquiry_pattern, 379
- vendor_id
 - scsi_read_dvd_struct_data_generic_dcb, 310
- vendor_specific
 - scsi_format_unit, 225
- vendor_specific0
 - scsi_inquiry_data, 232
- vendor_specific1
 - scsi_inquiry_data, 232
- VERIFY
 - scsi_da.h, 783
- version
 - scsi_inquiry_data, 232
- version1
 - scsi_inquiry_data, 232
- version2
 - scsi_inquiry_data, 232
- version3
 - scsi_inquiry_data, 232
- version4
 - scsi_inquiry_data, 232
- version5
 - scsi_inquiry_data, 232
- version6
 - scsi_inquiry_data, 232
- version7
 - scsi_inquiry_data, 232

- scsi_inquiry_data, 232
- version8
 - scsi_inquiry_data, 232
- version_num
 - ccb_pathinq, 86
- vif
 - volume_tag, 424
- vitf
 - scsi_send_volume_tag_parameters, 370
- volume
 - cd_audio_page::port_control, 118
- volume_size
 - ccb_calc_geometry, 56
- volume_tag, 424
 - reserved, 424
 - vif, 424
 - vsn, 424
- vsn
 - volume_tag, 424
- vu_flags
 - ccb_eng_exec, 66
- vuhba_flags
 - ccb_pathinq, 86
- VUHBALEN
 - cam_ccb.h, 453
- W
 - scsi_all.c, 592
- wb_full_ratio
 - scsi_dev_conf_page, 221
- west_digital
 - cam_xpt.c, 568
- work_queue
 - targ_softc, 418
 - targbh_softc, 422
- wrbuf16
 - scsi_ses.c, 982
- wrdelay_time
 - scsi_dev_conf_page, 221
- WRITE_10
 - scsi_all.h, 643
- WRITE_12
 - scsi_all.h, 643
- WRITE_16
 - scsi_all.h, 644
- WRITE_6
 - scsi_all.h, 644
- WRITE_AND_VERIFY
 - scsi_da.h, 783
- WRITE_BUFFER
 - scsi_all.h, 644
- write_comp
 - disk_pages::flexible_disk_page, 149
- WRITE_FILEMARKS
 - scsi_sa.h, 947
- write_prot_status
 - read_dvd_struct_write_prot, 189
- write_retry_count
 - scsi_da_rw_recovery_page, 214
- writespeed
 - scsi_set_speed, 376
- wrslot_stat
 - scsi_ses.c, 982
- wwnn
 - ccb_pathinq_settings_fc, 88
 - ccb_trans_settings_fc, 111
- wwpn
 - ccb_pathinq_settings_fc, 88
 - ccb_trans_settings_fc, 111
- xfer_len
 - scsi_play_10, 276
 - scsi_play_12, 277
 - scsi_play_rel_12, 280
 - scsi_send_receive, 367
- xfr_rate_0
 - disk_pages::flexible_disk_page, 149
- xfr_rate_1
 - disk_pages::flexible_disk_page, 149
- XPORT_ATA
 - cam_ccb.h, 456
- XPORT_FC
 - cam_ccb.h, 456
- XPORT_PPB
 - cam_ccb.h, 456
- XPORT_SAS
 - cam_ccb.h, 456
- xport_specific
 - ccb_pathinq, 86
 - ccb_trans_settings, 110
- XPORT_SPI
 - cam_ccb.h, 456
- XPORT_SSA
 - cam_ccb.h, 456
- XPORT_UNKNOWN
 - cam_ccb.h, 456
- XPORT_UNSPECIFIED
 - cam_ccb.h, 456
- XPORT_USB
 - cam_ccb.h, 456
- XPORT_VERSION_UNKNOWN
 - cam_ccb.h, 453
- XPORT_VERSION_UNSPECIFIED
 - cam_ccb.h, 453
- XPT_ABORT
 - cam_ccb.h, 461
- XPT_ACCEPT_TARGET_IO
 - cam_ccb.h, 461

- xpt_action
 - cam_xpt.c, [532](#)
