Space Shuttle Ultra

1.20a

Generated by Doxygen 1.5.5

Thu Jul 10 14:04:26 2008

Contents

1 Deprecated List **1**

2 Namespace Index **3**

2.1 Namespace List 3

3 Class Index **5**

3.1 Class Hierarchy 5

4 Class Index **7**

4.1 Class List 7

5 File Index **9**

5.1 File List 9

6 Namespace Documentation **13**

6.1 discsignals Namespace Reference 13

6.2 dps Namespace Reference 14

6.3 eps Namespace Reference 16

6.4 eva_docking Namespace Reference 17

6.5 gnc Namespace Reference 18

6.6 hydraulics Namespace Reference 19

6.7 mission Namespace Reference 20

6.8 mps Namespace Reference 21

6.9 vc Namespace Reference 24

7 Class Documentation **27**

7.1 mps::_DATA_TABLE Struct Reference 27

7.2 gnc::AccAssembly Class Reference 29

7.3 AirDataProbeSystem Class Reference 31

7.4 dps::AP101S Class Reference 35

7.5 Atlantis Class Reference 37
7.6 Atlantis_SRB Class Reference 68
7.7 Atlantis_Tank Class Reference 70
7.8 AtlantisAPUHYD Class Reference 71
7.9 AtlantisConfig Class Reference 74
7.10 AtlantisSubsystem Class Reference 77
7.11 mps::ATVC Class Reference 83
7.12 eva_docking::BasicExternalAirlock Class Reference 85
7.13 vc::BasicLight Class Reference 89
7.14 vc::BasicPanel Class Reference 91
7.15 vc::BasicSwitch Class Reference 97
7.16 BasicValve Class Reference 100
7.17 vc::BasicVCComponent Class Reference 102
7.18 dps::BIU Class Reference 110
7.19 mps::BLOCK_II Class Reference 112
7.20 CommModeHandler Class Reference 116
7.21 CRT Class Reference 120
7.22 CRT::SavePrm Struct Reference 132
7.23 DAPConfig Struct Reference 133
7.24 dps::DEUCHAR Struct Reference 135
7.25 discsignals::DiscInPort Class Reference 136
7.26 discsignals::DiscOutPort Class Reference 138
7.27 discsignals::DiscreteBundle Class Reference 140
7.28 discsignals::DiscreteBundleManager Class Reference 142
7.29 discsignals::DiscretePort Class Reference 144
7.30 mps::EIU Class Reference 146
7.31 vc::EventTimer Class Reference 150
7.32 eva_docking::ExtAirlock Class Reference 152
7.33 FLOWSTATE Struct Reference 154
7.34 GDIParams Struct Reference 155
7.35 GearOp Class Reference 156
7.36 hydraulics::HydActuator Class Reference 161
7.37 eps::IACElement Class Reference 163
7.38 eps::IDCElement Class Reference 164
7.39 dps::IDP Class Reference 165
7.40 dps::IDPSoftware Class Reference 170

7.41 vc::Indicator7Seg Class Reference . . . 172
7.42 ISubsystem Class Reference . . . 174
7.43 Keyboard Class Reference . . . 177
7.44 KeyboardInput Struct Reference . . . 179
7.45 KuBandSystem Class Reference . . . 180
7.46 vc::LockableLever Class Reference . . . 182
7.47 dps::MasterTimingUnit Class Reference . . . 184
7.48 dps::MDM Class Reference . . . 190
7.49 vc::MDU Class Reference . . . 192
7.50 MechActuator Class Reference . . . 199
7.51 MFD_GDIPARAM Struct Reference . . . 202
7.52 mission::Mission Class Reference . . . 203
7.53 MLP Class Reference . . . 205
7.54 mps::MPS Class Reference . . . 208
7.55 eva_docking::ODS Class Reference . . . 213
7.56 OMSPod Class Reference . . . 215
7.57 OMSSubsystem Class Reference . . . 220
7.58 PanelA4 Class Reference . . . 222
7.59 vc::PanelA7A8ODS Class Reference . . . 225
7.60 PanelA8 Class Reference . . . 231
7.61 PanelC2 Class Reference . . . 234
7.62 PanelC3 Class Reference . . . 237
7.63 vc::PanelF6 Class Reference . . . 240
7.64 vc::PanelF7 Class Reference . . . 243
7.65 PanelF7 Class Reference . . . 245
7.66 vc::PanelF8 Class Reference . . . 247
7.67 vc::PanelGroup Class Reference . . . 250
7.68 PanelO3 Class Reference . . . 252
7.69 PanelR2 Class Reference . . . 255
7.70 PayloadBayOp Class Reference . . . 262
7.71 dps::RAM16 Class Reference . . . 270
7.72 RefPoints Struct Reference . . . 271
7.73 vc::RussianCB Class Reference . . . 272
7.74 dps::ShuttleBus Class Reference . . . 274
7.75 mps::SSME Class Reference . . . 275
7.76 SSU_FSS_RSS Class Reference . . . 284

7.77 vc::StandardSwitch Class Reference 287
7.78 vc::StdSwitch2 Class Reference 290
7.79 vc::StdSwitch3 Class Reference 292
7.80 Stopwatch Class Reference 294
7.81 SubsystemDirector Class Reference 295
7.82 ValveTypeBool Class Reference 297
7.83 vc::VCMOUSEEVENTPOINT Struct Reference 299

8 File Documentation **301**

8.1 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AC_resource.h File Reference 301
8.2 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AirDataProbeSystem.cpp File Reference . . 302
8.3 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AirDataProbeSystem.h File Reference . . . 303
8.4 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Atlantis.cpp File Reference 304
8.5 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Atlantis.h File Reference 311
8.6 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Atlantis_SRB.cpp File Reference 331
8.7 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Atlantis_Tank.cpp File Reference 332
8.8 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AtlantisAPUHYD.cpp File Reference . . . 334
8.9 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AtlantisAPUHYD.h File Reference 335
8.10 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AtlantisConfig.cpp File Reference 337
8.11 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AtlantisSubsystem.cpp File Reference . . . 339
8.12 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AtlantisSubsystem.h File Reference 340
8.13 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/BasicValve.cpp File Reference 341
8.14 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/BasicValve.h File Reference 342
8.15 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/CommModeHandler.cpp File Reference . . 343
8.16 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/CommModeHandler.h File Reference . . . 344
8.17 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Common.cpp File Reference 345
8.18 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/CRT.cpp File Reference 348
8.19 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/CRT.h File Reference 353
8.20 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscInPort.cpp File Reference . 355
8.21 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscInPort.h File Reference . . 356
8.22 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscOutPort.cpp File Reference 357
8.23 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscOutPort.h File Reference . 358
8.24 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscreteBundle.cpp File Reference 359
8.25 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscreteBundle.h File Reference 360
8.26 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscreteBundleManager.cpp File Reference 361

8.27 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscreteBundleManager.h File Reference . . . 362
8.28 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscretePort.cpp File Reference 363
8.29 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscretePort.h File Reference . 364
8.30 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/discsignals.h File Reference . . 365
8.31 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/AP101S.cpp File Reference 366
8.32 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/AP101S.h File Reference 367
8.33 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/BIU.h File Reference 368
8.34 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/dps_defs.h File Reference 369
8.35 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/GPCVariables.h File Reference 370
8.36 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/IDP.cpp File Reference 371
8.37 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/IDP.h File Reference 372
8.38 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/MasterTimingUnit.cpp File Reference . 374
8.39 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/MasterTimingUnit.h File Reference . . 375
8.40 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/MDM.cpp File Reference 377
8.41 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/MDM.h File Reference 378
8.42 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/RAM16.h File Reference 379
8.43 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/ShuttleBus.cpp File Reference 380
8.44 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/ShuttleBus.h File Reference 381
8.45 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/eps/eps_defs.h File Reference 382
8.46 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/eva_docking/BasicExtAirlock.cpp File Reference 383
8.47 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/eva_docking/BasicExtAirlock.h File Reference 384
8.48 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/eva_docking/ExtAirlock.cpp File Reference 385
8.49 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/eva_docking/ExtAirlock.h File Reference . 386
8.50 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/eva_docking/ODS.h File Reference 387
8.51 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/FSSRSS/meshres_FSS_RSS.h File Reference 388
8.52 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/FSSRSS/SSU_FSS_RSS.cpp File Reference 395
8.53 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/FSSRSS/SSU_FSS_RSS.h File Reference . 397
8.54 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/GearOp.cpp File Reference 398
8.55 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/GearOp.h File Reference 399
8.56 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/gnc/AA.h File Reference 400
8.57 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/GPC.cpp File Reference 401
8.58 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/HALFC/HALFC.cpp File Reference 403
8.59 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/HALFC/StdAfx.cpp File Reference 404
8.60 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/HALFC/StdAfx.h File Reference 405

8.61 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/hydraulics/HydActuator.h File Reference . 406
8.62 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/ISubsystem.h File Reference 407
8.63 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Keyboard.cpp File Reference 408
8.64 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Keyboard.h File Reference 409
8.65 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/KuBandSystem.cpp File Reference 410
8.66 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/KuBandSystem.h File Reference 411
8.67 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/MechActuator.h File Reference 412
8.68 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/meshres.h File Reference 413
8.69 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/meshres_ku.h File Reference 417
8.70 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/meshres_odsvc.h File Reference 418
8.71 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/meshres_RMS.h File Reference 423
8.72 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/meshres_vc.h File Reference 425
8.73 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/meshres_vc_additions.h File Reference . . 443
8.74 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mission/Mission.h File Reference 447
8.75 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/MLP/MLP.cpp File Reference 448
8.76 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/MLP/MLP.h File Reference 449
8.77 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/ATVC.cpp File Reference 450
8.78 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/ATVC.h File Reference 451
8.79 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/BLOCK_II.cpp File Reference 452
8.80 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/BLOCK_II.h File Reference 453
8.81 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/EIU.cpp File Reference 454
8.82 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/EIU.h File Reference 455
8.83 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/MPS.cpp File Reference 456
8.84 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/MPS.h File Reference 457
8.85 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/SSME.cpp File Reference 459
8.86 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/SSME.h File Reference 460
8.87 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/OMSPod.cpp File Reference 462
8.88 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/OMSPod.h File Reference 463
8.89 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/OMSSubsystem.cpp File Reference 465
8.90 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/OMSSubsystem.h File Reference 466
8.91 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelA4.cpp File Reference 467
8.92 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelA4.h File Reference 468
8.93 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelA8.cpp File Reference 469
8.94 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelA8.h File Reference 470
8.95 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelC2.cpp File Reference 471
8.96 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelC2.h File Reference 473

8.97 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelC3.cpp File Reference 475
8.98 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelC3.h File Reference 476
8.99 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelF7.cpp File Reference 478
8.100 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelF7.cpp File Reference 479
8.101 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelF7.h File Reference 480
8.102 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelF7.h File Reference 481
8.103 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelO3.cpp File Reference 482
8.104 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelO3.h File Reference 483
8.105 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelR2.cpp File Reference 485
8.106 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelR2.h File Reference 486
8.107 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PlBayOp.cpp File Reference 487
8.108 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PlBayOp.h File Reference 489
8.109 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/resource.h File Reference 490
8.110 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/resource_RMS.h File Reference 494
8.111 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/SubsystemDirector.cpp File Reference . . . 495
8.112 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/SubsystemDirector.h File Reference 496
8.113 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/util/Stopwatch.cpp File Reference 497
8.114 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/util/Stopwatch.h File Reference 498
8.115 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/ValveTypeBool.cpp File Reference 499
8.116 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/ValveTypeBool.h File Reference 500
8.117 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicLight.h File Reference 501
8.118 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicPanel.cpp File Reference 502
8.119 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicPanel.h File Reference 503
8.120 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicSwitch.cpp File Reference 504
8.121 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicSwitch.h File Reference 505
8.122 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicVCComponent.cpp File Reference 506
8.123 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicVCComponent.h File Reference . . 507
8.124 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/EventTimer.h File Reference 508
8.125 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/Indicator7Seg.cpp File Reference 509
8.126 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/Indicator7Seg.h File Reference 510
8.127 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/MDU.cpp File Reference 511
8.128 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/MDU.h File Reference 512
8.129 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelA7A8ODS.cpp File Reference . . 513
8.130 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelA7A8ODS.h File Reference 514
8.131 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelF6.cpp File Reference 515
8.132 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelF6.h File Reference 516

8.133 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelF8.cpp File Reference 517
8.134 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelF8.h File Reference 518
8.135 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelGroup.cpp File Reference 519
8.136 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelGroup.h File Reference 520
8.137 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/StandardSwitch.cpp File Reference . . . 521
8.138 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/StandardSwitch.h File Reference 522
8.139 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/vc_defs.h File Reference 524

Chapter 1

Deprecated List

Member AtlantisSubsystem::AddMeshes(const VECTOR3 &ofs) Don't use, instead, use Realize() to tell the Atlantis class which meshes it should use for special slots.

Chapter 2

Namespace Index

2.1 Namespace List

Here is a list of all namespaces with brief descriptions:

discsignals 13
dps 14
eps 16
eva_docking 17
gnc 18
hydraulics 19
mission 20
mps 21
vc 24

Chapter 3

Class Index

3.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

mps::_DATA_TABLE 27
Atlantis 37
Atlantis_SRB 68
Atlantis_Tank 70
AtlantisConfig 74
vc::BasicPanel 91
  vc::PanelA7A8ODS 225
  vc::PanelF6 240
  vc::PanelF7 243
  vc::PanelF8 247
BasicValve 100
dps::BIU 110
CommModeHandler 116
CRT 120
CRT::SavePrm 132
DAPConfig 133
dps::DEUCHAR 135
discsignals::DiscreteBundle 140
discsignals::DiscreteBundleManager 142
discsignals::DiscretePort 144
  discsignals::DiscInPort 136
  discsignals::DiscOutPort 138
FLOWSTATE 154
GDIParams 155
GearOp 156
hydraulics::HydActuator 161
eps::IACElement 163
eps::IDCElement 164
dps::IDPSoftware 170
ISubsystem 174
  AtlantisSubsystem 77
    AirDataProbeSystem 31

AtlantisAPUHYD . . . 71
dps::AP101S . . . 35
dps::IDP . . . 165
dps::MasterTimingUnit . . . 184
dps::MDM . . . 190
eva_docking::BasicExternalAirlock . . . 85
eva_docking::ExtAirlock . . . 152
eva_docking::ODS . . . 213
gnc::AccAssembly . . . 29
KuBandSystem . . . 180
MechActuator . . . 199
mps::ATVC . . . 83
mps::EIU . . . 146
mps::MPS . . . 208
mps::SSME . . . 275
mps::BLOCK_II . . . 112
OMSSubsystem . . . 220
vc::BasicVCComponent . . . 102
vc::BasicLight . . . 89
vc::BasicSwitch . . . 97
vc::LockableLever . . . 182
vc::StandardSwitch . . . 287
vc::RussianCB . . . 272
vc::StdSwitch2 . . . 290
vc::StdSwitch3 . . . 292
vc::EventTimer . . . 150
vc::Indicator7Seg . . . 172
vc::MDU . . . 192
Keyboard . . . 177
KeyboardInput . . . 179
MFD_GDIPARAM . . . 202
mission::Mission . . . 203
MLP . . . 205
OMSPod . . . 215
PanelA4 . . . 222
PanelA8 . . . 231
PanelC2 . . . 234
PanelC3 . . . 237
PanelF7 . . . 245
vc::PanelGroup . . . 250
PanelO3 . . . 252
PanelR2 . . . 255
PayloadBayOp . . . 262
dps::RAM16 . . . 270
RefPoints . . . 271
dps::ShuttleBus . . . 274
SSU_FSS_RSS . . . 284
Stopwatch . . . 294
SubsystemDirector . . . 295
ValveTypeBool . . . 297
vc::VCMOUSEEVENTPOINT . . . 299

Chapter 4

Class Index

4.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

mps::_DATA_TABLE 27
gnc::AccAssembly 29
AirDataProbeSystem 31
dps::AP101S 35
Atlantis 37
Atlantis_SRB 68
Atlantis_Tank 70
AtlantisAPUHYD 71
AtlantisConfig 74
AtlantisSubsystem 77
mps::ATVC 83
eva_docking::BasicExternalAirlock 85
vc::BasicLight 89
vc::BasicPanel 91
vc::BasicSwitch 97
BasicValve 100
vc::BasicVCComponent 102
dps::BIU 110
mps::BLOCK_II 112
CommModeHandler 116
CRT 120
CRT::SavePrm 132
DAPConfig 133
dps::DEUCHAR 135
discsignals::DiscInPort 136
discsignals::DiscOutPort 138
discsignals::DiscreteBundle 140
discsignals::DiscreteBundleManager 142
discsignals::DiscretePort 144
mps::EIU 146
vc::EventTimer 150
eva_docking::ExtAirlock 152
FLOWSTATE 154

GDIParams . . . 155
GearOp . . . 156
hydraulics::HydActuator . . . 161
eps::IACElement . . . 163
eps::IDCElement . . . 164
dps::IDP . . . 165
dps::IDPSoftware . . . 170
vc::Indicator7Seg . . . 172
ISubsystem . . . 174
Keyboard . . . 177
KeyboardInput . . . 179
KuBandSystem . . . 180
vc::LockableLever . . . 182
dps::MasterTimingUnit . . . 184
dps::MDM . . . 190
vc::MDU . . . 192
MechActuator . . . 199
MFD_GDIPARAM . . . 202
mission::Mission . . . 203
MLP . . . 205
mps::MPS . . . 208
eva_docking::ODS . . . 213
OMSPod . . . 215
OMSSubsystem . . . 220
PanelA4 . . . 222
vc::PanelA7A8ODS . . . 225
PanelA8 . . . 231
PanelC2 . . . 234
PanelC3 . . . 237
vc::PanelF6 . . . 240
vc::PanelF7 . . . 243
PanelF7 . . . 245
vc::PanelF8 . . . 247
vc::PanelGroup . . . 250
PanelO3 . . . 252
PanelR2 . . . 255
PayloadBayOp . . . 262
dps::RAM16 . . . 270
RefPoints . . . 271
vc::RussianCB . . . 272
dps::ShuttleBus . . . 274
mps::SSME . . . 275
SSU_FSS_RSS . . . 284
vc::StandardSwitch . . . 287
vc::StdSwitch2 . . . 290
vc::StdSwitch3 . . . 292
Stopwatch . . . 294
SubsystemDirector . . . 295
ValveTypeBool . . . 297
vc::VCMOUSEEVENTPOINT . . . 299

Chapter 5

File Index

5.1 File List

Here is a list of all files with brief descriptions:

G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AC_resource.h 301
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AirDataProbeSystem.cpp 302
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AirDataProbeSystem.h 303
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Atlantis.cpp 304
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Atlantis.h 311
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Atlantis_SRB.cpp 331
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Atlantis_Tank.cpp 332
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AtlantisAPUHYD.cpp 334
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AtlantisAPUHYD.h 335
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AtlantisConfig.cpp 337
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AtlantisSubsystem.cpp 339
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AtlantisSubsystem.h 340
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/BasicValve.cpp 341
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/BasicValve.h 342
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/CommModeHandler.cpp 343
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/CommModeHandler.h 344
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Common.cpp 345
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/CRT.cpp 348
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/CRT.h 353
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/GearOp.cpp 398
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/GearOp.h 399
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/GPC.cpp 401
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/ISubsystem.h 407
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Keyboard.cpp 408
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Keyboard.h 409
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/KuBandSystem.cpp 410
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/KuBandSystem.h 411
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/MechActuator.h 412
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/meshres.h 413
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/meshres_ku.h 417
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/meshres_odsvc.h 418
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/meshres_RMS.h 423
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/meshres_vc.h 425

G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/meshres_vc_additions.h 443
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/OMSPod.cpp 462
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/OMSPod.h 463
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/OMSSubsystem.cpp 465
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/OMSSubsystem.h 466
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelA4.cpp 467
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelA4.h 468
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelA8.cpp 469
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelA8.h 470
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelC2.cpp 471
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelC2.h 473
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelC3.cpp 475
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelC3.h 476
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelF7.cpp 478
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelF7.h 480
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelO3.cpp 482
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelO3.h 483
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelR2.cpp 485
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelR2.h 486
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PlBayOp.cpp 487
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PlBayOp.h 489
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/resource.h 490
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/resource_RMS.h 494
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/SubsystemDirector.cpp 495
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/SubsystemDirector.h 496
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/ValveTypeBool.cpp 499
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/ValveTypeBool.h 500
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscInPort.cpp 355
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscInPort.h 356
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscOutPort.cpp 357
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscOutPort.h 358
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscreteBundle.cpp 359
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscreteBundle.h 360
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscreteBundleManager.cpp 361
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscreteBundleManager.h 362
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscretePort.cpp 363
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscretePort.h 364
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/discsignals.h 365
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/AP101S.cpp 366
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/AP101S.h 367
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/BIU.h 368
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/dps_defs.h 369
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/GPCVariables.h 370
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/IDP.cpp 371
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/IDP.h 372
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/MasterTimingUnit.cpp 374
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/MasterTimingUnit.h 375
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/MDM.cpp 377
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/MDM.h 378
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/RAM16.h 379
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/ShuttleBus.cpp 380
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/ShuttleBus.h 381
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/eps/eps_defs.h 382
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/eva_docking/BasicExtAirlock.cpp 383