 - cam_xpt.h, [572](#)
- xpt_add_periph
 - cam_xpt.c, [534](#)
 - cam_xpt_periph.h, [580](#)
- xpt_alloc_ccb
 - cam_xpt.c, [534](#)
 - cam_xpt_periph.h, [580](#)
- xpt_alloc_ccb_nowait
 - cam_xpt.c, [535](#)
 - cam_xpt_periph.h, [580](#)
- xpt_alloc_device
 - cam_xpt.c, [535](#)
- xpt_alloc_target
 - cam_xpt.c, [535](#)
- xpt_announce_periph
 - cam_xpt.c, [536](#)
 - cam_xpt_periph.h, [580](#)
- xpt_async
 - cam_xpt.c, [536](#)
 - cam_xpt.h, [574](#)
- xpt_async_bcast
 - cam_xpt.c, [537](#)
- xpt_bus_deregister
 - cam_xpt.c, [537](#)
 - cam_xpt_sim.h, [584](#)
- xpt_bus_register
 - cam_xpt.c, [537](#)
 - cam_xpt_sim.h, [585](#)
- xpt_busfunc_t
 - cam_xpt.c, [524](#)
- XPT_CALC_GEOMETRY
 - cam_ccb.h, [461](#)
- XPT_CCB_INVALID
 - cam_ccb.h, [453](#)
- xpt_cdevsw
 - cam_xpt.c, [568](#)
- xpt_compile_path
 - cam_xpt.c, [538](#)
- xpt_config
 - cam_xpt.c, [538](#)
- xpt_config_hook
 - cam_xpt.c, [568](#)
- XPT_CONT_TARGET_IO
 - cam_ccb.h, [461](#)
- xpt_create_path
 - cam_xpt.c, [539](#)
 - cam_xpt.h, [575](#)
- XPT_DEBUG
 - cam_ccb.h, [460](#)
- XPT_DEPTH_BUS
 - cam_xpt.c, [525](#)
- XPT_DEPTH_DEVICE
 - cam_xpt.c, [525](#)
- XPT_DEPTH_PERIPH
 - cam_xpt.c, [525](#)
- XPT_DEPTH_TARGET
 - cam_xpt.c, [525](#)
- xpt_dev_async
 - cam_xpt.c, [539](#)
- xpt_dev_ccbq_resize
 - cam_xpt.c, [540](#)
- XPT_DEV_MATCH
 - cam_ccb.h, [460](#)
- xpt_devicefunc_t
 - cam_xpt.c, [524](#)
- xpt_devise_transport
 - cam_xpt.c, [540](#)
- xpt_done
 - cam_xpt.c, [541](#)
 - cam_xpt_sim.h, [585](#)
- XPT_EN_LUN
 - cam_ccb.h, [461](#)
- XPT_ENG_EXEC
 - cam_ccb.h, [461](#)
- XPT_ENG_INQ
 - cam_ccb.h, [461](#)
- XPT_FC_DEV_QUEUED
 - cam_ccb.h, [460](#)
- XPT_FC_GROUP
 - cam_ccb.h, [454](#)
- XPT_FC_GROUP_COMMON
 - cam_ccb.h, [454](#)
- XPT_FC_GROUP_HBA_ENGINE
 - cam_ccb.h, [454](#)
- XPT_FC_GROUP_MASK
 - cam_ccb.h, [454](#)
- XPT_FC_GROUP_SCSI_CONTROL
 - cam_ccb.h, [454](#)
- XPT_FC_GROUP_TMODE
 - cam_ccb.h, [454](#)
- XPT_FC_GROUP_VENDOR_UNIQUE
 - cam_ccb.h, [454](#)
- XPT_FC_IS_DEV_QUEUED
 - cam_ccb.h, [454](#)
- XPT_FC_IS_QUEUED
 - cam_ccb.h, [454](#)
- XPT_FC_QUEUED
 - cam_ccb.h, [460](#)
- XPT_FC_USER_CCB
 - cam_ccb.h, [460](#)
- XPT_FC_XPT_ONLY
 - cam_ccb.h, [460](#)
- xpt_find_bus
 - cam_xpt.c, [541](#)
- xpt_find_device
 - cam_xpt.c, [541](#)

- xpt_find_quirk
 - cam_xpt.c, 541
- xpt_find_target
 - cam_xpt.c, 541
- xpt_finishconfig
 - cam_xpt.c, 541
- XPT_FLAG_OPEN
 - cam_xpt.c, 525
- xpt_flags
 - cam_xpt.c, 525
- xpt_for_all_busses
 - cam_xpt.c, 542
- xpt_for_all_devices
 - cam_xpt.c, 542
- xpt_free_ccb
 - cam_xpt.c, 542