G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/eva_docking/BasicExtAirlock.h 384
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/eva_docking/ExtAirlock.cpp 385
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/eva_docking/ExtAirlock.h 386
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/eva_docking/ODS.h 387
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/FSSRSS/meshres_FSS_RSS.h 388
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/FSSRSS/SSU_FSS_RSS.cpp 395
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/FSSRSS/SSU_FSS_RSS.h 397
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/gnc/AA.h 400
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/HALFC/HALFC.cpp 403
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/HALFC/StdAfx.cpp 404
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/HALFC/StdAfx.h 405
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/hydraulics/HydActuator.h 406
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mission/Mission.h 447
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/MLP/MLP.cpp 448
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/MLP/MLP.h 449
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/ATVC.cpp 450
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/ATVC.h 451
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/BLOCK_II.cpp 452
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/BLOCK_II.h 453
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/EIU.cpp 454
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/EIU.h 455
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/MPS.cpp 456
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/MPS.h 457
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/SSME.cpp 459
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/SSME.h 460
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/util/Stopwatch.cpp 497
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/util/Stopwatch.h 498
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicLight.h 501
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicPanel.cpp 502
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicPanel.h 503
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicSwitch.cpp 504
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicSwitch.h 505
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicVCComponent.cpp 506
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicVCComponent.h 507
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/EventTimer.h 508
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/Indicator7Seg.cpp 509
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/Indicator7Seg.h 510
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/MDU.cpp 511
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/MDU.h 512
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelA7A8ODS.cpp 513
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelA7A8ODS.h 514
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelF6.cpp 515
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelF6.h 516
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelF7.cpp 479
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelF7.h 481
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelF8.cpp 517
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelF8.h 518
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelGroup.cpp 519
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelGroup.h 520
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/StandardSwitch.cpp 521
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/StandardSwitch.h 522
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/vc_defs.h 524

Chapter 6

Namespace Documentation

6.1 discsignals Namespace Reference

Classes

- class DiscInPort
- class DiscOutPort
- class DiscreteBundle
- class DiscreteBundleManager
- class DiscretePort

6.2 dps Namespace Reference

Classes

- class AP101S
- class BIU
- struct DEUCHAR
- class IDPSoftware
- class IDP
- class MasterTimingUnit
- class MDM
- class RAM16
- class ShuttleBus

Typedefs

- typedef short MTU_ACCU_INDEX
- typedef short MTU_EVTTMR_INDEX

Enumerations

- enum MAJORFUNCTION { GNC = 0, SM = 1, PL = 2 }
- enum EVENT_TIMER_MODE { COUNT_UP, COUNT_DOWN, COUNT_TEST }
- enum EVENT_TIMER_CONTROL { COUNT_STOPPED, COUNT_STARTED }

Variables

- const char DEUATT_NORMAL = 0
- const char DEUATT_OVERBRIGHT = 1
- const char DEUATT_FLASHING = 2
- const char DEUATT_UNDERLINED = 4
- const MTU_EVTTMR_INDEX TIMER_FORWARD = 0
- const MTU_EVTTMR_INDEX TIMER_AFT = 1

6.2.1 Typedef Documentation

6.2.1.1 typedef short dps::MTU_ACCU_INDEX

6.2.1.2 typedef short dps::MTU_EVTTMR_INDEX

6.2.2 Enumeration Type Documentation

6.2.2.1 enum dps::EVENT_TIMER_CONTROL

Enumerator:

COUNT_STOPPED

COUNT_STARTED

6.2.2.2 enum dps::EVENT_TIMER_MODE

Enumerator:

COUNT_UP

COUNT_DOWN

COUNT_TEST

6.2.2.3 enum dps::MAJORFUNCTION

Enumerator:

GNC

SM

PL

6.2.3 Variable Documentation

6.2.3.1 const char dps::DEUATT_FLASHING = 2

6.2.3.2 const char dps::DEUATT_NORMAL = 0

6.2.3.3 const char dps::DEUATT_OVERBRIGHT = 1

6.2.3.4 const char dps::DEUATT_UNDERLINED = 4

6.2.3.5 const MTU_EVTTMR_INDEX dps::TIMER_AFT = 1

6.2.3.6 const MTU_EVTTMR_INDEX dps::TIMER_FORWARD = 0

6.3 eps Namespace Reference

Classes

- class IDCElement
- class IACElement

Enumerations

- enum ACPHASE { PHASE1 = 0, PHASE2, PHASE3 }

6.3.1 Enumeration Type Documentation

6.3.1.1 enum eps::ACPHASE

Enumerator:

PHASE1

PHASE2

PHASE3

6.4 eva_docking Namespace Reference

Classes

- class BasicExternalAirlock
- class ExtAirlock
- class ODS

6.4.1 Detailed Description

EVA and Docking systems

6.5 gnc Namespace Reference

Classes

- class AccAssembly

6.6 hydraulics Namespace Reference

Classes

- class HydActuator

6.7 mission Namespace Reference

Classes

- class Mission

6.7.1 Detailed Description

Contains all class definitions related to Mission Data Files and handling mission data in the simulation

6.8 mps Namespace Reference

Classes

- class ATVC
- class BLOCK_II
- class EIU
- class MPS
- struct _DATA_TABLE
- class SSME

Typedefs

- typedef enum mps::_MPS_VALVE MPS_VALVE
- typedef enum mps::_ENGINE_STATUS ENGINE_STATUS
- typedef enum mps::_DCU DCU
- typedef struct mps::_DATA_TABLE DATA_TABLE

Enumerations

- enum _MPS_VALVE {

  PV1, PV2, PV3, PV4,

  PV5, PV6, PV7, PV8,

  PV9, PV10, PV11, PV12,

  PV13, PV17, PV18, PV19,

  PV20, PV21, PV22, PD1,

  PD2, PD3 }
- enum _ENGINE_STATUS {

  CHECKOUT_STANDBY, CHECKOUT_COMPONENTCHECKOUT, STARTPREPARATION_-PURGESEQUENCE1, STARTPREPARATION_PURGESEQUENCE2,

  STARTPREPARATION_PURGESEQUENCE3, STARTPREPARATION_-PURGESEQUENCE4, STARTPREPARATION_ENGINEREADY, STARTMAINSTAGE_-STARTPHASESTARTINITIATION,

  STARTMAINSTAGE_STARTPHASETHRUSTBUILDUP, STARTMAINSTAGE_-MAINSTAGEPHASENORMALCONTROL, STARTMAINSTAGE_FIXEDDENSITY, STARTMAINSTAGE_THRUSTLIMITING,

  STARTMAINSTAGE_ELECTRICALLOCKUP, STARTMAINSTAGE_-HYDRAULICLOCKUP, SHUTDOWN_THROTTLINGTOZEROTHRUST, SHUTDOWN_-PROPELLANTVALVESCLOSED,

  SHUTDOWN_FAILSAFEPNEUMATIC, POSTSHUTDOWN_STANDBY, POSTSHUTDOWN_-OXIDIZERDUMP, POSTSHUTDOWN_TERMINATESEQUENCE,

  PROM_STANDBY }
- enum _DCU { DCU_A, DCU_B }

Variables

- const double BLOCK_II_RPL_THRUST = 2090664.159
- const double BLOCK_II_FPL_THRUST = 2275094.273
- const double BLOCK_II_ISP0 = 453∗9.80665
- const double BLOCK_II_ISP1 = 363∗9.80665

6.8.1 Typedef Documentation

6.8.1.1 typedef struct mps::_DATA_TABLE mps::DATA_TABLE

6.8.1.2 typedef enum mps::_DCU mps::DCU

6.8.1.3 typedef enum mps::_ENGINE_STATUS mps::ENGINE_STATUS

6.8.1.4 typedef enum mps::_MPS_VALVE mps::MPS_VALVE

6.8.2 Enumeration Type Documentation

6.8.2.1 enum mps::_DCU

Enumerator:

DCU_A

DCU_B

6.8.2.2 enum mps::_ENGINE_STATUS

Enumerator:

CHECKOUT_STANDBY

CHECKOUT_COMPONENTCHECKOUT

STARTPREPARATION_PURGESEQUENCE1

STARTPREPARATION_PURGESEQUENCE2

STARTPREPARATION_PURGESEQUENCE3

STARTPREPARATION_PURGESEQUENCE4

STARTPREPARATION_ENGINEREADY

STARTMAINSTAGE_STARTPHASESTARTINITIATION

STARTMAINSTAGE_STARTPHASETHRUSTBUILDUP

STARTMAINSTAGE_MAINSTAGEPHASENORMALCONTROL

STARTMAINSTAGE_FIXEDDENSITY

STARTMAINSTAGE_THRUSTLIMITING

STARTMAINSTAGE_ELECTRICALLOCKUP

STARTMAINSTAGE_HYDRAULICLOCKUP

SHUTDOWN_THROTTLINGTOZEROTHRUST

SHUTDOWN_PROPELLANTVALVESCLOSED

SHUTDOWN_FAILSAFEPNEUMATIC

POSTSHUTDOWN_STANDBY
POSTSHUTDOWN_OXIDIZERDUMP
POSTSHUTDOWN_TERMINATESEQUENCE
PROM_STANDBY

6.8.2.3 enum mps::_MPS_VALVE

Enumerator:

PV1
PV2
PV3
PV4
PV5
PV6
PV7
PV8
PV9
PV10
PV11
PV12
PV13
PV17
PV18
PV19
PV20
PV21
PV22
PD1
PD2
PD3

6.8.3 Variable Documentation

6.8.3.1 const double mps::BLOCK_II_FPL_THRUST = 2275094.273

6.8.3.2 const double mps::BLOCK_II_ISP0 = 453∗9.80665

6.8.3.3 const double mps::BLOCK_II_ISP1 = 363∗9.80665

6.8.3.4 const double mps::BLOCK_II_RPL_THRUST = 2090664.159

6.9 vc Namespace Reference

Classes

- class BasicLight
- class BasicPanel
- class BasicSwitch
- class BasicVCComponent
- class EventTimer
- class Indicator7Seg
- class MDU
- class PanelA7A8ODS
- class PanelF6
- class PanelF7
- class PanelF8
- class PanelGroup
- class LockableLever
- class StandardSwitch
- class StdSwitch2
- class StdSwitch3
- class RussianCB
- struct VCMOUSEEVENTPOINT

Typedefs

- typedef MDU ∗ PMDU

Functions

- void BitmapLocation (char ch, int &x, int &y)

Variables

- const int MDUID_CDR1 = 0
- const int MDUID_CDR2 = 1
- const int MDUID_PLT1 = 2
- const int MDUID_PLT2 = 3
- const int MDUID_CRT1 = 4
- const int MDUID_CRT2 = 5
- const int MDUID_CRT3 = 6
- const int MDUID_CRT4 = 7
- const int MDUID_MFD1 = 8
- const int MDUID_MFD2 = 9
- const int MDUID_AFD = 10

6.9.1 Detailed Description

Define the public contents of the VC namespace.

6.9.2 Typedef Documentation

6.9.2.1 typedef MDU∗ vc::PMDU

6.9.3 Function Documentation

6.9.3.1 void vc::BitmapLocation (char *ch*, int & *x*, int & *y*)

6.9.4 Variable Documentation

6.9.4.1 const int vc::MDUID_AFD = 10

6.9.4.2 const int vc::MDUID_CDR1 = 0

6.9.4.3 const int vc::MDUID_CDR2 = 1

6.9.4.4 const int vc::MDUID_CRT1 = 4

6.9.4.5 const int vc::MDUID_CRT2 = 5

6.9.4.6 const int vc::MDUID_CRT3 = 6

6.9.4.7 const int vc::MDUID_CRT4 = 7

6.9.4.8 const int vc::MDUID_MFD1 = 8

6.9.4.9 const int vc::MDUID_MFD2 = 9

6.9.4.10 const int vc::MDUID_PLT1 = 2

6.9.4.11 const int vc::MDUID_PLT2 = 3

Chapter 7

Class Documentation

7.1 mps::_DATA_TABLE Struct Reference

```
#include <SSME.h>
```

Public Attributes

- double timeTAG
- ENGINE_STATUS STATUS
- double cmdPC
- bool StartEnable
- bool ShutdownEnable
- double powerlevel
- double posOPOV
- double posFPOV
- double posMFV
- double posMOV
- double posCCV
- double timeESC
- double timeCO

7.1.1 Member Data Documentation

7.1.1.1 double mps::_DATA_TABLE::timeTAG

7.1.1.2 ENGINE_STATUS mps::_DATA_TABLE::STATUS

7.1.1.3 double mps::_DATA_TABLE::cmdPC

7.1.1.4 bool mps::_DATA_TABLE::StartEnable

7.1.1.5 bool mps::_DATA_TABLE::ShutdownEnable

7.1.1.6 double mps::_DATA_TABLE::powerlevel

7.1.1.7 double mps::_DATA_TABLE::posOPOV

7.1.1.8 double mps::_DATA_TABLE::posFPOV

7.1.1.9 double mps::_DATA_TABLE::posMFV

7.1.1.10 double mps::_DATA_TABLE::posMOV

7.1.1.11 double mps::_DATA_TABLE::posCCV

7.1.1.12 double mps::_DATA_TABLE::timeESC

7.1.1.13 double mps::_DATA_TABLE::timeCO

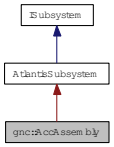
The documentation for this struct was generated from the following file:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/SSME.h

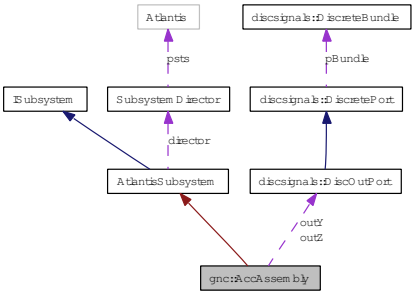
7.2 gnc::AccAssembly Class Reference

`#include <AA.h>`

Inheritance diagram for gnc::AccAssembly:



Collaboration diagram for gnc::AccAssembly:



Public Member Functions

- AccAssembly (SubsystemDirector *direct, const string &ident, const VECTOR3 &ref)
- virtual ~AccAssembly ()

Private Attributes

- VECTOR3 ref
- VECTOR3 vY
- VECTOR3 vZ
- DiscOutPort outY
- DiscOutPort outZ

7.2.1 Constructor & Destructor Documentation

7.2.1.1 gnc::AccAssembly::AccAssembly (SubsystemDirector * *direct*, const string & *ident*, const VECTOR3 & *ref*)

Parameters:

direct Reference to the responsible SubsystemDirector

ident Identification of the AA

ref Reference position of the assembly in local vessel coordinates

7.2.1.2 virtual gnc::AccAssembly::~AccAssembly () [virtual]

7.2.2 Member Data Documentation

7.2.2.1 VECTOR3 gnc::AccAssembly::ref [private]

7.2.2.2 VECTOR3 gnc::AccAssembly::vY [private]

7.2.2.3 VECTOR3 gnc::AccAssembly::vZ [private]

7.2.2.4 DiscOutPort gnc::AccAssembly::outY [private]

7.2.2.5 DiscOutPort gnc::AccAssembly::outZ [private]

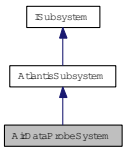
The documentation for this class was generated from the following file:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/gnc/AA.h

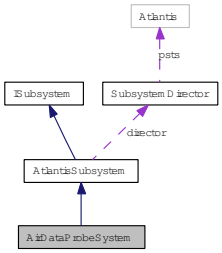
7.3 AirDataProbeSystem Class Reference

```
#include <AirDataProbeSystem.h>
```

Inheritance diagram for AirDataProbeSystem:



Collaboration diagram for AirDataProbeSystem:



Public Types

- enum DEPLOY_MODE { ADPS_DEPLOY, ADPS_STOW }
- enum STOW_ENABLE_STATE { ADPS_STOW_ENABLE, ADPS_STOW_INHIBIT }

Public Member Functions

- virtual void OnPropagate (double fSimT, double fDeltaT, double fMJD)
- double GetTotalTemperature (int side) const
- double GetDynamicPressure (int side) const
- virtual double GetStaticPressure (int side) const
- virtual void OnPostStep (double fSimT, double fDeltaT, double fMJD)
- bool IsDeployed (int side) const
- void SetDeployMode (int side, DEPLOY_MODE mode)
- virtual void OnPreStep (double fSimT, double fDeltaT, double fMJD)
- AirDataProbeSystem (SubsystemDirector *_director)
- virtual ~AirDataProbeSystem ()

Private Types

- enum DEPLOY_STATE { ADPS_DEPLOYED, ADPS_DEPLOYING, ADPS_STOWING, ADPS_-STOWED }

Private Attributes

- double right_deploy [2]
- double left_deploy [2]
- DEPLOY_MODE left_mode [2]
- DEPLOY_MODE right_mode [2]
- DEPLOY_STATE left_state [2]
- DEPLOY_STATE right_state [2]
- STOW_ENABLE_STATE left_stow_state [2]
- STOW_ENABLE_STATE right_stow_state [2]

7.3.1 Member Enumeration Documentation

7.3.1.1 enum AirDataProbeSystem::DEPLOY_STATE [private]

Enumerator:

ADPS_DEPLOYED

ADPS_DEPLOYING

ADPS_STOWING

ADPS_STOWED

7.3.1.2 enum AirDataProbeSystem::DEPLOY_MODE

Enumerator:

ADPS_DEPLOY

ADPS_STOW

7.3.1.3 enum AirDataProbeSystem::STOW_ENABLE_STATE

Enumerator:

ADPS_STOW_ENABLE

ADPS_STOW_INHIBIT

7.3.2 Constructor & Destructor Documentation

7.3.2.1 AirDataProbeSystem::AirDataProbeSystem (SubsystemDirector * *_director*)

7.3.2.2 AirDataProbeSystem::~AirDataProbeSystem () [virtual]

7.3.3 Member Function Documentation

7.3.3.1 void AirDataProbeSystem::OnPropagate (double *fSimT*, double *fDeltaT*, double *fMJD*) [virtual]

Propagate the state vector from old state to new state.

Parameters:

fSimT current simulation time

fDeltaT Simulation time step length

fMJD current modified julian date

Reimplemented from AtlantisSubsystem.

7.3.3.2 double AirDataProbeSystem::GetTotalTemperature (int *side*) const

7.3.3.3 double AirDataProbeSystem::GetDynamicPressure (int *side*) const

7.3.3.4 double AirDataProbeSystem::GetStaticPressure (int *side*) const `[virtual]`

7.3.3.5 void AirDataProbeSystem::OnPostStep (double *fSimT*, double *fDeltaT*, double *fMJD*) `[virtual]`

Finish calculations after the propagation of the state vector

Parameters:

fSimT current simulation time

fDeltaT Simulation time step length

fMJD current modified julian date

Reimplemented from AtlantisSubsystem.

Here is the call graph for this function:

7.3.3.6 bool AirDataProbeSystem::IsDeployed (int *side*) const

7.3.3.7 void AirDataProbeSystem::SetDeployMode (int *side*, DEPLOY_MODE *mode*)

7.3.3.8 void AirDataProbeSystem::OnPreStep (double *fSimT*, double *fDeltaT*, double *fMJD*) `[virtual]`

Prepare and execute calculations before the propagation of the state vector

Parameters:

fSimT current simulation time

fDeltaT Simulation time step length

fMJD current modified julian date

Reimplemented from AtlantisSubsystem.

7.3.4 Member Data Documentation

7.3.4.1 double AirDataProbeSystem::right_deploy[2] [private]

7.3.4.2 double AirDataProbeSystem::left_deploy[2] [private]

7.3.4.3 DEPLOY_MODE AirDataProbeSystem::left_mode[2] [private]

7.3.4.4 DEPLOY_MODE AirDataProbeSystem::right_mode[2] [private]

7.3.4.5 DEPLOY_STATE AirDataProbeSystem::left_state[2] [private]

7.3.4.6 DEPLOY_STATE AirDataProbeSystem::right_state[2] [private]

7.3.4.7 STOW_ENABLE_STATE AirDataProbeSystem::left_stow_state[2] [private]

7.3.4.8 STOW_ENABLE_STATE AirDataProbeSystem::right_stow_state[2] [private]

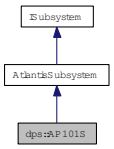
The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AirDataProbeSystem.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AirDataProbeSystem.cpp

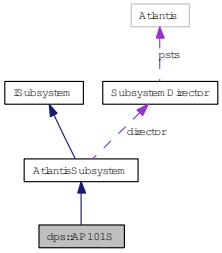
7.4 dps::AP101S Class Reference

`#include <AP101S.h>`

Inheritance diagram for dps::AP101S:



Collaboration diagram for dps::AP101S:



Public Member Functions

- AP101S (SubsystemDirector *_director, const string &_ident)
- virtual ~AP101S ()

Protected Member Functions

- void LoadMemoryConfiguration (const GPCMemoryConfiguration *pMemConfig)
- void LoadSoftware (const GPCSoftware *pSoftware)

Private Attributes

- long lMemory [262144]

7.4.1 Constructor & Destructor Documentation

7.4.1.1 dps::AP101S::AP101S (SubsystemDirector * *_director*, const string & *_ident*)

7.4.1.2 dps::AP101S::~AP101S () [virtual]

7.4.2 Member Function Documentation

7.4.2.1 void dps::AP101S::LoadMemoryConfiguration (const GPCMemoryConfiguration * *pMemConfig*) [protected]

7.4.2.2 void dps::AP101S::LoadSoftware (const GPCSoftware * *pSoftware*) [protected]

7.4.3 Member Data Documentation

7.4.3.1 long dps::AP101S::lMemory[262144] [private]

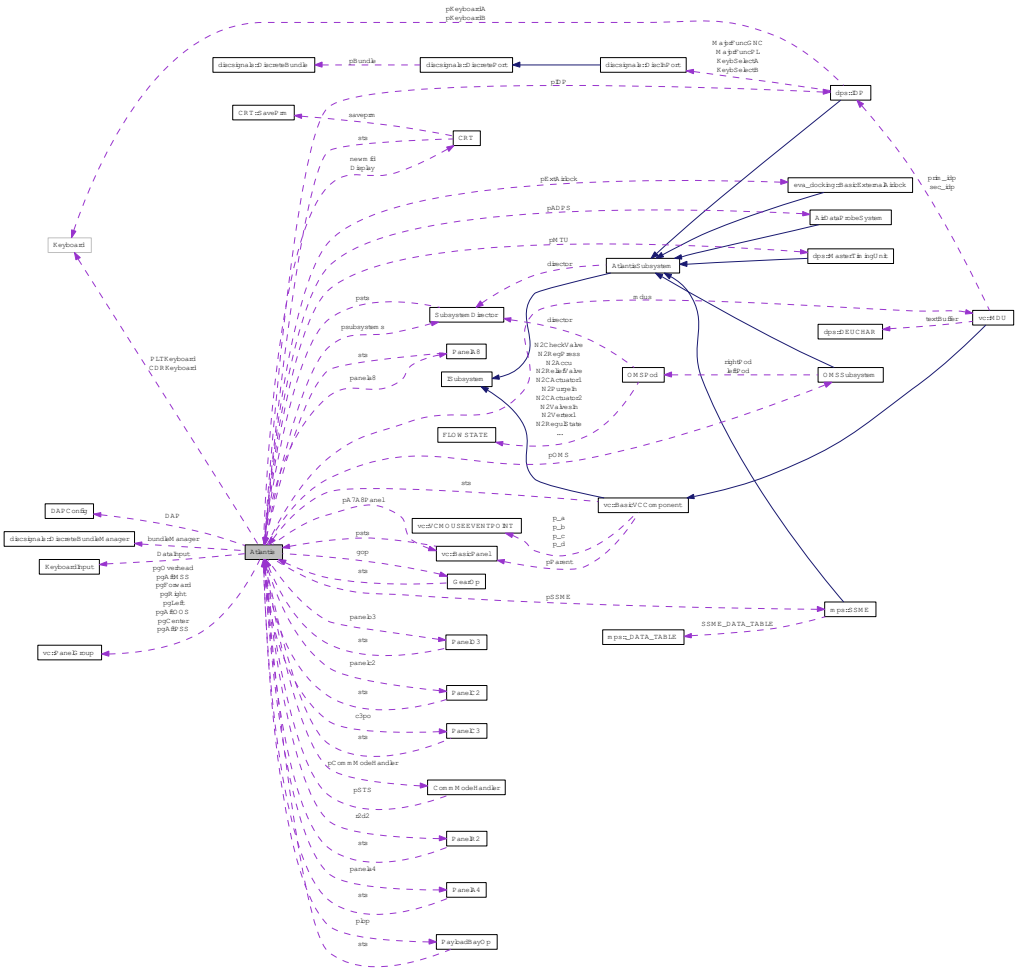
The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/AP101S.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/AP101S.cpp

7.5 Atlantis Class Reference

`#include <Atlantis.h>`

Collaboration diagram for Atlantis:



Public Types

- enum { VCM_FLIGHTDECK = 0, VCM_MIDDECK, VCM_AIRLOCK }

Public Member Functions

- Atlantis (OBJHANDLE hObj, int fmodel)
- ~Atlantis ()
- void AddOrbiterVisual (const VECTOR3 &ofs)
- void AddSRBVisual (int which, const VECTOR3 &ofs)
- void AddTankVisual (const VECTOR3 &ofs)
- virtual DiscreteBundleManager * BundleManager () const

- VECTOR3 CalcAnimationFKArm ()
- void CalcAnimationFKArm (VECTOR3 &pos, VECTOR3 &dir)
- void clbkAnimate (double simt)
- int clbkConsumeBufferedKey (DWORD key, bool down, char *kstate)
- void clbkDrawHUD (int mode, const HUDPAINTSPEC *hps, HDC hDC)
- void clbkFocusChanged (bool getfocus, OBJHANDLE hNewVessel, OBJHANDLE hOldVessel)
- bool clbkLoadGenericCockpit ()
- void clbkLoadStateEx (FILEHANDLE scn, void *vs)
- bool clbkLoadVC (int id)
- void clbkMFDMode (int mfd, int mode)
- bool clbkPlaybackEvent (double simt, double event_t, const char *event_type, const char *event)
- void clbkPostStep (double simt, double simdt, double mjd)
- void clbkPreStep (double simT, double simDT, double mjd)
- void clbkSaveState (FILEHANDLE scn)
- void clbkSetClassCaps (FILEHANDLE cfg)
- void clbkSetStateEx (const void *status)
- bool clbkVCMouseEvent (int id, int event, VECTOR3 &p)
- bool clbkVCRedrawEvent (int id, int event, SURFHANDLE surf)
- void clbkVisualCreated (VISHANDLE vis, int refcount)
- void clbkVisualDestroyed (VISHANDLE vis, int refcount)
- void ClearMeshes ()
- void DefineAnimations (void)
- virtual short GetETPropellant (unsigned short usGPCID)
- virtual unsigned short GetGPCMET (unsigned short usGPCID, unsigned short &usDay, unsigned short &usHour, unsigned short &usMin, unsigned short &usSec)
- virtual short GetGPCRefHDot (unsigned short usGPCID, double &fRefHDot)
- virtual unsigned short GetGPCLVLHVel (unsigned short usGPCID, VECTOR3 &vel)
- virtual dps::IDP * GetIDP (unsigned short usIDPNumber) const
- virtual short GetLastCreatedMFD () const
- virtual bool GetLiftOffFlag () const
- virtual vc::MDU * GetMDU (unsigned short usMDUID) const
- virtual double GetOMSPressure (OMS_REF oms_ref, unsigned short tank_id)
- virtual const VECTOR3 & GetOrbiterCoGOffset () const
- virtual short GetSRBChamberPressure (unsigned short which_srb)
- virtual bool HasExternalAirlock () const
- virtual bool IsValidSPEC (int gpc, int spec)
- void Jettison ()
- void OperateSpeedbrake (AnimState::Action action)
- void PaintMarkings (SURFHANDLE tex)
- virtual bool RegisterMDU (unsigned short usMDUID, vc::MDU *pMDU)
- void RevertSpeedbrake ()
- void SeparateBoosters (double srb_time)
- void SeparateMMU (void)
- void SeparateTank (void)
- virtual void SetAirDataProbeDeployment (int side, double position)
- void SetAnimationArm (UINT anim, double state)
- void SetAnimationIKArm (VECTOR3 arm_dpos)
- void SetBayDoorLatchPosition (int gang, double pos)
- void SetBayDoorPosition (double pos)
- void SetETUmbDoorPosition (double pos, int door)

- void SetKuAntennaPosition (double pos)
- virtual void SetKuGimbalAngles (double fAlpha, double fbeta)
- void SetLastCreatedMFD (unsigned short usMDU)
- void SetLaunchConfiguration (void)
- void SetOrbiterConfiguration (void)
- void SetOrbiterTankConfiguration (void)
- void SetPostLaunchConfiguration (double srbtime)
- void SetRadiatorPosition (double pos)
- void SetRadLatchPosition (double pos)
- void SetSpeedbrake (double tgt)
- void ToggleGrapple (void)
- void AutoGrappleSequence ()
- void AutoReleaseSequence ()
- void ToggleArrest (void)
- void UpdateMesh ()
- void UpdateRMSPositions ()
- void UpdateRMSAngles ()
- void UpdateSSMEGimbalAnimations ()

Public Attributes

- AnimState::Action spdb_status
- int ___iCurrentManifold
- char WingName [256]
- int VCMode
- int status
- bool bManualSeparate
- double t0
- double met
- int MET [4]
- int Launch_time [4]
- int MET_Add [4]
- WORD srb_id1
- WORD srb_id2
- enum Atlantis:: { ... } vcDeckMode
- UINT mesh_orbiter
- UINT mesh_cockpit
- UINT mesh_vc
- UINT mesh_middeck
- UINT mesh_rms
- UINT mesh_tank
- UINT mesh_srb [2]
- UINT mesh_kuband
- UINT mesh_ods
- UINT mesh_cargo_static
- vc::BasicPanel * pA7A8Panel
- double spdb_proc
- double spdb_tgt
- double ldoor_drag
- double rdoor_drag

- bool center_arm
- bool arm_moved
- double center_arm_t
- bool do_eva
- bool do_plat
- bool do_cargostatic
- VECTOR3 orbiter_ofs
- VECTOR3 ofs_sts_sat
- VECTOR3 cargo_static_ofs
- VISHANDLE vis
- MESHHANDLE hOrbiterMesh
- MESHHANDLE hOrbiterCockpitMesh
- MESHHANDLE hOrbiterVCMesh
- MESHHANDLE hMidDeckMesh
- MESHHANDLE hOrbiterRMSMesh
- MESHHANDLE hTankMesh
- MESHHANDLE hSRBMesh [2]
- MESHHANDLE hODSMesh
- MESHHANDLE hKUBandMesh
- char cargo_static_mesh_name [256]
- ATTACHMENTHANDLE sat_attach
- ATTACHMENTHANDLE rms_attach
- ATTACHMENTHANDLE obss_attach
- ATTACHMENTHANDLE ahHDP
- VECTOR3 arm_tip [3]
- SubsystemDirector * psubsystems
- PayloadBayOp * plop
- GearOp * gop
- PanelA4 * panela4
- PanelA8 * panela8
- PanelC2 * panelc2
- PanelC3 * c3po
- PanelO3 * panelo3
- PanelR2 * r2d2
- Keyboard * CDRKeyboard
- Keyboard * PLTKeyboard
- bool PitchActive
- bool YawActive
- bool RollActive
- OBJHANDLE ThisVessel

Protected Member Functions

- void AddKUBandVisual (const VECTOR3 ofs)
- void TriggerLiftOff ()
- void DisplayCameraLabel (const char *pszLabel)

Private Types

- enum { MNVR_OFF, MNVR_STARTING, MNVR_IN_PROGRESS, MNVR_COMPLETE }

Private Member Functions

- void StopAllManifolds ()
- void FireAllNextManifold ()
- void AddPrimaryRCSExhaust (THRUSTER_HANDLE thX)
- void CreateRightARCS (const VECTOR3 &ref_pos)
- void CreateLeftARCS (const VECTOR3 &ref_pos)
- void CreateFRCS (const VECTOR3 &ref_pos)
- void CreateMPSGOXVents (const VECTOR3 &ref_pos)
- void CreateOrbiterTanks ()
- void DefineSSMEExhaust ()
- void SignalGSEBreakHDP ()
- void SignalGSEStart ()
- void ShowMidDeck ()
- void HideMidDeck ()
- void ShowODS ()
- void HideODS ()
- void DefineKUBandAnimations ()
- void LaunchClamps ()
- void CreateAttControls_RCS (VECTOR3 center)
- void DisableAllRCS ()
- void EnableAllRCS ()
- void DisableControlSurfaces ()
- void EnableControlSurfaces ()
- void ToggleVCMode ()
- bool ArmCradled ()
- void UpdateMPMMicroswitches ()
- void UpdateMRLMicroswitches ()
- bool SatGrappled () const
- bool SatStowed () const
- ATTACHMENTHANDLE CanArrest () const
- void AutoMainGimbal ()
- void SteerGimbal ()
- void RateCommand ()
- void Throttle (double dt)
- void FailEngine (int engine)
- void InitializeAutopilot ()
- double CalcNetThrust ()
- void CalcThrustAngles ()
- double CalculateAzimuth ()
- void MajorCycle ()
- void Navigate ()
- void Estimate ()
- void Guide ()
- void GPC (double dt)
- void Maneuver (double dt)
- void SetILoads ()
- bool Input (int mfd, int change, char *Name, char *Data=NULL)
- void GimbalOMS (VECTOR3 Targets)
- void LoadManeuver ()

- void UpdateDAP ()
- void AttControl (double SimdT)
- void CalcManeuverTargets (VECTOR3 NullRates)
- void SetRates (VECTOR3 &Rates)
- void CalcRequiredRates (VECTOR3 &Rates)
- void CalcRequiredRates (VECTOR3 &Rates, const VECTOR3 &NullRates)
- void CalcLVLHAttitude (VECTOR3 &Output)
- VECTOR3 ConvertAnglesBetweenM50AndOrbiter (const VECTOR3 &Angles, bool ToOrbiter=false)
- VECTOR3 ConvertOrbiterAnglesToLocal (const VECTOR3 &Angles)
- VECTOR3 ConvertVectorBetweenOrbiterAndM50 (const VECTOR3 &Input)
- VECTOR3 ConvertLVLHAnglesToM50 (const VECTOR3 &Input)
- VECTOR3 CalcRelLVLHAttitude (VECTOR3 &Target)
- VECTOR3 CalcPitchYawRollAngles (VECTOR3 &RelAttitude)
- VECTOR3 CalcPitchYawRollAngles ()
- double NullStartAngle (double Rates, AXIS Axis)
- void EnableThrusters (const int Thrusters[ ], int nThrusters)
- void DisableThrusters (const int Thrusters[ ], int nThrusters)
- VECTOR3 GetPYR (VECTOR3 Pitch, VECTOR3 YawRoll)
- VECTOR3 GetPYR2 (VECTOR3 Pitch, VECTOR3 YawRoll)
- void RotateVector (const VECTOR3 &Initial, const VECTOR3 &Angles, VECTOR3 &Result)
- void RotateVectorPYR (const VECTOR3 &Initial, const VECTOR3 &Angles, VECTOR3 &Result)
- void GetRotMatrixX (double Angle, MATRIX3 &RotMatrixX)
- void GetRotMatrixY (double Angle, MATRIX3 &RotMatrixY)
- void GetRotMatrixZ (double Angle, MATRIX3 &RotMatrixZ)
- void MultiplyByMatrix (const VECTOR3 &Initial, const MATRIX3 &RotMatrix, VECTOR3 &Result)
- VECTOR3 Normalize (const VECTOR3 &a)
- VECTOR3 Projection (const VECTOR3 &u, const VECTOR3 &v)
- VECTOR3 ToRad (const VECTOR3 &Input)
- VECTOR3 ToDeg (const VECTOR3 &Input)
- VECTOR3 NormZ (VECTOR3 &v)
- VECTOR3 RotateVectorX (const VECTOR3 &v, double angle)
- VECTOR3 RotateVectorY (const VECTOR3 &v, double angle)
- VECTOR3 RotateVectorZ (const VECTOR3 &v, double angle)
- bool Eq (const double d1, const double d2, double dDiff=0.00001)
- double range (double min, double value, double max)
- double a (double t)
- double b0 (double TT)
- double bn (double TT, int n)
- double c0 (double TT)
- double cn (double TT, int n)
- void SetAnimationCameras ()

Private Attributes

- double slag1
- double slag2
- double slag3
- PSTREAM_HANDLE pshSlag1 [2]
- PSTREAM_HANDLE pshSlag2 [2]
- PSTREAM_HANDLE pshSlag3 [2]
- bool bSRBCutoffFlag
- bool bLiftOff
- bool bHasKUBand
- bool bHasODS
- bool bMidDeckVisible
- CommModeHandler * pCommModeHandler
- vc::PanelGroup pgForward
- vc::PanelGroup pgLeft
- vc::PanelGroup pgCenter
- vc::PanelGroup pgRight
- vc::PanelGroup pgOverhead
- vc::PanelGroup pgAftOOS
- vc::PanelGroup pgAftPSS
- vc::PanelGroup pgAftMSS
- dps::MasterTimingUnit * pMTU
- dps::IDP * pIDP [4]
- OMSSubsystem * pOMS
- mps::SSME * pSSME [3]
- eva_docking::BasicExternalAirlock * pExtAirlock
- AirDataProbeSystem * pADPS
- SURFHANDLE tex_rcs
- bool bUseRealRCS
- unsigned short usCurrentPlayerChar
- bool bCommMode
- UINT anim_door
- UINT anim_rad
- UINT anim_clatch [4]
- UINT anim_portTS
- UINT anim_kubd
- UINT anim_elev
- UINT anim_bf
- UINT anim_laileron
- UINT anim_raileron
- UINT anim_rudder
- UINT anim_spdb
- UINT anim_dummy
- UINT anim_letumbdoor
- UINT anim_retumbdoor
- UINT anim_ssmeTyaw
- UINT anim_ssmeTpitch
- UINT anim_ssmeRyaw
- UINT anim_ssmeRpitch

- UINT anim_ssmeLyaw
- UINT anim_ssmeLpitch
- UINT anim_adpl
- UINT anim_adpr
- UINT anim_kualpha
- UINT anim_kubeta
- UINT anim_camFLyaw
- UINT anim_camFLpitch
- UINT anim_camFRyaw
- UINT anim_camFRpitch
- UINT anim_camBLyaw
- UINT anim_camBLpitch
- UINT anim_camBRyaw
- UINT anim_camBRpitch
- double camFLyaw
- double camFLpitch
- double camFRyaw
- double camFRpitch
- double camBLyaw
- double camBLpitch
- double camBRyaw
- double camBRpitch
- bool cameraLowRate
- bool cameraMoved
- int cameraControl
- MGROUP_TRANSFORM * CameraFLYaw
- MGROUP_TRANSFORM * CameraFLPitch
- MGROUP_TRANSFORM * CameraFRYaw
- MGROUP_TRANSFORM * CameraFRPitch
- MGROUP_TRANSFORM * CameraBLYaw
- MGROUP_TRANSFORM * CameraBLPitch
- MGROUP_TRANSFORM * CameraBRYaw
- MGROUP_TRANSFORM * CameraBRPitch
- PROPELLANT_HANDLE ph_oms
- PROPELLANT_HANDLE ph_tank
- PROPELLANT_HANDLE ph_srb
- PROPELLANT_HANDLE ph_frcs
- PROPELLANT_HANDLE ph_lrcs
- PROPELLANT_HANDLE ph_rrcs
- THRUSTER_HANDLE th_main [3]
- THRUSTER_HANDLE th_oms [2]
- THRUSTER_HANDLE th_srb [2]
- THRUSTER_HANDLE th_att_rcs [18]
- THRUSTER_HANDLE th_att_lin [10]
- THRUSTER_HANDLE th_ssme_gox [3]
- THRUSTER_HANDLE th_ssme_loxdump [3]
- THRUSTER_HANDLE thManFRCS1 [4]
- THRUSTER_HANDLE thManFRCS2 [4]
- THRUSTER_HANDLE thManFRCS3 [4]
- THRUSTER_HANDLE thManFRCS4 [2]