 - cam_xpt_periph.h, 581
- xpt_free_path
 - cam_xpt.c, 543
 - cam_xpt.h, 575
- xpt_freeze_devq
 - cam_xpt.c, 543
 - cam_xpt_sim.h, 586
- xpt_freeze_simq
 - cam_xpt.c, 543
 - cam_xpt_sim.h, 586
- XPT_GDEV_STATS
 - cam_ccb.h, 460
- XPT_GDEV_TYPE
 - cam_ccb.h, 460
- XPT_GDEVLIST
 - cam_ccb.h, 460
- xpt_get_ccb
 - cam_xpt.c, 543
- XPT_GET_TRAN_SETTINGS
 - cam_ccb.h, 461
- XPT_IMMED_NOTIFY
 - cam_ccb.h, 461
- xpt_init
 - cam_xpt.c, 543
- xpt_links
 - ccb_hdr, 79
- xpt_merge_ccb
 - cam_xpt.c, 544
 - cam_xpt.h, 576
- XPT_NOOP
 - cam_ccb.h, 460
- XPT_NOTIFY_ACK
 - cam_ccb.h, 461
- xpt_opcode
 - cam_ccb.h, 460
- xpt_path_comp
 - cam_xpt.c, 544
 - cam_xpt.h, 576
- XPT_PATH_INQ
 - cam_ccb.h, 460
- xpt_path_lun_id
 - cam_xpt.c, 544
 - cam_xpt.h, 576
- xpt_path_path_id
 - cam_xpt.c, 545
 - cam_xpt.h, 576
- xpt_path_periph
 - cam_xpt.c, 545
 - cam_xpt.h, 576
- xpt_path_sim
 - cam_xpt.c, 545
 - cam_xpt.h, 576
- XPT_PATH_STATS
 - cam_ccb.h, 460
- xpt_path_string
 - cam_xpt.c, 545
 - cam_xpt.h, 577
- xpt_path_target_id
 - cam_xpt.c, 545
 - cam_xpt.h, 577
- xpt_pdrvfunc_t
 - cam_xpt.c, 524
- xpt_periph
 - cam_periph.h, 492
- xpt_periph_init
 - cam_xpt.c, 545
- xpt_periphfunc_t
 - cam_xpt.c, 524
- xpt_polled_action
 - cam_xpt.c, 545
 - cam_xpt_periph.h, 581
- xpt_print
 - cam_xpt.c, 546
 - cam_xpt.h, 577
- xpt_print_path
 - cam_xpt.c, 546
 - cam_xpt.h, 577
- xpt_quirk_entry, 425
 - inq_pat, 425
 - maxtags, 425
 - mintags, 425
 - quirks, 425
- xpt_quirk_table
 - cam_xpt.c, 568
- xpt_quirk_table_size
 - cam_xpt.c, 569
- XPT_REL_SIMQ
 - cam_ccb.h, 460
- xpt_release_bus
 - cam_xpt.c, 546
- xpt_release_ccb
 - cam_xpt.c, 546

- cam_xpt_periph.h, 581
- xpt_release_device
 - cam_xpt.c, 547
- xpt_release_devq
 - cam_xpt.c, 547
 - cam_xpt_sim.h, 586
- xpt_release_devq_device
 - cam_xpt.c, 548
- xpt_release_devq_timeout
 - cam_xpt.c, 548, 569
- xpt_release_path
 - cam_xpt.c, 548
- xpt_release_simq
 - cam_xpt.c, 549
 - cam_xpt_sim.h, 586
- xpt_release_simq_timeout
 - cam_xpt.c, 549, 569
- xpt_release_target
 - cam_xpt.c, 550
- xpt_remove_periph
 - cam_xpt.c, 550
 - cam_xpt_periph.h, 582
- xpt_rescan
 - cam_xpt.c, 550
 - cam_xpt.h, 577
- XPT_RESET_BUS
 - cam_ccb.h, 461
- XPT_RESET_DEV
 - cam_ccb.h, 461
- xpt_run_dev_allocq
 - cam_xpt.c, 550
- xpt_run_dev_sendq
 - cam_xpt.c, 551
- XPT_SASYNC_CB
 - cam_ccb.h, 460
- XPT_SCAN_BUS
 - cam_ccb.h, 460
- xpt_scan_bus
 - cam_xpt.c, 551
- xpt_scan_bus_info, 426
 - counter, 426
 - cpu, 426
 - request_ccb, 426
- XPT_SCAN_LUN
 - cam_ccb.h, 461
- xpt_scan_lun
 - cam_xpt.c, 552
- xpt_schedule