- THRUSTER_HANDLE thManFRCS5 [2]
- THRUSTER_HANDLE thManLRCS1 [3]
- THRUSTER_HANDLE thManLRCS2 [3]
- THRUSTER_HANDLE thManLRCS3 [3]
- THRUSTER_HANDLE thManLRCS4 [3]
- THRUSTER_HANDLE thManLRCS5 [2]
- THRUSTER_HANDLE thManRRCS1 [3]
- THRUSTER_HANDLE thManRRCS2 [3]
- THRUSTER_HANDLE thManRRCS3 [3]
- THRUSTER_HANDLE thManRRCS4 [3]
- THRUSTER_HANDLE thManRRCS5 [2]
- UINT ex_main [3]
- UINT ex_retro [2]
- bool RCSEnabled
- THGROUP_HANDLE thg_main
- THGROUP_HANDLE thg_srb
- THGROUP_HANDLE thg_retro
- CTRLSURFHANDLE hrudder
- CTRLSURFHANDLE hlaileron
- CTRLSURFHANDLE hraileron
- CTRLSURFHANDLE helevator
- CTRLSURFHANDLE hbodyflap
- bool ControlSurfacesEnabled
- bool bSSMEGOXVent
- VECTOR3 SSMET_GOX_REF1
- VECTOR3 SSMEL_GOX_REF1
- VECTOR3 SSMER_GOX_REF1
- PROPELLANT_HANDLE apu_tank [3]
- PROPELLANT_HANDLE oms_helium_tank [2]
- PROPELLANT_HANDLE mps_helium_tank [3]
- int MPS_Manifold_Press [2]
- int MPS_He_Reg_Press [4]
- int MPS_He_Tank_Press [4]
- int Hydraulic_Press [3]
- ANIMATIONCOMPONENT_HANDLE hAC_arm
- ANIMATIONCOMPONENT_HANDLE hAC_sat
- ANIMATIONCOMPONENT_HANDLE hAC_satref
- MGROUP_TRANSFORM * rms_anim [7]
- MGROUP_TRANSFORM * rms_rollout_anim
- UINT anim_arm_sy
- UINT anim_arm_sp
- UINT anim_arm_ep
- UINT anim_arm_wp
- UINT anim_arm_wy
- UINT anim_arm_wr
- UINT anim_arm_ee
- UINT anim_rollout
- double arm_sy
- double arm_sp
- double arm_ep

- double arm_wp
- double arm_wy
- double arm_wr
- double sy_angle
- double sp_angle
- double ep_angle
- double wp_angle
- double wy_angle
- double wr_angle
- double sp_null
- double ep_null
- AnimState RMSRollout
- int MPM_Microswitches [2][2]
- double shoulder_brace
- double MRL [2]
- int MRL_FwdMicroswitches [2][3]
- int MRL_MidMicroswitches [2][3]
- int MRL_AftMicroswitches [2][3]
- bool RMS
- bool DisplayJointAngles
- bool bGrappleInProgress
- bool bReleaseInProgress
- AnimState Grapple
- AnimState Rigidize
- AnimState Extend
- int EEGrappleMode
- VECTOR3 arm_wrist_pos
- VECTOR3 arm_ee_pos
- VECTOR3 arm_ee_dir
- VECTOR3 arm_ee_rot
- VECTOR3 arm_tgt_pos
- VECTOR3 arm_tgt_dir
- double lu
- double ll
- double wp_wy
- double wy_ee
- double shoulder_neutral
- double shoulder_range
- double shoulder_min
- double shoulder_max
- double elbow_neutral
- double elbow_range
- double elbow_min
- double elbow_max
- double wrist_neutral
- double wrist_range
- double wrist_min
- double wrist_max
- double wrist_yaw_neutral
- double wrist_yaw_range

- double wrist_yaw_min
- double wrist_yaw_max
- double wrist_roll_neutral
- double wrist_roll_range
- double wrist_roll_min
- double wrist_roll_max
- VECTOR3 shoulder_pos
- MGROUP_TRANSFORM * sat_anim
- MGROUP_TRANSFORM * sat_ref
- bool reset_mmu
- bool reset_sat
- OBJHANDLE hMMU
- OBJHANDLE hSAT
- double jettison_time
- bool render_cockpit
- VCHUDSPEC huds
- unsigned short usLastMDUID
- EXTMFDSPEC mfds [11]
- double mfdbright [11]
- double pl_mass
- VECTOR3 GVesselPos
- VECTOR3 GVesselVel
- VESSELSTATUS Status
- VECTOR3 LVLH_X
- VECTOR3 LVLH_Y
- VECTOR3 LVLH_Z
- double Throttle_Bucket [2]
- double OMS_Assist [2]
- double RollToHeadsUp
- double TgtInc
- double TgtLAN
- double TgtSpd
- double TgtAlt
- double TgtFPA
- double THeading
- double TAp
- double TPe
- double TTrA
- double TEcc
- double TgtRad
- double MaxThrust
- bool bAutopilot
- bool bThrottle
- bool bMECO
- bool bZThrust
- double tMECO
- double tSRBSep
- double dV
- VECTOR3 EngineNullPosition [3]
- int EngineFail

- double EngineFailTime
- bool bEngineFail
- VECTOR3 Thrust
- double target_pitch
- double last_target_pitch
- double CmdPDot
- double target_yaw
- double last_target_yaw
- double CmdYDot
- double target_roll
- double last_target_roll
- double CmdRDot
- double TLastMajorCycle
- VECTOR3 rh0
- VECTOR3 rv
- VECTOR3 vv
- VECTOR3 hv
- double r
- double v
- double h
- double theta
- double omega
- double phi
- VECTOR3 rh
- VECTOR3 thetah
- VECTOR3 hh
- VECTOR3 posMoon
- VECTOR3 velMoon
- VECTOR3 rmh
- double vr
- double vtheta
- double vh
- double Radius
- double SidDay
- double fh
- double pitch
- double yaw
- double roll
- double mu
- double Re
- double g
- double F
- double m
- double Isp
- double a0
- double tau
- double ThrAngleP
- double ThrAngleY
- double T
- double p

- double deltah
- double deltav
- double hT
- double metCutoff
- double ftheta
- double fdottheta
- double fdotdottheta
- double rbar
- double deltatheta
- double thetaT
- double fr
- double fdotr
- double d1
- double d2
- double d3
- double d4
- double A
- double B
- double C
- double TLastAB
- double fhdotrh
- double eCurrent
- int ops
- int SMOps
- int last_mfd
- KeyboardInput DataInput [3]
- int CRT_SEL [2]
- int item
- CRT ∗ Display [3]
- CRT ∗ newmfd
- DiscreteBundleManager ∗ bundleManager
- vc::MDU ∗ mdus [11]
- VECTOR3 NullRates
- int OMS
- double tig
- double TIG [4]
- double IgnitionTime
- bool BurnInProg
- bool BurnCompleted
- double WT
- VECTOR3 PEG7
- VECTOR3 DeltaV
- double C1
- double C2
- double HT
- double ThetaT
- VECTOR3 Trim
- int TV_ROLL
- double DeltaVTot
- double DeltaVTotms

- double BurnTime
- VECTOR3 VGO
- double vgoTot
- bool MNVRLOAD
- VECTOR3 BurnAtt
- double OMSGimbal [2][2]
- bool ManeuverinProg
- enum Atlantis:: { ... } ManeuverStatus
- double MNVR_TIME
- int START_TIME [4]
- int TGT_ID
- int BODY_VECT
- double P
- double Y
- double OM
- bool MNVR
- bool TRK
- bool ROT
- bool Pitch
- bool Yaw
- bool Roll
- VECTOR3 InertialOrientationRad
- VECTOR3 AngularVelocity
- VECTOR3 CurrentAttitude
- VECTOR3 LVLHOrientationReqd
- VECTOR3 LVLHError
- VECTOR3 LVLHRateVector
- VECTOR3 MNVR_OPTION
- VECTOR3 TRKROT_OPTION
- VECTOR3 REQD_ATT
- VECTOR3 PitchYawRoll
- VECTOR3 TargetAttM50
- VECTOR3 TargetAttOrbiter
- VECTOR3 ReqdRates
- VECTOR3 TrackRates
- ORBITPARAM oparam
- ELEMENTS el
- double TrkRate
- double Mass
- VECTOR3 PMI
- VECTOR3 Torque
- DAPConfig DAP [3]
- int edit
- int DAPMode [2]
- double RotRate
- double AttDeadband
- double RateDeadband
- bool NoseThrusters
- bool TailThrusters
- bool Thrusters

- int JetsEnabled
- double * stage1guidance [2]
- int stage1guidance_size
- double fTimeCameraLabel
- NOTEHANDLE nhCameraLabel
- char pszCameraLabelBuffer [80]

Friends

- class PayloadBayOp
- class GearOp
- class PanelA4
- class PanelA8
- class PanelC2
- class PanelC3
- class PanelO3
- class PanelR2
- class Keyboard
- class CRT
- class vc::MDU
- BOOL CALLBACK RMS_DlgProc (HWND hWnd, UINT uMsg, WPARAM wParam, LPARAM lParam)
- BOOL CALLBACK PAYCAM_DlgProc (HWND hWnd, UINT uMsg, WPARAM wParam, LPARAM lParam)

7.5.1 Member Enumeration Documentation

7.5.1.1 anonymous enum

Enumerator:

VCM_FLIGHTDECK

VCM_MIDDECK

VCM_AIRLOCK

7.5.1.2 anonymous enum [private]

Enumerator:

MNVR_OFF

MNVR_STARTING

MNVR_IN_PROGRESS

MNVR_COMPLETE

7.5.2 Constructor & Destructor Documentation

7.5.2.1 Atlantis::Atlantis (OBJHANDLE *hObj*, int *fmodel*)

7.5.2.2 Atlantis::~Atlantis ()

7.5.3 Member Function Documentation

7.5.3.1 void Atlantis::AddOrbiterVisual (const VECTOR3 & *ofs*)

7.5.3.2 void Atlantis::AddSRBVisual (int *which*, const VECTOR3 & *ofs*)

7.5.3.3 void Atlantis::AddTankVisual (const VECTOR3 & *ofs*)

7.5.3.4 virtual DiscreteBundleManager* Atlantis::BundleManager () const [virtual]

7.5.3.5 VECTOR3 Atlantis::CalcAnimationFKArm ()

7.5.3.6 void Atlantis::CalcAnimationFKArm (VECTOR3 & *pos*, VECTOR3 & *dir*)

7.5.3.7 void Atlantis::clbkAnimate (double *simt*)

7.5.3.8 int Atlantis::clbkConsumeBufferedKey (DWORD *key*, bool *down*, char * *kstate*)

7.5.3.9 void Atlantis::clbkDrawHUD (int *mode*, const HUDPAINTSPEC * *hps*, HDC *hDC*)

7.5.3.10 void Atlantis::clbkFocusChanged (bool *getfocus*, OBJHANDLE *hNewVessel*, OBJHANDLE *hOldVessel*)

7.5.3.11 bool Atlantis::clbkLoadGenericCockpit ()

7.5.3.12 void Atlantis::clbkLoadStateEx (FILEHANDLE *scn*, void * *vs*)

7.5.3.13 bool Atlantis::clbkLoadVC (int *id*)

7.5.3.14 void Atlantis::clbkMFDMode (int *mfd*, int *mode*)

7.5.3.15 bool Atlantis::clbkPlaybackEvent (double *simt*, double *event_t*, const char * *event_type*, const char * *event*)

7.5.3.16 void Atlantis::clbkPostStep (double *simt*, double *simdt*, double *mjd*)

7.5.3.17 void Atlantis::clbkPreStep (double *simT*, double *simDT*, double *mjd*)

7.5.3.18 void Atlantis::clbkSaveState (FILEHANDLE *scn*)

7.5.3.19 void Atlantis::clbkSetClassCaps (FILEHANDLE *cfg*)

7.5.3.20 void Atlantis::clbkSetStateEx (const void * *status*)

7.5.3.21 bool Atlantis::clbkVCMouseEvent (int *id*, int *event*, VECTOR3 & *p*)

7.5.3.22 bool Atlantis::clbkVCRedrawEvent (int *id*, int *event*, SURFHANDLE *surf*)

7.5.3.23 void Atlantis::clbkVisualCreated (VISHANDLE *vis*, int *refcount*)

7.5.3.24 void Atlantis::clbkVisualDestroyed (VISHANDLE *vis*, int *refcount*)

7.5.3.25 void Atlantis::ClearMeshes ()

7.5.3.26 void Atlantis::DefineAnimations (void)

7.5.3.95 bool Atlantis::ArmCradled () [private]

7.5.3.96 void Atlantis::UpdateMPMMicroswitches () [private]

7.5.3.97 void Atlantis::UpdateMRLMicroswitches () [private]

7.5.3.98 bool Atlantis::SatGrappled () const [inline, private]

7.5.3.99 bool Atlantis::SatStowed () const [inline, private]

7.5.3.100 ATTACHMENTHANDLE Atlantis::CanArrest () const [private]

7.5.3.101 void Atlantis::AutoMainGimbal () [private]

7.5.3.102 void Atlantis::SteerGimbal () [private]

7.5.3.103 void Atlantis::RateCommand () [private]

7.5.3.104 void Atlantis::Throttle (double *dt*) [private]

7.5.3.105 void Atlantis::FailEngine (int *engine*) [private]

7.5.3.106 void Atlantis::InitializeAutopilot () [private]

7.5.3.107 double Atlantis::CalcNetThrust () [private]

7.5.3.108 void Atlantis::CalcThrustAngles () [private]

7.5.3.109 double Atlantis::CalculateAzimuth () [private]

7.5.3.110 void Atlantis::MajorCycle () [private]

7.5.3.111 void Atlantis::Navigate () [private]

7.5.3.112 void Atlantis::Estimate () [private]

7.5.3.113 void Atlantis::Guide () [private]

7.5.3.114 void Atlantis::GPC (double *dt*) [private]

7.5.3.115 void Atlantis::Maneuver (double *dt*) [private]

7.5.3.116 void Atlantis::SetILoads () [private]

7.5.3.117 bool Atlantis::Input (int *mfd*, int *change*, char * *Name*, char * *Data* = NULL) [private]

7.5.3.118 void Atlantis::GimbalOMS (VECTOR3 *Targets*) [private]

7.5.3.119 void Atlantis::LoadManeuver () [private]

7.5.3.120 void Atlantis::UpdateDAP () [private]

7.5.3.121 void Atlantis::AttControl (double *SimdT*) [private]

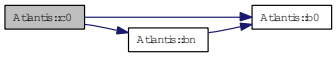
7.5.3.122 void Atlantis::CalcManeuverTargets (VECTOR3 *NullRates*) [private]

7.5.3.123 void Atlantis::SetRates (VECTOR3 & *Rates*) [private]

7.5.3.124 void Atlantis::CalcRequiredRates (VECTOR3 & *Rates*) [private]

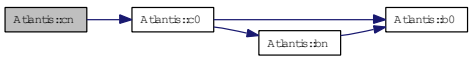
7.5.3.158 double Atlantis::c0 (double *TT*) [inline, private]

Here is the call graph for this function:



7.5.3.159 double Atlantis::cn (double *TT*, int *n*) [inline, private]

Here is the call graph for this function:



7.5.3.160 void Atlantis::SetAnimationCameras () [private]

7.5.3.161 void Atlantis::AddKUBandVisual (const VECTOR3 *ofs*) [protected]

7.5.3.162 void Atlantis::TriggerLiftOff () [protected]

7.5.3.163 void Atlantis::DisplayCameraLabel (const char * *pszLabel*) [protected]

7.5.4 Friends And Related Function Documentation

7.5.4.1 friend class PayloadBayOp [friend]

7.5.4.2 friend class GearOp [friend]

7.5.4.3 friend class PanelA4 [friend]

7.5.4.4 friend class PanelA8 [friend]

7.5.4.5 friend class PanelC2 [friend]

7.5.4.6 friend class PanelC3 [friend]

7.5.4.7 friend class PanelO3 [friend]

7.5.4.8 friend class PanelR2 [friend]

7.5.4.9 friend class Keyboard [friend]

7.5.4.10 friend class CRT [friend]

7.5.4.11 friend class vc::MDU [friend]

7.5.4.12 BOOL CALLBACK RMS_DlgProc (HWND *hWnd*, UINT *uMsg*, WPARAM *wParam*, LPARAM *lParam*) [friend]

7.5.4.13 BOOL CALLBACK PAYCAM_DlgProc (HWND *hWnd*, UINT *uMsg*, WPARAM *wParam*, LPARAM *lParam*) [friend]

7.5.5 Member Data Documentation

7.5.5.1 AnimState::Action Atlantis::spdb_status

7.5.5.2 int Atlantis::___iCurrentManifold

7.5.5.3 char Atlantis::WingName[256]

7.5.5.4 int Atlantis::VCMode

7.5.5.5 int Atlantis::status

Structural configuration

- 0 launch configuration
- 1 SRB's engaged
- 2 SRB's separated
- 3 Tank separated/orbiter only

7.5.5.6 bool Atlantis::bManualSeparate

7.5.5.7 double Atlantis::t0

7.5.5.8 double Atlantis::met

7.5.5.9 int Atlantis::MET[4]

7.5.5.10 int Atlantis::Launch_time[4]

7.5.5.11 int Atlantis::MET_Add[4]

7.5.5.12 WORD Atlantis::srb_id1

7.5.5.13 WORD Atlantis::srb_id2

7.5.5.14 enum { ... } Atlantis::vcDeckMode

7.5.5.15 UINT Atlantis::mesh_orbiter

7.5.5.16 UINT Atlantis::mesh_cockpit

7.5.5.17 UINT Atlantis::mesh_vc

7.5.5.18 UINT Atlantis::mesh_middeck

7.5.5.19 UINT Atlantis::mesh_rms

7.5.5.20 UINT Atlantis::mesh_tank

7.5.5.21 UINT Atlantis::mesh_srb[2]

7.5.5.22 UINT Atlantis::mesh_kuband

7.5.5.23 UINT Atlantis::mesh_ods

7.5.5.24 UINT Atlantis::mesh_cargo_static

7.5.5.25 vc::BasicPanel* Atlantis::pA7A8Panel

Pointer to the A7A8 custom panel region

7.5.5.26 double Atlantis::spdb_proc

7.5.5.27 double Atlantis::spdb_tgt

7.5.5.28 double Atlantis::ldoor_drag

7.5.5.29 double Atlantis::rdoor_drag

7.5.5.30 bool Atlantis::center_arm

7.5.5.31 bool Atlantis::arm_moved

7.5.5.32 double Atlantis::center_arm_t

7.5.5.33 bool Atlantis::do_eva

7.5.5.34 bool Atlantis::do_plat

7.5.5.35 bool Atlantis::do_cargostatic

7.5.5.36 VECTOR3 Atlantis::orbiter_ofs

7.5.5.37 VECTOR3 Atlantis::ofs_sts_sat

7.5.5.38 VECTOR3 Atlantis::cargo_static_ofs

7.5.5.39 VISHANDLE Atlantis::vis

7.5.5.40 MESHHANDLE Atlantis::hOrbiterMesh

7.5.5.41 MESHHANDLE Atlantis::hOrbiterCockpitMesh

7.5.5.42 MESHHANDLE Atlantis::hOrbiterVCMesh

7.5.5.43 MESHHANDLE Atlantis::hMidDeckMesh

7.5.5.44 MESHHANDLE Atlantis::hOrbiterRMSMesh

7.5.5.45 MESHHANDLE Atlantis::hTankMesh

7.5.5.46 MESHHANDLE Atlantis::hSRBMesh[2]

7.5.5.47 MESHHANDLE Atlantis::hODSMesh

7.5.5.48 MESHHANDLE Atlantis::hKUBandMesh

7.5.5.49 char Atlantis::cargo_static_mesh_name[256]

7.5.5.50 ATTACHMENTHANDLE Atlantis::sat_attach

7.5.5.51 ATTACHMENTHANDLE Atlantis::rms_attach

7.5.5.52 ATTACHMENTHANDLE Atlantis::obss_attach

7.5.5.53 ATTACHMENTHANDLE Atlantis::ahHDP

7.5.5.54 VECTOR3 Atlantis::arm_tip[3]

7.5.5.55 SubsystemDirector* Atlantis::psubsystems

7.5.5.56 PayloadBayOp* Atlantis::plop

7.5.5.95 AirDataProbeSystem* Atlantis::pADPS [private]

7.5.5.96 SURFHANDLE Atlantis::tex_rcs [private]

7.5.5.97 bool Atlantis::bUseRealRCS [private]

7.5.5.98 unsigned short Atlantis::usCurrentPlayerChar [private]

7.5.5.99 bool Atlantis::bCommMode [private]

7.5.5.100 UINT Atlantis::anim_door [private]

7.5.5.101 UINT Atlantis::anim_rad [private]

7.5.5.102 UINT Atlantis::anim_clatch[4] [private]

7.5.5.103 UINT Atlantis::anim_portTS [private]

7.5.5.104 UINT Atlantis::anim_kubd [private]

7.5.5.105 UINT Atlantis::anim_elev [private]

7.5.5.106 UINT Atlantis::anim_bf [private]

7.5.5.107 UINT Atlantis::anim_laileron [private]

7.5.5.108 UINT Atlantis::anim_raileron [private]

7.5.5.109 UINT Atlantis::anim_rudder [private]

7.5.5.110 UINT Atlantis::anim_spdb [private]

7.5.5.111 UINT Atlantis::anim_dummy [private]

7.5.5.112 UINT Atlantis::anim_letumbdoor [private]

7.5.5.113 UINT Atlantis::anim_retumbdoor [private]

7.5.5.114 UINT Atlantis::anim_ssmeTyaw [private]

7.5.5.115 UINT Atlantis::anim_ssmeTpitch [private]

7.5.5.116 UINT Atlantis::anim_ssmeRyaw [private]

7.5.5.117 UINT Atlantis::anim_ssmeRpitch [private]

7.5.5.118 UINT Atlantis::anim_ssmeLyaw [private]

7.5.5.119 UINT Atlantis::anim_ssmeLpitch [private]

7.5.5.120 UINT Atlantis::anim_adpl [private]

7.5.5.121 UINT Atlantis::anim_adpr [private]

7.5.5.122 UINT Atlantis::anim_kualpha [private]

7.5.5.123 UINT Atlantis::anim_kubeta [private]

7.5.5.124 UINT Atlantis::anim_camFLyaw [private]

7.5.5.125 UINT Atlantis::anim_camFLpitch [private]

7.5.5.163 THRUSTER_HANDLE Atlantis::th_ssme_loxdump[3] [private]

To be used for visualizing the LOX MPS dump after MECO.

7.5.5.164 THRUSTER_HANDLE Atlantis::thManFRCS1[4] [private]

Forward Manifold 1

7.5.5.165 THRUSTER_HANDLE Atlantis::thManFRCS2[4] [private]

Forward Manifold 2

7.5.5.166 THRUSTER_HANDLE Atlantis::thManFRCS3[4] [private]

Forward Manifold 3

7.5.5.167 THRUSTER_HANDLE Atlantis::thManFRCS4[2] [private]

Forward Manifold 4

7.5.5.168 THRUSTER_HANDLE Atlantis::thManFRCS5[2] [private]

Forward Manifold 5

7.5.5.169 THRUSTER_HANDLE Atlantis::thManLRCS1[3] [private]

Left Manifold 1

7.5.5.170 THRUSTER_HANDLE Atlantis::thManLRCS2[3] [private]

Left Manifold 2

7.5.5.171 THRUSTER_HANDLE Atlantis::thManLRCS3[3] [private]

Left Manifold 3

7.5.5.172 THRUSTER_HANDLE Atlantis::thManLRCS4[3] [private]

Left Manifold 4

7.5.5.173 THRUSTER_HANDLE Atlantis::thManLRCS5[2] [private]

Left Manifold 5

7.5.5.174 THRUSTER_HANDLE Atlantis::thManRRCS1[3] [private]

Right Manifold 1

7.5.5.175 THRUSTER_HANDLE Atlantis::thManRRCS2[3] [private]

Right Manifold 2

7.5.5.176 THRUSTER_HANDLE Atlantis::thManRRCS3[3] [private]

Right Manifold 3

7.5.5.177 THRUSTER_HANDLE Atlantis::thManRRCS4[3] [private]

Right Manifold 4

7.5.5.178 THRUSTER_HANDLE Atlantis::thManRRCS5[2] [private]

Right Manifold 5

7.5.5.179 UINT Atlantis::ex_main[3] [private]

7.5.5.180 UINT Atlantis::ex_retro[2] [private]

7.5.5.181 bool Atlantis::RCSEnabled [private]

7.5.5.182 THGROUP_HANDLE Atlantis::thg_main [private]

7.5.5.183 THGROUP_HANDLE Atlantis::thg_srb [private]

7.5.5.184 THGROUP_HANDLE Atlantis::thg_retro [private]

7.5.5.185 CTRLSURFHANDLE Atlantis::hrudder [private]

7.5.5.186 CTRLSURFHANDLE Atlantis::hlaileron [private]

7.5.5.187 CTRLSURFHANDLE Atlantis::hraileron [private]

7.5.5.188 CTRLSURFHANDLE Atlantis::helevator [private]

7.5.5.189 CTRLSURFHANDLE Atlantis::hbodyflap [private]

7.5.5.190 bool Atlantis::ControlSurfacesEnabled [private]

7.5.5.191 bool Atlantis::bSSMEGOXVent [private]

7.5.5.192 VECTOR3 Atlantis::SSMET_GOX_REF1 [private]

7.5.5.193 VECTOR3 Atlantis::SSMEL_GOX_REF1 [private]

7.5.5.194 VECTOR3 Atlantis::SSMER_GOX_REF1 [private]

7.5.5.195 PROPELLANT_HANDLE Atlantis::apu_tank[3] [private]

7.5.5.196 PROPELLANT_HANDLE Atlantis::oms_helium_tank[2] [private]

7.5.5.197 PROPELLANT_HANDLE Atlantis::mps_helium_tank[3] [private]

7.5.5.198 int Atlantis::MPS_Manifold_Press[2] [private]

7.5.5.199 int Atlantis::MPS_He_Reg_Press[4] [private]

7.5.5.200 int Atlantis::MPS_He_Tank_Press[4] [private]

7.5.5.201 int Atlantis::Hydraulic_Press[3] [private]

7.5.5.202 ANIMATIONCOMPONENT_HANDLE Atlantis::hAC_arm [private]

7.5.5.203 ANIMATIONCOMPONENT_HANDLE Atlantis::hAC_sat [private]

7.5.5.204 ANIMATIONCOMPONENT_HANDLE Atlantis::hAC_satref [private]

7.5.5.205 MGROUP_TRANSFORM* Atlantis::rms_anim[7] [private]

7.5.5.206 MGROUP_TRANSFORM * Atlantis::rms_rollout_anim [private]

7.5.5.207 UINT Atlantis::anim_arm_sy [private]

7.5.5.208 UINT Atlantis::anim_arm_sp [private]

7.5.5.209 UINT Atlantis::anim_arm_ep [private]

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Atlantis.h

7.6 Atlantis_SRB Class Reference

```
#include <Atlantis.h>
```

Public Member Functions

- Atlantis_SRB (OBJHANDLE hObj)
- void SetRefTime (void)
- void clbkSetClassCaps (FILEHANDLE cfg)
- void clbkPostStep (double simt, double simdt, double mjd)
- void clbkPostCreation ()

Private Attributes

- MESHHANDLE hSRBMesh_Left
- MESHHANDLE hSRBMesh_Right
- double t0
- double srb_separation_time
- bool bMainEngine
- bool bSeparationEngine
- double slag1
- double slag2
- double slag3
- MESHHANDLE hSRBMesh
- PROPELLANT_HANDLE ph_main
- PROPELLANT_HANDLE phBSM
- THRUSTER_HANDLE th_main
- THRUSTER_HANDLE th_bolt
- THRUSTER_HANDLE thBSM [3]

7.6.1 Constructor & Destructor Documentation

7.6.1.1 Atlantis_SRB::Atlantis_SRB (OBJHANDLE *hObj*)

7.6.2 Member Function Documentation

7.6.2.1 void Atlantis_SRB::SetRefTime (void)

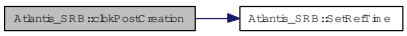
7.6.2.2 void Atlantis_SRB::clbkSetClassCaps (FILEHANDLE *cfg*)

7.6.2.3 void Atlantis_SRB::clbkPostStep (double *simt*, double *simdt*, double *mjd*)

Here is the call graph for this function:

7.6.2.4 void Atlantis_SRB::clbkPostCreation ()

Here is the call graph for this function:



7.6.3 Member Data Documentation

7.6.3.1 MESHHANDLE Atlantis_SRB::hSRBMesh_Left [private]

7.6.3.2 MESHHANDLE Atlantis_SRB::hSRBMesh_Right [private]

7.6.3.3 double Atlantis_SRB::t0 [private]

7.6.3.4 double Atlantis_SRB::srb_separation_time [private]

7.6.3.5 bool Atlantis_SRB::bMainEngine [private]

7.6.3.6 bool Atlantis_SRB::bSeparationEngine [private]

7.6.3.7 double Atlantis_SRB::slag1 [private]

7.6.3.8 double Atlantis_SRB::slag2 [private]

7.6.3.9 double Atlantis_SRB::slag3 [private]

7.6.3.10 MESHHANDLE Atlantis_SRB::hSRBMesh [private]

7.6.3.11 PROPELLANT_HANDLE Atlantis_SRB::ph_main [private]

7.6.3.12 PROPELLANT_HANDLE Atlantis_SRB::phBSM [private]

7.6.3.13 THRUSTER_HANDLE Atlantis_SRB::th_main [private]

7.6.3.14 THRUSTER_HANDLE Atlantis_SRB::th_bolt [private]

7.6.3.15 THRUSTER_HANDLE Atlantis_SRB::thBSM[3] [private]

The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Atlantis.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Atlantis_SRB.cpp

7.7 Atlantis_Tank Class Reference

```
#include <Atlantis.h>
```

Public Member Functions

- Atlantis_Tank (OBJHANDLE hObj)
- void clbkSetClassCaps (FILEHANDLE cfg)
- void clbkPostStep (double simt, double simdt, double mjd)

Private Attributes

- MESHHANDLE hTankMesh

7.7.1 Constructor & Destructor Documentation

7.7.1.1 Atlantis_Tank::Atlantis_Tank (OBJHANDLE *hObj*)

7.7.2 Member Function Documentation

7.7.2.1 void Atlantis_Tank::clbkSetClassCaps (FILEHANDLE *cfg*)

7.7.2.2 void Atlantis_Tank::clbkPostStep (double *simt*, double *simdt*, double *mjd*)

7.7.3 Member Data Documentation

7.7.3.1 MESHHANDLE Atlantis_Tank::hTankMesh [private]

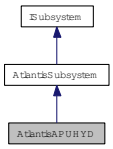
The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Atlantis.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Atlantis_Tank.cpp

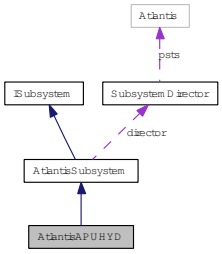
7.8 AtlantisAPUHYD Class Reference

`#include <AtlantisAPUHYD.h>`

Inheritance diagram for AtlantisAPUHYD:



Collaboration diagram for AtlantisAPUHYD:



Public Member Functions

- virtual void clbkPreStep (double fSimT, double fDeltaT)
- double GetGasGeneratorTemperature (APU_INDEX apu_index) const
- double GetAPUTurbineRPM (APU_INDEX apu_index) const
- virtual void clbkPostStep (double fSimT, double fDeltaT)
- AtlantisAPUHYD (SubsystemDirector *_director)
- virtual ~AtlantisAPUHYD ()

Protected Member Functions

- void CalculateAPU (APU_INDEX apu_index)

Private Attributes

- bool fValve1Fail [3]
- bool fValve2Fail [3]
- double fValve1Pulse [3]
- double fValve2Pulse [3]
- double fTurbineRPM [3]
- double fGGTemperature [3]

- double fGGBedTemp [3]
- double fLubeOilTemp [3]
- double fAPUStartUpDelay [3]
- bool fAPUControllerPower [3]

7.8.1 Constructor & Destructor Documentation

7.8.1.1 AtlantisAPUHYD::AtlantisAPUHYD (SubsystemDirector * *_director*)

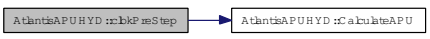
7.8.1.2 AtlantisAPUHYD::~AtlantisAPUHYD () [virtual]

7.8.2 Member Function Documentation

7.8.2.1 void AtlantisAPUHYD::CalculateAPU (APU_INDEX *apu_index*) [protected]

7.8.2.2 void AtlantisAPUHYD::clbkPreStep (double *fSimT*, double *fDeltaT*) [virtual]

Here is the call graph for this function:



7.8.2.3 double AtlantisAPUHYD::GetGasGeneratorTemperature (APU_INDEX *apu_index*) const

7.8.2.4 double AtlantisAPUHYD::GetAPUTurbineRPM (APU_INDEX *apu_index*) const

7.8.2.5 void AtlantisAPUHYD::clbkPostStep (double *fSimT*, double *fDeltaT*) [virtual]

7.8.3 Member Data Documentation

7.8.3.1 bool AtlantisAPUHYD::fValve1Fail[3] [private]

7.8.3.2 bool AtlantisAPUHYD::fValve2Fail[3] [private]

7.8.3.3 double AtlantisAPUHYD::fValve1Pulse[3] [private]

7.8.3.4 double AtlantisAPUHYD::fValve2Pulse[3] [private]

7.8.3.5 double AtlantisAPUHYD::fTurbineRPM[3] [private]

7.8.3.6 double AtlantisAPUHYD::fGGTemperature[3] [private]

7.8.3.7 double AtlantisAPUHYD::fGGBedTemp[3] [private]

7.8.3.8 double AtlantisAPUHYD::fLubeOilTemp[3] [private]

7.8.3.9 double AtlantisAPUHYD::fAPUStartUpDelay[3] [private]

7.8.3.10 bool AtlantisAPUHYD::fAPUControllerPower[3] [private]

The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AtlantisAPUHYD.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AtlantisAPUHYD.cpp

7.9 AtlantisConfig Class Reference

Public Member Functions

- AtlantisConfig ()
- char ∗ Name ()
- char ∗ Description ()
- bool clbkOpen (HWND hLaunchpad)
- bool TexHiresEnabled () const
- void TexEnableHires (bool enable)
- bool MshHiresEnabled () const
- bool MshEnableHires (bool enable)
- void InitDialog (HWND hWnd)
- void Apply (HWND hWnd)

Static Public Member Functions

- static BOOL CALLBACK DlgProc (HWND, UINT, WPARAM, LPARAM)

7.9.1 Constructor & Destructor Documentation

7.9.1.1 AtlantisConfig::AtlantisConfig () [inline]

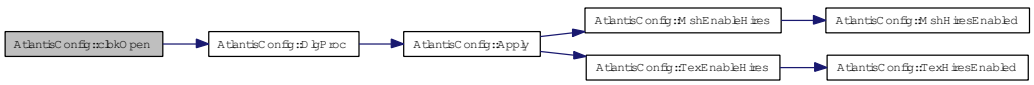
7.9.2 Member Function Documentation

7.9.2.1 char∗ AtlantisConfig::Name () [inline]

7.9.2.2 char ∗ AtlantisConfig::Description ()

7.9.2.3 bool AtlantisConfig::clbkOpen (HWND *hLaunchpad*)

Here is the call graph for this function:



7.9.2.4 bool AtlantisConfig::TexHiresEnabled () const

7.9.2.5 void AtlantisConfig::TexEnableHires (bool *enable*)

Here is the call graph for this function:

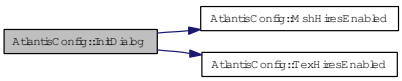
7.9.2.6 bool AtlantisConfig::MshHiresEnabled () const

7.9.2.7 bool AtlantisConfig::MshEnableHires (bool *enable*)

Here is the call graph for this function:

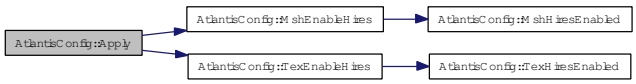
7.9.2.8 void AtlantisConfig::InitDialog (HWND *hWnd*)

Here is the call graph for this function:



7.9.2.9 void AtlantisConfig::Apply (HWND *hWnd*)

Here is the call graph for this function:



7.9.2.10 BOOL CALLBACK AtlantisConfig::DlgProc (HWND *hWnd*, UINT *uMsg*, WPARAM *wParam*, LPARAM *lParam*) [static]

Here is the call graph for this function:

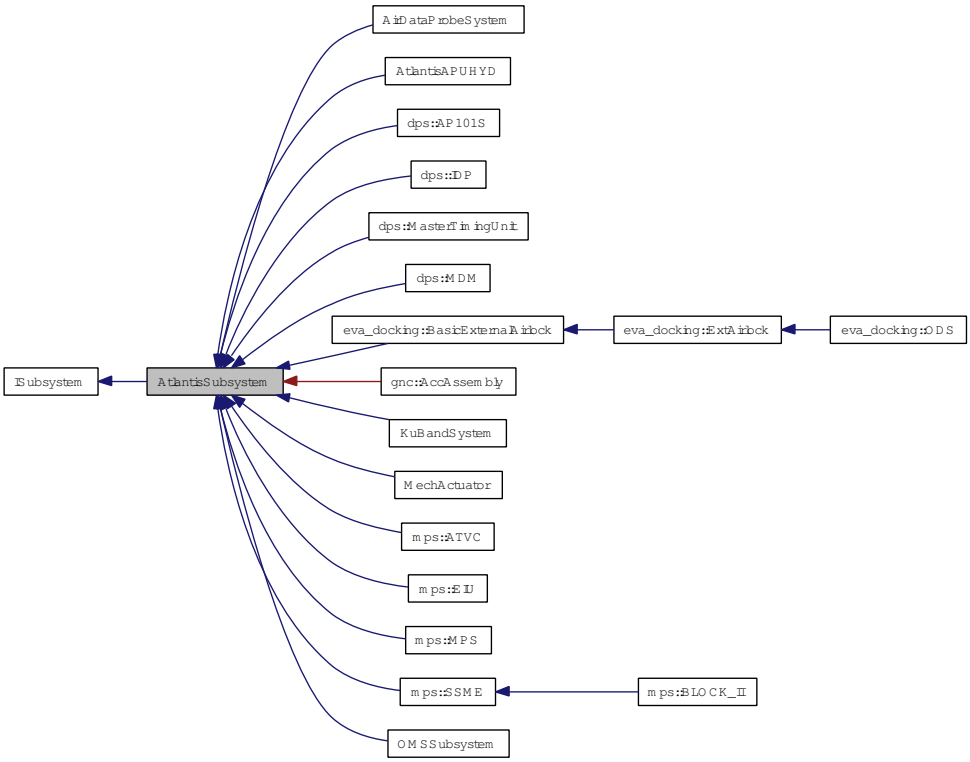
The documentation for this class was generated from the following file:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AtlantisConfig.cpp

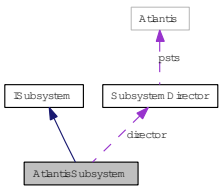
7.10 AtlantisSubsystem Class Reference

`#include <AtlantisSubsystem.h>`

Inheritance diagram for AtlantisSubsystem:



Collaboration diagram for AtlantisSubsystem:



Public Member Functions

- AtlantisSubsystem (SubsystemDirector *_director, const string &_ident)
- virtual ~AtlantisSubsystem ()
- Atlantis * STS ()
- virtual void AddMeshes (const VECTOR3 &ofs)

- virtual void Realize ()
- virtual bool RegisterComponent (SubsystemComponent ∗pComponent)
- virtual bool DelComponent (SubsystemComponent ∗pComponent)
- virtual unsigned long CountComponents () const
- virtual void OnPreStep (double fSimT, double fDeltaT, double fMJD)
- virtual void OnPostStep (double fSimT, double fDeltaT, double fMJD)
- virtual void OnPropagate (double fSimT, double fDeltaT, double fMJD)
- virtual void OnPlaybackEvent (double fSimT, double fEventT, const char ∗event_t, const char ∗event)
- virtual bool OnParseLine (const char ∗line)
- virtual void OnSaveState (FILEHANDLE scn) const
- virtual void OnDumpToLog () const
- virtual const string & GetQualifiedIdentifier () const
- virtual const string & GetIdentifier () const
- virtual bool OnFail (const string &strFailureCode)
- virtual double GetSubsystemEmptyMass () const

Protected Attributes

- SubsystemDirector ∗ director
- string ident

7.10.1 Detailed Description

base class for all subsystem simulations.

7.10.2 Constructor & Destructor Documentation

7.10.2.1 AtlantisSubsystem::AtlantisSubsystem (SubsystemDirector ∗ *_director*, const string & *_ident*)

Create a new Atlantis subsystem object

7.10.2.2 AtlantisSubsystem::∼AtlantisSubsystem () `[virtual]`

Destructor

7.10.3 Member Function Documentation

7.10.3.1 Atlantis ∗ AtlantisSubsystem::STS ()

Get a reference to the Space Shuttle external physical model.

Returns:

reference to the space shuttle object. If possible.

Here is the call graph for this function:

7.10.3.2 void AtlantisSubsystem::AddMeshes (const VECTOR3 & *ofs*) [virtual]

Add new meshes to the visual.

Deprecated

Don't use, instead, use Realize() to tell the Atlantis class which meshes it should use for special slots.

Reimplemented in eva_docking::BasicExternalAirlock, eva_docking::ExtAirlock, and eva_docking::ODS.

7.10.3.3 void AtlantisSubsystem::Realize () [virtual]

Create the physical representation of the subsystem. This means:

- creation of thrusters
- discrete line connections to VC components
- discrete line connections to other subsystems
- connection to Shuttle Bus systems
- hydraulic connections
- coolant loop connections
- connections to logical propellant resources
-

Implements ISubsystem.

Reimplemented in eva_docking::BasicExternalAirlock, and mps::SSME.