 - cam_xpt.c, 553
 - cam_xpt_periph.h, 582
- xpt_schedule_dev
 - cam_xpt.c, 553
- xpt_schedule_dev_allocq
 - cam_xpt.c, 554
- xpt_schedule_dev_sendq
 - cam_xpt.c, 554
- XPT SCSI_IO
 - cam_ccb.h, 460
- XPT_SDEV_TYPE
 - cam_ccb.h, 460
- XPT_SET_TRAN_SETTINGS
 - cam_ccb.h, 461
- xpt_set_transfer_settings
 - cam_xpt.c, 554
- xpt_setup_ccb
 - cam_xpt.c, 555
 - cam_xpt.h, 578
- xpt_softc, 427
 - flags, 427
 - generation, 427
- xpt_start_tags
 - cam_xpt.c, 556
- XPT_TARGET_IO
 - cam_ccb.h, 461
- xpt_targetfunc_t
 - cam_xpt.c, 524
- XPT_TERM_IO
 - cam_ccb.h, 461
- xpt_toggle_tags
 - cam_xpt.c, 556
- xpt_traverse_config, 428
 - depth, 428
 - tr_arg, 428
 - tr_func, 428
- xpt_traverse_depth
 - cam_xpt.c, 525
- XPT_VUNIQUE
 - cam_ccb.h, 461
- xptaction
 - cam_xpt.c, 557
- xptbusmatch
 - cam_xpt.c, 557
- xptbustraverse
 - cam_xpt.c, 557
- xptclose
 - cam_xpt.c, 557, 569
- xptconfigbuscountfunc
 - cam_xpt.c, 558, 569
- xptconfigfunc
 - cam_xpt.c, 558, 569
- xptdefbusfunc
 - cam_xpt.c, 558, 569
- xptdefdevicefunc
 - cam_xpt.c, 559, 569
- xptdefperiphfunc
 - cam_xpt.c, 559, 569
- xptdeftargetfunc
 - cam_xpt.c, 559, 569

- xptdevicematch
 - cam_xpt.c, [559](#)
- xptdevicetraverse
 - cam_xpt.c, [560](#)
- xptdone
 - cam_xpt.c, [560](#)
- xptedtbusfunc
 - cam_xpt.c, [560](#), [569](#)
- xptedtdevicefunc
 - cam_xpt.c, [560](#), [570](#)
- xptedtmatch
 - cam_xpt.c, [561](#)
- xptedtperiphfunc
 - cam_xpt.c, [561](#), [570](#)
- xptedttargetfunc
 - cam_xpt.c, [562](#), [570](#)
- xptioctl
 - cam_xpt.c, [562](#), [570](#)
- xptnextfreepathid
 - cam_xpt.c, [563](#)
- xptopen
 - cam_xpt.c, [563](#), [570](#)
- xptpassannouncefunc
 - cam_xpt.c, [563](#), [570](#)
- xptpathid
 - cam_xpt.c, [563](#)
- xptpdperiphtraverse
 - cam_xpt.c, [564](#)
- xptdrvtraverse
 - cam_xpt.c, [564](#)
- xptperiphlistmatch
 - cam_xpt.c, [564](#)
- xptperiphmatch
 - cam_xpt.c, [564](#)
- xptperiphtraverse
 - cam_xpt.c, [565](#)
- xptlistdrvfunc
 - cam_xpt.c, [565](#), [570](#)
- xptlistperiphfunc
 - cam_xpt.c, [565](#), [570](#)
- xtpoll
 - cam_xpt.c, [565](#)
- xptregister
 - cam_xpt.c, [565](#)
- xptscandone
 - cam_xpt.c, [566](#)
- xptsetasynbusfunc
 - cam_xpt.c, [566](#), [570](#)
- xptsetasynfunc
 - cam_xpt.c, [566](#), [570](#)
- xpttargettraverse
 - cam_xpt.c, [567](#)
- xsoftc
 - cam_xpt.c, [570](#)
- year
 - scsi_read_dvd_struct_data_disc_id, [307](#)
- zeros0
 - scsi_read_dvd_struct_data_layer_desc, [313](#)
- zeros1
 - scsi_read_dvd_struct_data_layer_desc, [313](#)
- zeros2
 - scsi_read_dvd_struct_data_layer_desc, [313](#)