7.10.3.4 bool AtlantisSubsystem::RegisterComponent (SubsystemComponent * *pComponent*) [virtual]

Register a subsystem component for use inside this subsystem for automated handling. Not in use now.

Warning:

Does not have an effect.

Implements ISubsystem.

7.10.3.5 bool AtlantisSubsystem::DelComponent (SubsystemComponent * *pComponent*) [virtual]

Delete a subsystem component, if no longer used

Warning:

Does not have an effect.

Implements ISubsystem.

7.10.3.6 unsigned long AtlantisSubsystem::CountComponents () const [virtual]

Count the number of subsystem components.

Returns:

always zero, as subsystem component handling is not active.

Implements ISubsystem.

7.10.3.7 void AtlantisSubsystem::OnPreStep (double *fSimT*, double *fDeltaT*, double *fMJD*) [virtual]

Prepare and execute calculations before the propagation of the state vector

Parameters:

fSimT current simulation time

fDeltaT Simulation time step length

fMJD current modified julian date

Implements ISubsystem.

Reimplemented in AirDataProbeSystem, dps::MasterTimingUnit, eva_docking::BasicExternalAirlock, and MechActuator.

7.10.3.8 void AtlantisSubsystem::OnPostStep (double *fSimT*, double *fDeltaT*, double *fMJD*) [virtual]

Finish calculations after the propagation of the state vector

Parameters:

fSimT current simulation time

fDeltaT Simulation time step length

fMJD current modified julian date

Implements ISubsystem.

Reimplemented in AirDataProbeSystem, eva_docking::BasicExternalAirlock, MechActuator, mps::ATVC, mps::EIU, mps::MPS, and mps::SSME.

7.10.3.9 void AtlantisSubsystem::OnPropagate (double *fSimT*, double *fDeltaT*, double *fMJD*) [virtual]

Propagate the state vector from old state to new state.

Parameters:

fSimT current simulation time

fDeltaT Simulation time step length

fMJD current modified julian date

Implements ISubsystem.

Reimplemented in AirDataProbeSystem, dps::MasterTimingUnit, eva_docking::BasicExternalAirlock, and MechActuator.

7.10.3.10 void AtlantisSubsystem::OnPlaybackEvent (double *fSimT*, double *fEventT*, const char * *event_t*, const char * *event*) [virtual]

Implements ISubsystem.

Reimplemented in eva_docking::BasicExternalAirlock.

7.10.3.11 bool AtlantisSubsystem::OnParseLine (const char * *line*) [virtual]

Parse a line of the scenario file.

Warning:

Should get replaced by a block oriented scenario file format, like NASSP, making this function deprecated.

Implements ISubsystem.

Reimplemented in dps::MasterTimingUnit, eva_docking::BasicExternalAirlock, mps::BLOCK_II, mps::MPS, and mps::SSME.

7.10.3.12 void AtlantisSubsystem::OnSaveState (FILEHANDLE *scn*) const [virtual]

Save the state of the subsystem

Implements ISubsystem.

Reimplemented in dps::MasterTimingUnit, eva_docking::BasicExternalAirlock, mps::BLOCK_II, mps::MPS, and mps::SSME.

7.10.3.13 void AtlantisSubsystem::OnDumpToLog () const [virtual]

Dump the current state and error conditions into the Orbiter.log or a special SSU.log. Don't modificate state.

Implements ISubsystem.

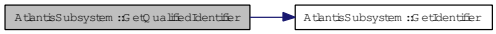
Reimplemented in eva_docking::BasicExternalAirlock.

7.10.3.14 const string & AtlantisSubsystem::GetQualifiedIdentifier () const [virtual]

get the full identifier of the subsystem, including parent container, if needed. Usually the same as GetIdentifier.

Implements ISubsystem.

Here is the call graph for this function:



7.10.3.15 const string & AtlantisSubsystem::GetIdentifier () const [virtual]

get the identifier of the subsystem. This identifier can be used for locating the subsystem inside the physical model AND the scenario file. should not contain white spaces.

Implements ISubsystem.

7.10.3.16 bool AtlantisSubsystem::OnFail (const string & *strFailureCode*) [virtual]

Entry point for triggering timed failures. Will be used for failure simulation.

Implements ISubsystem.

7.10.3.17 virtual double AtlantisSubsystem::GetSubsystemEmptyMass () const [inline, virtual]

Get the dry mass of the subsystem, if this subsystem is optional and causes changes to the standard orbiter dry mass. For example RMS, external airlock/ODS, etc.

Returns:

The subsystem dry mass in kg.

Reimplemented in eva_docking::BasicExternalAirlock, and eva_docking::ODS.

7.10.4 Member Data Documentation

7.10.4.1 SubsystemDirector* AtlantisSubsystem::director [protected]

7.10.4.2 string AtlantisSubsystem::ident [protected]

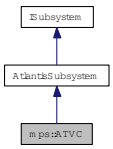
The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AtlantisSubsystem.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AtlantisSubsystem.cpp

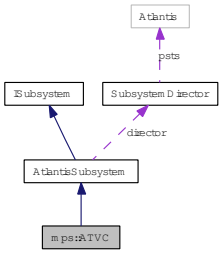
7.11 mps::ATVC Class Reference

`#include <ATVC.h>`

Inheritance diagram for mps::ATVC:



Collaboration diagram for mps::ATVC:



Public Member Functions

- ATVC (SubsystemDirector ∗)
- ∼ATVC (void)
- virtual void OnPostStep (double, double, double)

7.11.1 Constructor & Destructor Documentation

7.11.1.1 mps::ATVC::ATVC (SubsystemDirector ∗ *_director*)

7.11.1.2 mps::ATVC::∼ATVC (void)

7.11.2 Member Function Documentation

7.11.2.1 void mps::ATVC::OnPostStep (double *fSimT*, double *fDeltaT*, double *fMJD*) [virtual]

Finish calculations after the propagation of the state vector

Parameters:

fSimT current simulation time

fDeltaT Simulation time step length

fMJD current modified julian date

Reimplemented from AtlantisSubsystem.

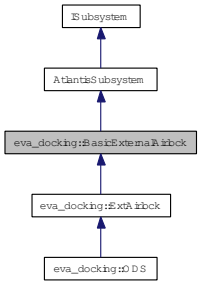
The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/ATVC.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/ATVC.cpp

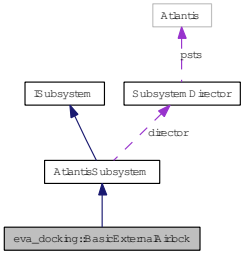
7.12 eva_docking::BasicExternalAirlock Class Reference

`#include <BasicExtAirlock.h>`

Inheritance diagram for eva_docking::BasicExternalAirlock:



Collaboration diagram for eva_docking::BasicExternalAirlock:



Public Member Functions

- BasicExternalAirlock (SubsystemDirector *pdirect, const string &ident)
- virtual ~BasicExternalAirlock ()
- virtual void AddMeshes (const VECTOR3 &ofs)
- virtual void Realize ()
- virtual double GetSubsystemEmptyMass () const
- virtual void OnPreStep (double fSimT, double fDeltaT, double fMJD)
- virtual void OnPostStep (double fSimT, double fDeltaT, double fMJD)
- virtual void OnPropagate (double fSimT, double fDeltaT, double fMJD)
- virtual void OnPlaybackEvent (double fSimT, double fEventT, const char *event_t, const char *event)
- virtual bool OnParseLine (const char *line)
- virtual void OnSaveState (FILEHANDLE scn) const
- virtual void OnDumpToLog () const

7.12.1 Constructor & Destructor Documentation

7.12.1.1 eva_docking::BasicExternalAirlock::BasicExternalAirlock (SubsystemDirector * *pdirect*, const string & *ident*)

7.12.1.2 eva_docking::BasicExternalAirlock::~BasicExternalAirlock () [virtual]

7.12.2 Member Function Documentation

7.12.2.1 void eva_docking::BasicExternalAirlock::AddMeshes (const VECTOR3 & *ofs*) [virtual]

Add new meshes to the visual.

Deprecated

Don't use, instead, use Realize() to tell the Atlantis class which meshes it should use for special slots.

Reimplemented from AtlantisSubsystem.

Reimplemented in eva_docking::ExtAirlock, and eva_docking::ODS.

7.12.2.2 void eva_docking::BasicExternalAirlock::Realize () [virtual]

Create the physical representation of the subsystem. This means:

- creation of thrusters
- discrete line connections to VC components
- discrete line connections to other subsystems
- connection to Shuttle Bus systems
- hydraulic connections
- coolant loop connections
- connections to logical propellant resources
-

Reimplemented from AtlantisSubsystem.

7.12.2.3 double eva_docking::BasicExternalAirlock::GetSubsystemEmptyMass () const [virtual]

Get the dry mass of the subsystem, if this subsystem is optional and causes changes to the standard orbiter dry mass. For example RMS, external airlock/ODS, etc.

Returns:

The subsystem dry mass in kg.

Reimplemented from AtlantisSubsystem.

Reimplemented in eva_docking::ODS.

7.12.2.4 void eva_docking::BasicExternalAirlock::OnPreStep (double *fSimT*, double *fDeltaT*, double *fMJD*) [virtual]

Prepare and execute calculations before the propagation of the state vector

Parameters:

fSimT current simulation time
fDeltaT Simulation time step length
fMJD current modified julian date

Reimplemented from AtlantisSubsystem.

7.12.2.5 void eva_docking::BasicExternalAirlock::OnPostStep (double *fSimT*, double *fDeltaT*, double *fMJD*) [virtual]

Finish calculations after the propagation of the state vector

Parameters:

fSimT current simulation time
fDeltaT Simulation time step length
fMJD current modified julian date

Reimplemented from AtlantisSubsystem.

7.12.2.6 void eva_docking::BasicExternalAirlock::OnPropagate (double *fSimT*, double *fDeltaT*, double *fMJD*) [virtual]

Propagate the state vector from old state to new state.

Parameters:

fSimT current simulation time
fDeltaT Simulation time step length
fMJD current modified julian date

Reimplemented from AtlantisSubsystem.

7.12.2.7 void eva_docking::BasicExternalAirlock::OnPlaybackEvent (double *fSimT*, double *fEventT*, const char * *event_t*, const char * *event*) [virtual]

Reimplemented from AtlantisSubsystem.

7.12.2.8 bool eva_docking::BasicExternalAirlock::OnParseLine (const char * *line*) [virtual]

Parse a line of the scenario file.

Warning:

Should get replaced by a block oriented scenario file format, like NASSP, making this function deprecated.

Reimplemented from AtlantisSubsystem.

7.12.2.9 void eva_docking::BasicExternalAirlock::OnSaveState (FILEHANDLE *scn*) const [virtual]

Save the state of the subsystem

Reimplemented from AtlantisSubsystem.

7.12.2.10 void eva_docking::BasicExternalAirlock::OnDumpToLog () const [virtual]

Dump the current state and error conditions into the Orbiter.log or a special SSU.log. Don't modificate state.

Reimplemented from AtlantisSubsystem.

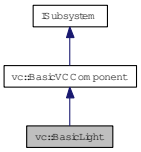
The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/eva_docking/BasicExtAirlock.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/eva_docking/BasicExtAirlock.cpp

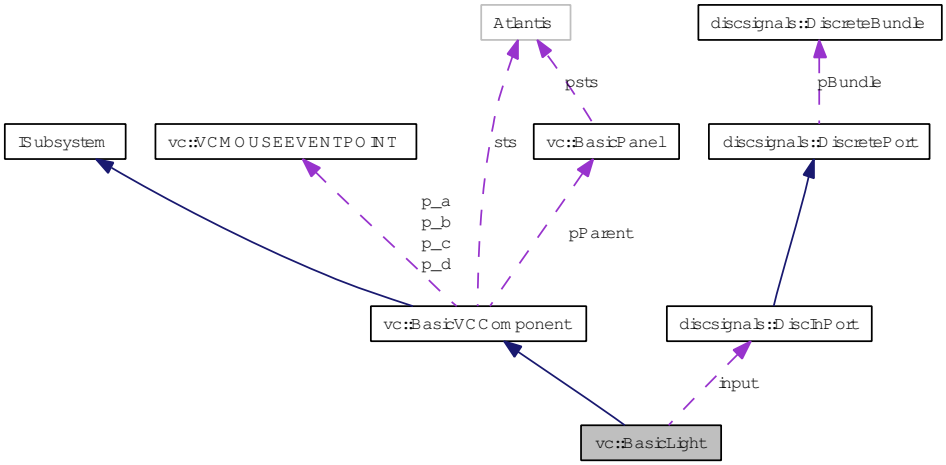
7.13 vc::BasicLight Class Reference

`#include <BasicLight.h>`

Inheritance diagram for vc::BasicLight:



Collaboration diagram for vc::BasicLight:



Public Member Functions

- BasicLight (Atlantis *_sts, const string &_ident)
- virtual ~BasicLight ()
- virtual bool Connect (unsigned short usPort,::discsignals::DiscreteBundle *pBundle, unsigned short usLine)

Private Attributes

- DiscInPort input

7.13.1 Constructor & Destructor Documentation

7.13.1.1 vc::BasicLight::BasicLight (Atlantis * *_sts*, const string & *_ident*)

7.13.1.2 virtual vc::BasicLight::~BasicLight () [virtual]

7.13.2 Member Function Documentation

7.13.2.1 virtual bool vc::BasicLight::Connect (unsigned short *usPort*, ::discsignals::DiscreteBundle * *pBundle*, unsigned short *usLine*) [virtual]

7.13.3 Member Data Documentation

7.13.3.1 DiscInPort vc::BasicLight::input [private]

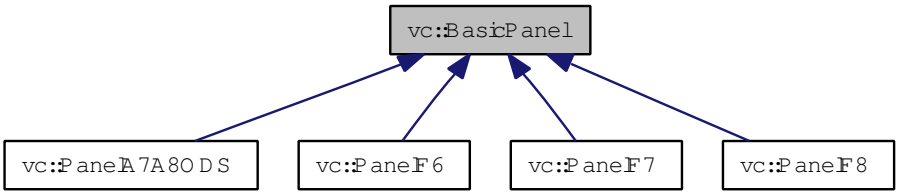
The documentation for this class was generated from the following file:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicLight.h

7.14 vc::BasicPanel Class Reference

`#include <BasicPanel.h>`

Inheritance diagram for vc::BasicPanel:



Collaboration diagram for vc::BasicPanel:

Public Member Functions

- BasicPanel (Atlantis *sts, const string &_name)
- virtual ~BasicPanel ()
- Atlantis * STS () const
- virtual void DefineVCAnimations (UINT vcidx)
- virtual void RegisterVC ()
- virtual void DefineVC ()=0
- virtual UINT GetVCMeshIndex () const
- virtual void ReplaceComponent (unsigned long ulIndex, BasicVCComponent *pReplacement)
- void SetAID (unsigned long aid)
- virtual BasicVCComponent * Add (BasicVCComponent *pNew)
- virtual void AddMeshes (const VECTOR3 &ofs)
- virtual void Realize ()
- virtual bool OnVCMouseEvent (int id, int _event, VECTOR3 &p)
- virtual bool OnVCRedrawEvent (int id, int _event, SURFHANDLE surf)
- virtual void OnPreStep (double fSimT, double fDeltaT, double fMJD)
- virtual void OnPostStep (double fSimT, double fDeltaT, double fMJD)
- virtual void OnPropagate (double fSimT, double fDeltaT, double fMJD)
- virtual bool OnParseLine (char *line)
- virtual void OnSaveState (FILEHANDLE scn)
- virtual const string & GetIdentifier () const
- virtual const string & GetQualifiedIdentifier () const

- virtual bool HasSwitch (const string &switchID)
- bool HasOwnVCMesh () const
- bool EnableCoordinateDisplayMode ()
- bool DisableCoordinateDisplayMode ()
- bool ToggleCoordinateDisplayMode ()

Protected Member Functions

- BasicSwitch * CreateSwitch2 (const string &_name, const VECTOR3 &_RefPos, UINT _GrpNum, RECT _r)
- BasicSwitch * CreateSwitch3 (const string &_name, const VECTOR3 &_RefPos, UINT _GrpNum, RECT _r)
- BasicSwitch * CreateSwitch3H (const string &_name, const VECTOR3 &_RefPos, UINT _GrpNum, RECT _r)
- virtual void DefinePanelMouseArea ()
- virtual void DefineSwitches ()
- void SetHasOwnVCMesh ()
- void AddAIDToMouseEventList (UINT aid)
- void DeleteAllComponents ()
- bool DistributeMouseEvent (int _event, const VECTOR3 &p)
- bool FindComponent (const VECTOR3 &p, BasicVCComponent **foundElement) const

Private Attributes

- UINT local_vcidx
- Atlantis * psts
- unsigned long own_aid
- string name
- set< UINT > availableForMouse
- vector< BasicVCComponent * > components
- bool bHasOwnVCMesh
- bool bCoordinateDisplayMode

7.14.1 Constructor & Destructor Documentation

7.14.1.1 vc::BasicPanel::BasicPanel (Atlantis * *sts*, const string & *_name*)

7.14.1.2 vc::BasicPanel::~BasicPanel () [virtual]

Here is the call graph for this function:

7.14.2 Member Function Documentation

7.14.2.1 BasicSwitch* vc::BasicPanel::CreateSwitch2 (const string & *_name*, const VECTOR3 & *_RefPos*, UINT *_GrpNum*, RECT *_r*) [protected]

7.14.2.2 BasicSwitch* vc::BasicPanel::CreateSwitch3 (const string & *_name*, const VECTOR3 & *_RefPos*, UINT *_GrpNum*, RECT *_r*) [protected]

7.14.2.3 BasicSwitch* vc::BasicPanel::CreateSwitch3H (const string & *_name*, const VECTOR3 & *_RefPos*, UINT *_GrpNum*, RECT *_r*) [protected]

7.14.2.4 void vc::BasicPanel::DefinePanelMouseArea () [protected, virtual]

7.14.2.5 void vc::BasicPanel::DefineSwitches () [protected, virtual]

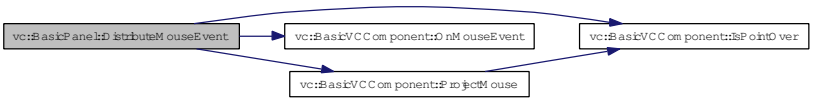
7.14.2.6 void vc::BasicPanel::SetHasOwnVCMesh () [inline, protected]

7.14.2.7 void vc::BasicPanel::AddAIDToMouseEventList (UINT *aid*) [protected]

7.14.2.8 void vc::BasicPanel::DeleteAllComponents () [protected]

7.14.2.9 bool vc::BasicPanel::DistributeMouseEvent (int *_event*, const VECTOR3 & *p*) [protected]

Here is the call graph for this function:



7.14.2.10 bool vc::BasicPanel::FindComponent (const VECTOR3 & *p*, BasicVCComponent ** *foundElement*) const [protected]

Here is the call graph for this function:

7.14.2.11 Atlantis * vc::BasicPanel::STS () const

7.14.2.12 void vc::BasicPanel::DefineVCAnimations (UINT *vcidx*) [virtual]

Reimplemented in vc::PanelA7A8ODS.

Here is the call graph for this function:

7.14.2.13 void vc::BasicPanel::RegisterVC () [virtual]

Reimplemented in vc::PanelA7A8ODS, vc::PanelF6, vc::PanelF7, and vc::PanelF8.

7.14.2.14 virtual void vc::BasicPanel::DefineVC () [pure virtual]

Implemented in vc::PanelA7A8ODS, vc::PanelF6, vc::PanelF7, and vc::PanelF8.

7.14.2.15 UINT vc::BasicPanel::GetVCMeshIndex () const [virtual]

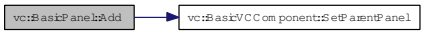
Reimplemented in vc::PanelA7A8ODS.

7.14.2.16 void vc::BasicPanel::ReplaceComponent (unsigned long *ulIndex*, BasicVCComponent * *pReplacement*) [virtual]

7.14.2.17 void vc::BasicPanel::SetAID (unsigned long *aid*) [inline]

7.14.2.18 BasicVCComponent * vc::BasicPanel::Add (BasicVCComponent * *pNew*) [virtual]

Here is the call graph for this function:



7.14.2.19 virtual void vc::BasicPanel::AddMeshes (const VECTOR3 & *ofs*) [inline, virtual]

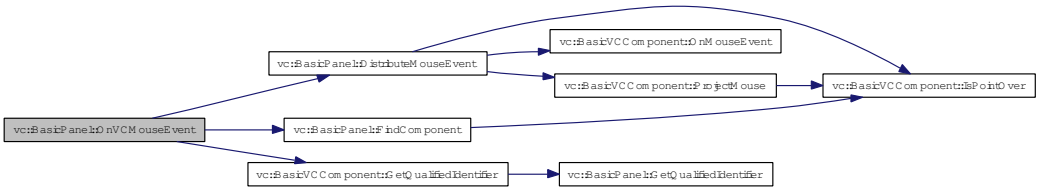
Reimplemented in vc::PanelA7A8ODS.

7.14.2.20 void vc::BasicPanel::Realize () [virtual]

Reimplemented in vc::PanelA7A8ODS.

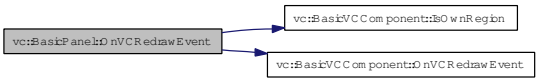
7.14.2.21 bool vc::BasicPanel::OnVCMouseEvent (int *id*, int *_event*, VECTOR3 & *p*) [virtual]

Here is the call graph for this function:



7.14.2.22 bool vc::BasicPanel::OnVCRedrawEvent (int *id*, int *_event*, SURFHANDLE *surf*) [virtual]

Here is the call graph for this function:



7.14.2.23 void vc::BasicPanel::OnPreStep (double *fSimT*, double *fDeltaT*, double *fMJD*) [virtual]

7.14.2.24 void vc::BasicPanel::OnPostStep (double *fSimT*, double *fDeltaT*, double *fMJD*) [virtual]

7.14.2.25 void vc::BasicPanel::OnPropagate (double *fSimT*, double *fDeltaT*, double *fMJD*) [virtual]

7.14.2.26 bool vc::BasicPanel::OnParseLine (char * *line*) [virtual]

7.14.2.27 void vc::BasicPanel::OnSaveState (FILEHANDLE *scn*) [virtual]

7.14.2.28 const string & vc::BasicPanel::GetIdentifier () const [virtual]

7.14.2.29 const string & vc::BasicPanel::GetQualifiedIdentifier () const [virtual]

7.14.2.30 bool vc::BasicPanel::HasSwitch (const string & *switchID*) [virtual]

7.14.2.31 bool vc::BasicPanel::HasOwnVCMesh () const [inline]

7.14.2.32 bool vc::BasicPanel::EnableCoordinateDisplayMode () [inline]

7.14.2.33 bool vc::BasicPanel::DisableCoordinateDisplayMode () [inline]

7.14.2.34 bool vc::BasicPanel::ToggleCoordinateDisplayMode () [inline]

7.14.3 Member Data Documentation

7.14.3.1 UINT vc::BasicPanel::local_vcidx [private]

7.14.3.2 Atlantis* vc::BasicPanel::psts [private]

7.14.3.3 unsigned long vc::BasicPanel::own_aid [private]

7.14.3.4 string vc::BasicPanel::name [private]

7.14.3.5 set<UINT> vc::BasicPanel::availableForMouse [private]

7.14.3.6 vector<BasicVCComponent*> vc::BasicPanel::components [private]

7.14.3.7 bool vc::BasicPanel::bHasOwnVCMesh [private]

7.14.3.8 bool vc::BasicPanel::bCoordinateDisplayMode [private]

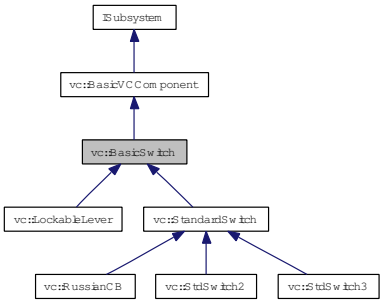
The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicPanel.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicPanel.cpp

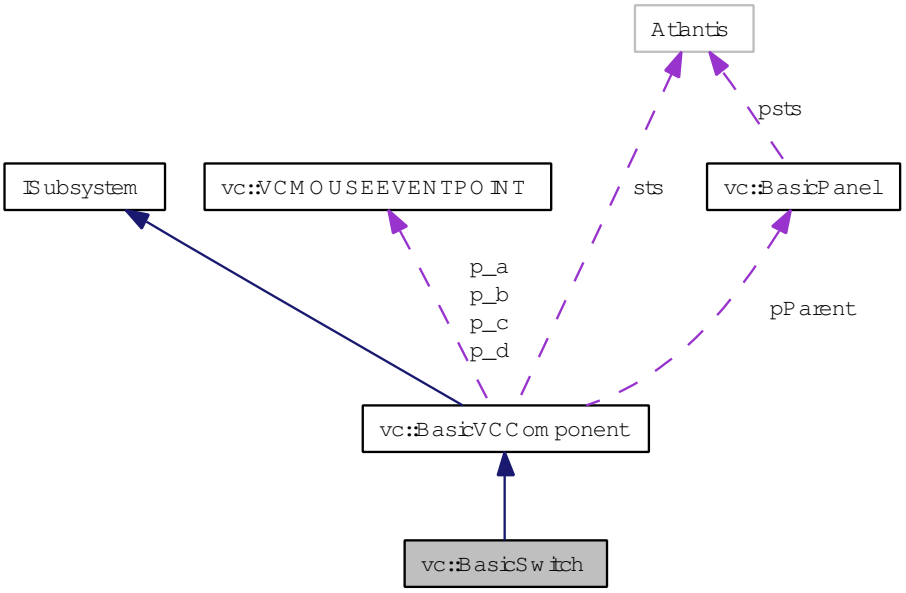
7.15 vc::BasicSwitch Class Reference

`#include <BasicSwitch.h>`

Inheritance diagram for vc::BasicSwitch:



Collaboration diagram for vc::BasicSwitch:



Public Member Functions

- BasicSwitch (Atlantis *_sts, unsigned short usNumPositions, const string &_ident)
- virtual ~BasicSwitch ()
- const string & GetLabel (int iPosition) const

- void SetOrientation (bool bHorizontal)
- virtual void DefineVCAnimations (UINT vc_idx)
- virtual bool OnMouseEvent (int _event, float x, float y)
- virtual void OnPositionUp ()
- virtual void OnPositionDown ()
- virtual void OnPositionChange (unsigned short usNewPosition)
- virtual void OnSaveState (FILEHANDLE scn) const

Protected Attributes

- bool bOrientation
- unsigned short usNumPositions
- vector< string > labels
- unsigned short usCurrentPosition

7.15.1 Constructor & Destructor Documentation

7.15.1.1 vc::BasicSwitch::BasicSwitch (Atlantis * *_sts*, unsigned short *usNumPositions*, const string & *_ident*)

7.15.1.2 vc::BasicSwitch::~BasicSwitch () [virtual]

7.15.2 Member Function Documentation

7.15.2.1 const string& vc::BasicSwitch::GetLabel (int *iPosition*) const

7.15.2.2 void vc::BasicSwitch::SetOrientation (bool *bHorizontal*)

7.15.2.3 void vc::BasicSwitch::DefineVCAnimations (UINT *vc_idx*) [virtual]

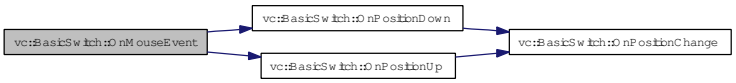
Reimplemented from vc::BasicVCComponent.

Reimplemented in vc::StandardSwitch, and vc::RussianCB.

7.15.2.4 bool vc::BasicSwitch::OnMouseEvent (int *_event*, float *x*, float *y*) [virtual]

Reimplemented from vc::BasicVCComponent.

Here is the call graph for this function:



7.15.2.5 void vc::BasicSwitch::OnPositionUp () [virtual]

Here is the call graph for this function:

7.15.2.6 void vc::BasicSwitch::OnPositionDown () [virtual]

Here is the call graph for this function:

7.15.2.7 void vc::BasicSwitch::OnPositionChange (unsigned short *usNewPosition*) [virtual]

Reimplemented in vc::StandardSwitch.

7.15.2.8 void vc::BasicSwitch::OnSaveState (FILEHANDLE *scn*) const [virtual]

Reimplemented from vc::BasicVCComponent.

7.15.3 Member Data Documentation

7.15.3.1 bool vc::BasicSwitch::bOrientation [protected]

7.15.3.2 unsigned short vc::BasicSwitch::usNumPositions [protected]

7.15.3.3 vector<string> vc::BasicSwitch::labels [protected]

7.15.3.4 unsigned short vc::BasicSwitch::usCurrentPosition [protected]

The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicSwitch.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicSwitch.cpp

7.16 BasicValve Class Reference

```
#include <BasicValve.h>
```

Public Member Functions

- bool Open (double)
- bool Open (void)
- bool Close (double)
- bool Close (void)
- bool Move (double)
- bool Move (double, double)
- double GetPos (void)
- void tmestp (double)
- void _backdoor (double)
- BasicValve (double, double)
- ~BasicValve (void)

Private Attributes

- double pos
- double maxrate
- double mpos
- double mrate

7.16.1 Constructor & Destructor Documentation

7.16.1.1 BasicValve::BasicValve (double *initpos*, double *imaxrate*)

7.16.1.2 BasicValve::~BasicValve (void)

7.16.2 Member Function Documentation

7.16.2.1 bool BasicValve::Open (double *rate*)

7.16.2.2 bool BasicValve::Open (void)

7.16.2.3 bool BasicValve::Close (double *rate*)

7.16.2.4 bool BasicValve::Close (void)

7.16.2.5 bool BasicValve::Move (double *tpos*)

7.16.2.6 bool BasicValve::Move (double *tpos*, double *rate*)

7.16.2.7 double BasicValve::GetPos (void)

7.16.2.8 void BasicValve::tmestp (double *dtme*)

7.16.2.9 void BasicValve::_backdoor (double *ipos*)

7.16.3 Member Data Documentation

7.16.3.1 double BasicValve::pos [private]

7.16.3.2 double BasicValve::maxrate [private]

7.16.3.3 double BasicValve::mpos [private]

7.16.3.4 double BasicValve::mrate [private]

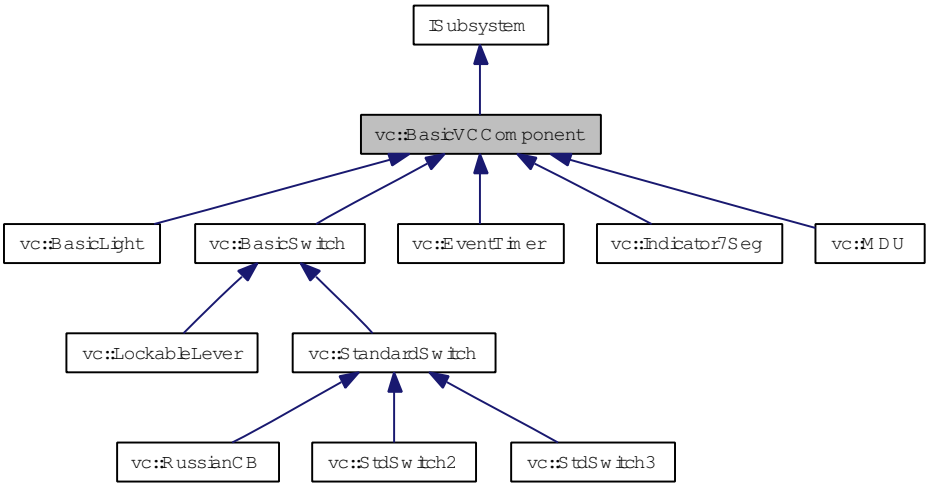
The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/BasicValve.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/BasicValve.cpp

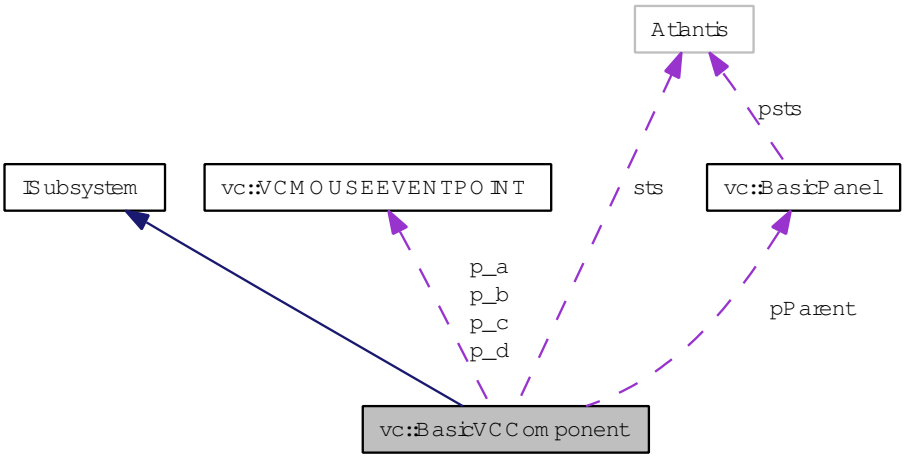
7.17 vc::BasicVCComponent Class Reference

`#include <BasicVCComponent.h>`

Inheritance diagram for vc::BasicVCComponent:



Collaboration diagram for vc::BasicVCComponent:



Public Member Functions

- BasicVCComponent (Atlantis *_sts)
- BasicVCComponent (Atlantis *_sts, const string &_ident)
- virtual ~BasicVCComponent ()
- void AddAIDToRedrawEventList (UINT aid)
- bool SetMouseRegion (float xmin, float ymin, float xmax, float ymax)
- void SetParentPanel (BasicPanel *pPanel)
- virtual void SetInitialAnimState (double fState=0.0)
- virtual void DefineVCAnimations (UINT vc_idx)
- virtual bool OnMouseEvent (int _event, float x, float y)
- virtual double GetState () const
- virtual const VECTOR3 & GetReference () const
- virtual const VECTOR3 & GetDirection () const
- bool IsOwnRegion (UINT aid) const
- bool IsPointOver (float x, float y) const
- bool ProjectMouse (float x, float y, float &mx, float &my) const
- void SetReference (const VECTOR3 &_ref)
- void SetReference (const VECTOR3 &_ref, const VECTOR3 &dir)
- void SetDirection (const VECTOR3 &_ref)
- virtual void Realize ()
- virtual bool RegisterComponent (SubsystemComponent *pComponent)
- virtual void RegisterVC ()
- virtual bool DelComponent (SubsystemComponent *pComponent)
- virtual unsigned long CountComponents () const
- virtual void OnPreStep (double fSimT, double fDeltaT, double fMJD)
- virtual void OnPostStep (double fSimT, double fDeltaT, double fMJD)
- virtual void OnPropagate (double fSimT, double fDeltaT, double fMJD)
- virtual void OnPlaybackEvent (double fSimT, double fEventT, const char *event_t, const char *event)
- virtual bool OnVCRedrawEvent (int id, int _event, SURFHANDLE surf)
- virtual bool OnParseLine (const char *line)
- virtual void OnSaveState (FILEHANDLE scn) const
- virtual void OnDumpToLog () const
- virtual const string & GetQualifiedIdentifier () const
- virtual const string & GetIdentifier () const
- virtual bool OnFail (const string &strFailureCode)

Protected Member Functions

- double InitialAnimState () const
- bool SetAnimation (UINT anim, double fState)
- Atlantis * STS () const
- bool VerifyAnimations ()
- bool WriteScenarioString (FILEHANDLE scn, const string &key, const string &value) const
- bool WriteScenarioInt (FILEHANDLE scn, const string &key, const int iValue) const
- bool WriteScenarioVector (FILEHANDLE scn, const string &key, const VECTOR3 &vector) const

Protected Attributes

- bool bHasReference
- bool bHasDirection
- bool bHasAnimations

Private Attributes

- Atlantis * sts
- BasicPanel * pParent
- string ident
- VECTOR3 reference
- VECTOR3 dir
- double fInitialAnimState
- VCMOUSEEVENTPOINT p_a
- VCMOUSEEVENTPOINT p_b
- VCMOUSEEVENTPOINT p_c
- VCMOUSEEVENTPOINT p_d
- float cache_lastx
- float cache_lasty
- float cache_mx
- float cache_my
- set< UINT > availableForRendering

7.17.1 Constructor & Destructor Documentation

7.17.1.1 vc::BasicVCComponent::BasicVCComponent (Atlantis * *_sts*)

7.17.1.2 vc::BasicVCComponent::BasicVCComponent (Atlantis * *_sts*, const string & *_ident*)

7.17.1.3 vc::BasicVCComponent::~BasicVCComponent () `[virtual]`

7.17.2 Member Function Documentation

7.17.2.1 double vc::BasicVCComponent::InitialAnimState () const `[inline, protected]`

7.17.2.2 bool vc::BasicVCComponent::SetAnimation (UINT *anim*, double *fState*) `[protected]`

Here is the call graph for this function:

7.17.2.3 Atlantis * vc::BasicVCComponent::STS () const [protected]

7.17.2.4 bool vc::BasicVCComponent::VerifyAnimations () [protected]

7.17.2.5 bool vc::BasicVCComponent::WriteScenarioString (FILEHANDLE *scn*, const string & *key*, const string & *value*) const [protected]

7.17.2.6 bool vc::BasicVCComponent::WriteScenarioInt (FILEHANDLE *scn*, const string & *key*, const int *iValue*) const [protected]

7.17.2.7 bool vc::BasicVCComponent::WriteScenarioVector (FILEHANDLE *scn*, const string & *key*, const VECTOR3 & *vector*) const [protected]

7.17.2.8 void vc::BasicVCComponent::AddAIDToRedrawEventList (UINT *aid*)

7.17.2.9 bool vc::BasicVCComponent::SetMouseRegion (float *xmin*, float *ymin*, float *xmax*, float *ymax*)

Set the mouse region of the switch inside the area. Only allows rectangular regions currently.

Parameters:

xmin The x coordinate of the left side

xmax The x coordinate of the right side

ymin The y coordinate of the top side

ymax The y coordinate of the bottom side.

7.17.2.10 void vc::BasicVCComponent::SetParentPanel (BasicPanel * *pPanel*)

7.17.2.11 void vc::BasicVCComponent::SetInitialAnimState (double *fState* = 0.0) [virtual]

7.17.2.12 void vc::BasicVCComponent::DefineVCAnimations (UINT *vc_idx*) [virtual]

Reimplemented in vc::BasicSwitch, vc::MDU, vc::StandardSwitch, and vc::RussianCB.

7.17.2.13 bool vc::BasicVCComponent::OnMouseEvent (int *_event*, float *x*, float *y*) [virtual]

Reimplemented in vc::BasicSwitch, and vc::MDU.

7.17.2.14 double vc::BasicVCComponent::GetState () const [virtual]

7.17.2.15 const VECTOR3 & vc::BasicVCComponent::GetReference () const [virtual]

7.17.2.16 const VECTOR3 & vc::BasicVCComponent::GetDirection () const [virtual]

7.17.2.17 bool vc::BasicVCComponent::IsOwnRegion (UINT *aid*) const

7.17.2.18 bool vc::BasicVCComponent::IsPointOver (float *x*, float *y*) const

7.17.2.19 bool vc::BasicVCComponent::ProjectMouse (float *x*, float *y*, float & *mx*, float & *my*) const

Here is the call graph for this function:

7.17.2.20 void vc::BasicVCComponent::SetReference (const VECTOR3 & *_ref*)

Set reference position

7.17.2.21 void vc::BasicVCComponent::SetReference (const VECTOR3 & *_ref*, const VECTOR3 & *dir*)

Set reference position and direction

7.17.2.22 void vc::BasicVCComponent::SetDirection (const VECTOR3 & *_ref*)

7.17.2.23 void vc::BasicVCComponent::Realize () [virtual]

Override this function to create all Orbiter objects required: Thrusters, animations, meshes.

Implements ISubsystem.

Reimplemented in vc::StandardSwitch.

7.17.2.24 bool vc::BasicVCComponent::RegisterComponent (SubsystemComponent * *pComponent*) [virtual]

Implements ISubsystem.

7.17.2.25 void vc::BasicVCComponent::RegisterVC () [virtual]

Reimplemented in vc::MDU.

7.17.2.26 bool vc::BasicVCComponent::DelComponent (SubsystemComponent * *pComponent*) [virtual]

Implements ISubsystem.

7.17.2.27 unsigned long vc::BasicVCComponent::CountComponents () const [virtual]

Implements ISubsystem.

7.17.2.28 void vc::BasicVCComponent::OnPreStep (double *fSimT*, double *fDeltaT*, double *fMJD*) [virtual]

Implements ISubsystem.

7.17.2.29 void vc::BasicVCComponent::OnPostStep (double *fSimT*, double *fDeltaT*, double *fMJD*) [virtual]

Implements ISubsystem.

7.17.2.30 void vc::BasicVCComponent::OnPropagate (double *fSimT*, double *fDeltaT*, double *fMJD*) [virtual]

Implements ISubsystem.

7.17.2.31 void vc::BasicVCComponent::OnPlaybackEvent (double *fSimT*, double *fEventT*, const char * *event_t*, const char * *event*) [virtual]

Implements ISubsystem.

7.17.2.32 bool vc::BasicVCComponent::OnVCRedrawEvent (int *id*, int *_event*, SURFHANDLE *surf*) [virtual]

7.17.2.33 bool vc::BasicVCComponent::OnParseLine (const char * *line*) [virtual]

Implements ISubsystem.

7.17.2.34 void vc::BasicVCComponent::OnSaveState (FILEHANDLE *scn*) const [virtual]

Implements ISubsystem.

Reimplemented in vc::BasicSwitch.

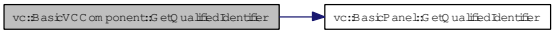
7.17.2.35 void vc::BasicVCComponent::OnDumpToLog () const [virtual]

Implements ISubsystem.

7.17.2.36 const string & vc::BasicVCComponent::GetQualifiedIdentifier () const [virtual]

Implements ISubsystem.

Here is the call graph for this function:



7.17.2.37 const string & vc::BasicVCComponent::GetIdentifier () const [virtual]

Get identification of this component

Returns:

The identification string, used for locating this component inside a panel

Implements ISubsystem.

7.17.2.38 bool vc::BasicVCComponent::OnFail (const string & *strFailureCode*) [virtual]

Implements ISubsystem.

7.17.3 Member Data Documentation

7.17.3.1 Atlantis* vc::BasicVCComponent::sts [private]

7.17.3.2 BasicPanel* vc::BasicVCComponent::pParent [private]

7.17.3.3 string vc::BasicVCComponent::ident [private]

7.17.3.4 VECTOR3 vc::BasicVCComponent::reference [private]

7.17.3.5 VECTOR3 vc::BasicVCComponent::dir [private]

7.17.3.6 double vc::BasicVCComponent::fInitialAnimState [private]

7.17.3.7 VCMOUSEEVENTPOINT vc::BasicVCComponent::p_a [private]

7.17.3.8 VCMOUSEEVENTPOINT vc::BasicVCComponent::p_b [private]

7.17.3.9 VCMOUSEEVENTPOINT vc::BasicVCComponent::p_c [private]

7.17.3.10 VCMOUSEEVENTPOINT vc::BasicVCComponent::p_d [private]

7.17.3.11 float vc::BasicVCComponent::cache_lastx [mutable, private]

7.17.3.12 float vc::BasicVCComponent::cache_lasty [mutable, private]

7.17.3.13 float vc::BasicVCComponent::cache_mx [mutable, private]

7.17.3.14 float vc::BasicVCComponent::cache_my [mutable, private]

7.17.3.15 set<UINT> vc::BasicVCComponent::availableForRendering [private]

7.17.3.16 bool vc::BasicVCComponent::bHasReference [protected]

7.17.3.17 bool vc::BasicVCComponent::bHasDirection [protected]

7.17.3.18 bool vc::BasicVCComponent::bHasAnimations [protected]

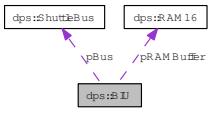
The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicVCComponent.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicVCComponent.cpp

7.18 dps::BIU Class Reference

`#include <BIU.h>`

Collaboration diagram for dps::BIU:



Public Member Functions

- BIU (RAM16 *pRAMBuffer, bool bIsMaster=false)
- virtual ~BIU ()
- void ConnectToBus (ShuttleBus *pBus)
- void MakeBusMaster ()
- void Listen ()
- virtual void OnNewMessage ()

Private Attributes

- RAM16 * pRAMBuffer
- ShuttleBus * pBus
- bool bIsBusMaster

7.18.1 Constructor & Destructor Documentation

7.18.1.1 dps::BIU::BIU (RAM16 ∗ *pRAMBuffer*, bool *bIsMaster* = `false`)

7.18.1.2 virtual dps::BIU::∼BIU () `[virtual]`

7.18.2 Member Function Documentation

7.18.2.1 void dps::BIU::ConnectToBus (ShuttleBus ∗ *pBus*)

7.18.2.2 void dps::BIU::MakeBusMaster ()

7.18.2.3 void dps::BIU::Listen ()

7.18.2.4 virtual void dps::BIU::OnNewMessage () `[virtual]`

7.18.3 Member Data Documentation

7.18.3.1 RAM16∗ dps::BIU::pRAMBuffer `[private]`

7.18.3.2 ShuttleBus∗ dps::BIU::pBus `[private]`

7.18.3.3 bool dps::BIU::bIsBusMaster `[private]`

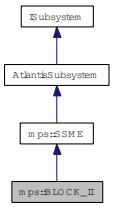
The documentation for this class was generated from the following file:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/BIU.h

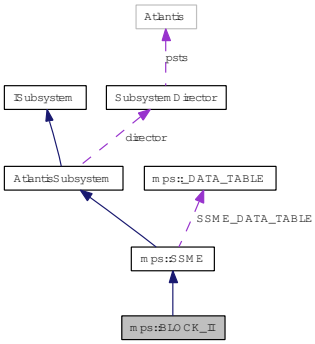
7.19 mps::BLOCK_II Class Reference

`#include <BLOCK_II.h>`

Inheritance diagram for mps::BLOCK_II:



Collaboration diagram for mps::BLOCK_II:



Public Member Functions

- BLOCK_II (SubsystemDirector *, const string &, int)
- ~BLOCK_II (void)
- void OnSaveState (FILEHANDLE) const
- bool OnParseLine (const char *)

Private Member Functions

- double dcPC_ESC (double)
- double dcMOV_ESC (double)
- double dcMFV_ESC (double)
- double dcCCV_ESC (double)
- double dcFPOV_ESC (double)
- double dcOPOV_ESC (double)
- double dcPC_CO (double)
- double dcMOV_CO (double)

- double dcMFV_CO (double)
- double dcCCV_CO (double)
- double dcFPOV_CO (double, double)
- double dcOPOV_CO (double, double)
- double dcPC_MS (double)
- double dcCCV_MS (double)
- double dcFPOV_MS (double)
- double dcOPOV_MS (double)

7.19.1 Constructor & Destructor Documentation

7.19.1.1 mps::BLOCK_II::BLOCK_II (SubsystemDirector * *_director*, const string & *_ident*, int *nID*)

7.19.1.2 mps::BLOCK_II::~BLOCK_II (void)

7.19.2 Member Function Documentation

7.19.2.1 double mps::BLOCK_II::dcPC_ESC (double *tme*) [private, virtual]

Implements mps::SSME.

7.19.2.2 double mps::BLOCK_II::dcMOV_ESC (double *tme*) [private, virtual]

Implements mps::SSME.

7.19.2.3 double mps::BLOCK_II::dcMFV_ESC (double *tme*) [private, virtual]

Implements mps::SSME.

7.19.2.4 double mps::BLOCK_II::dcCCV_ESC (double *tme*) [private, virtual]

Implements mps::SSME.

7.19.2.5 double mps::BLOCK_II::dcFPOV_ESC (double *tme*) [private, virtual]

Implements mps::SSME.

7.19.2.6 double mps::BLOCK_II::dcOPOV_ESC (double *tme*) [private, virtual]

Implements mps::SSME.

7.19.2.7 double mps::BLOCK_II::dcPC_CO (double *tme*) [private, virtual]

Implements mps::SSME.

7.19.2.8 double mps::BLOCK_II::dcMOV_CO (double *tme*) [private, virtual]

Implements mps::SSME.

7.19.2.9 double mps::BLOCK_II::dcMFV_CO (double *tme*) [private, virtual]

Implements mps::SSME.

7.19.2.10 double mps::BLOCK_II::dcCCV_CO (double *tme*) [private, virtual]

Implements mps::SSME.

7.19.2.11 double mps::BLOCK_II::dcFPOV_CO (double *tme*, double *POS1*) [private, virtual]

Implements mps::SSME.

7.19.2.12 double mps::BLOCK_II::dcOPOV_CO (double *tme*, double *POS1*) [private, virtual]

Implements mps::SSME.

7.19.2.13 double mps::BLOCK_II::dcPC_MS (double *dtime*) [private, virtual]

Implements mps::SSME.

7.19.2.14 double mps::BLOCK_II::dcCCV_MS (double *pc*) [private, virtual]

Implements mps::SSME.

7.19.2.15 double mps::BLOCK_II::dcFPOV_MS (double *pc*) [private, virtual]

Implements mps::SSME.

7.19.2.16 double mps::BLOCK_II::dcOPOV_MS (double *pc*) [private, virtual]

Implements mps::SSME.

7.19.2.17 void mps::BLOCK_II::OnSaveState (FILEHANDLE *scn*) const [virtual]

Save the state of the subsystem

Implements mps::SSME.

7.19.2.18 bool mps::BLOCK_II::OnParseLine (const char * *line*) [virtual]

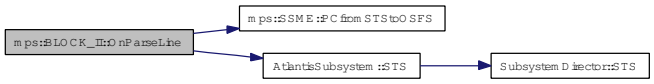
Parse a line of the scenario file.

Warning:

Should get replaced by a block oriented scenario file format, like NASSP, making this function deprecated.

Implements mps::SSME.

Here is the call graph for this function:



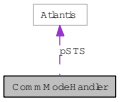
The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/BLOCK_II.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/BLOCK_II.cpp

7.20 CommModeHandler Class Reference

`#include <CommModeHandler.h>`

Collaboration diagram for CommModeHandler:



Public Member Functions

- CommModeHandler (Atlantis *_pSTS)
- void DefineAnnotations ()
- bool ConsumeBufferedKey (DWORD key, bool pressed, char *keymap)
- void PostStep (double fSimT, double fDeltaT)
- void OnReceiveFocus ()
- void OnLooseFocus ()
- bool IsInCommMode () const
- void EnterCommMode ()

Protected Member Functions

- void BlankReceiveAnnotation ()
- void BlankSendAnnotations ()
- void LeaveCommMode ()

Protected Attributes

- NOTEHANDLE nhLastSend
- NOTEHANDLE nhOptionPos
- NOTEHANDLE nhLastReceived
- bool bCommMode

Private Member Functions

- void ClearRecAnnotations ()
- void ClearSendAnnotations ()

Private Attributes

- Atlantis * pSTS
- string strDecompG2A
- string strDecompA2G
- string strCompA2G

- string strNegAnswer_Comp
- string strNegAnswer_Decomp
- string strPosAnswer_Comp
- string strPosAnswer_Decomp
- double fRecDisplayT0
- double fSendDisplayT0
- int iMenu

7.20.1 Constructor & Destructor Documentation

7.20.1.1 CommModeHandler::CommModeHandler (Atlantis * *_pSTS*)

7.20.2 Member Function Documentation

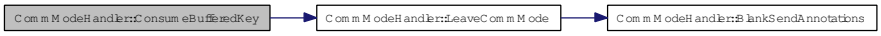
7.20.2.1 void CommModeHandler::ClearRecAnnotations () [private]

7.20.2.2 void CommModeHandler::ClearSendAnnotations () [private]

7.20.2.3 void CommModeHandler::DefineAnnotations ()

7.20.2.4 bool CommModeHandler::ConsumeBufferedKey (DWORD *key*, bool *pressed*, char * *keymap*)

Here is the call graph for this function:



7.20.2.5 void CommModeHandler::PostStep (double *fSimT*, double *fDeltaT*)

Here is the call graph for this function:

7.20.2.6 void CommModeHandler::OnReceiveFocus ()

7.20.2.7 void CommModeHandler::OnLooseFocus ()

Here is the call graph for this function:

7.20.2.8 bool CommModeHandler::IsInCommMode () const

7.20.2.9 void CommModeHandler::EnterCommMode ()

7.20.2.10 void CommModeHandler::BlankReceiveAnnotation () [protected]

7.20.2.11 void CommModeHandler::BlankSendAnnotations () [protected]

7.20.2.12 void CommModeHandler::LeaveCommMode () [protected]

Here is the call graph for this function:

7.20.3 Member Data Documentation

7.20.3.1 Atlantis* CommModeHandler::pSTS [private]

7.20.3.2 string CommModeHandler::strDecompG2A [private]

7.20.3.3 string CommModeHandler::strDecompA2G [private]

7.20.3.4 string CommModeHandler::strCompA2G [private]

7.20.3.5 string CommModeHandler::strNegAnswer_Comp [private]

7.20.3.6 string CommModeHandler::strNegAnswer_Decomp [private]

7.20.3.7 string CommModeHandler::strPosAnswer_Comp [private]

7.20.3.8 string CommModeHandler::strPosAnswer_Decomp [private]

7.20.3.9 double CommModeHandler::fRecDisplayT0 [private]

7.20.3.10 double CommModeHandler::fSendDisplayT0 [private]

7.20.3.11 int CommModeHandler::iMenu [private]

7.20.3.12 NOTEHANDLE CommModeHandler::nhLastSend [protected]

7.20.3.13 NOTEHANDLE CommModeHandler::nhOptionPos [protected]

7.20.3.14 NOTEHANDLE CommModeHandler::nhLastReceived [protected]

7.20.3.15 bool CommModeHandler::bCommMode [protected]

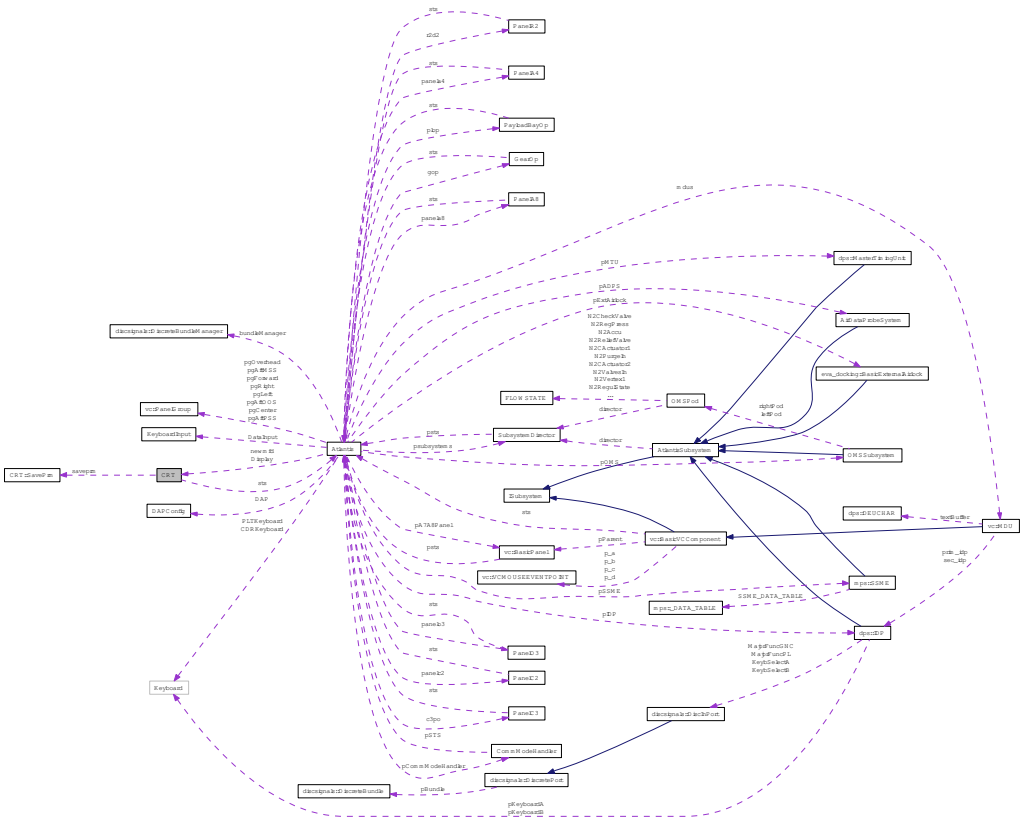
The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/CommModeHandler.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/CommModeHandler.cpp

7.21 CRT Class Reference

`#include <CRT.h>`

Collaboration diagram for CRT:



Public Member Functions

- CRT (DWORD w, DWORD h, VESSEL *vessel)
- ~CRT ()
- char * ButtonLabel (int bt)
- int ButtonMenu (const MFDBUTTONMENU **menu) const
- bool ConsumeKeyBuffered (DWORD key)
- bool ConsumeButton (int bt, int event)
- void Update (HDC hDC)
- bool Input (int change, char *Name)
- void Data (int id)
- void WriteStatus (FILEHANDLE scn) const
- void ReadStatus (FILEHANDLE scn)
- void StoreStatus () const
- void RecallStatus ()

Static Public Member Functions

- static int MsgProc (UINT msg, UINT mfd, WPARAM wparam, LPARAM lparam)

Public Attributes

- Atlantis * sts

Protected Member Functions

- bool CRTLine (HDC hdc, short sX1, short sY1, short sX2, short sY2)
- bool CRTTextOut (HDC hdc, short sCol, short sLine, char *text)
- void GNCSYSSUMM1 (HDC hdc)
- void DrawCommonHeader (HDC hdc)
- void PASSTRAJ (HDC hdc)
- void RecallCriticalFormat (int iFormatCode)

Protected Attributes

- char cDispTitle [20]
- HFONT hCRTFont

Private Member Functions

- void SetDisplayTitle (const char *pszTitle)
- VECTOR3 ToRad (const VECTOR3 &Input)
- VECTOR3 ToDeg (const VECTOR3 &Input)
- bool Eq (const double d1, const double d2)
- void OMSMPS (HDC hDC)
- void SPI (HDC hDC)
- void APUHYD (HDC hDC)
- void UNIVPTG (HDC hDC)
- void DAP_CONFIG (HDC hDC)
- void MNVR (HDC hDC)
- void GPCMEMORY (HDC hdc)
- void PDRS (HDC hDC)
- void DisplayScratchPad (HDC hDC)
- void DrawDelta (HDC hDC, int TopX, int TopY, int LBottomX, int RBottomX, int BottomY)
- void EnableThrusters (const int Thrusters[ ], int nThrusters)
- void DisableThrusters (const int Thrusters[ ], int nThrusters)
- void OMSGimbal (VECTOR3 Targets)
- void LoadManeuver ()
- void UpdateDAP ()

Private Attributes

- unsigned short usPageNumber
- unsigned short usDisp
- unsigned short usGPCDriver
- unsigned short usMDU
- VESSEL * vessel
- VECTOR3 GVesselPos
- VECTOR3 GVesselVel
- VESSELSTATUS Status
- int width
- int height
- int ops
- int item
- int spec
- int mode
- int display
- int data
- int id
- int MET [4]
- int Launch_time [4]
- int MET_Add [4]
- double dT
- bool landed
- int MPS_Manifold_Press [2]
- int MPS_He_Reg_Press [4]
- int MPS_He_Tank_Press [4]
- int Hydraulic_Press [3]
- double APU_Fuel_Last [3]
- double APU_Fuel_Rate
- double Simtime_last
- double Simtime
- VECTOR3 AngularVelocity
- int edit
- VECTOR3 PMI
- bool bTIMER
- int TGO [2]
- int RMS_SEL
- bool UpdateStatus

Static Private Attributes

- static struct CRT::SavePrm saveprm

Friends

- class Atlantis
- class Keyboard

Classes

- struct SavePrm

7.21.1 Constructor & Destructor Documentation

7.21.1.1 CRT::CRT (DWORD *w*, DWORD *h*, VESSEL * *vessel*)

Here is the call graph for this function:

7.21.1.2 CRT::~CRT ()

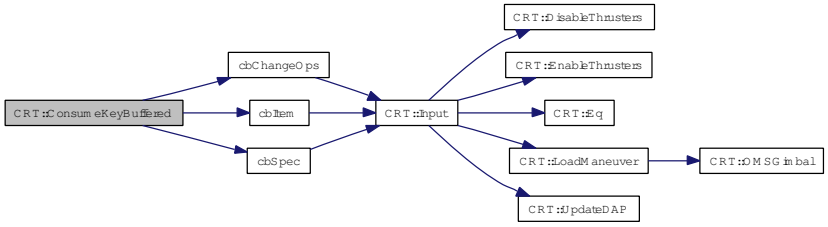
7.21.2 Member Function Documentation

7.21.2.1 char * CRT::ButtonLabel (int *bt*)

7.21.2.2 int CRT::ButtonMenu (const MFDBUTTONMENU ** *menu*) const

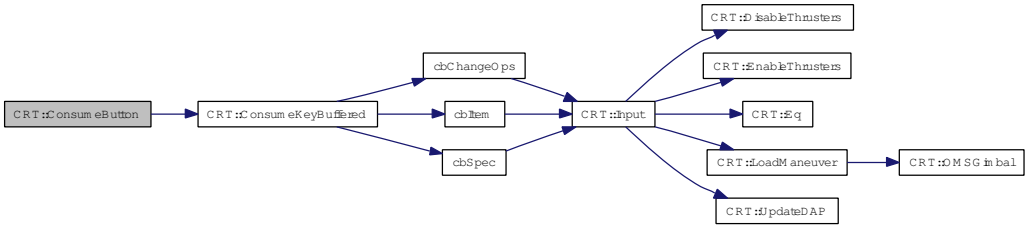
7.21.2.3 bool CRT::ConsumeKeyBuffered (DWORD *key*)

Here is the call graph for this function:



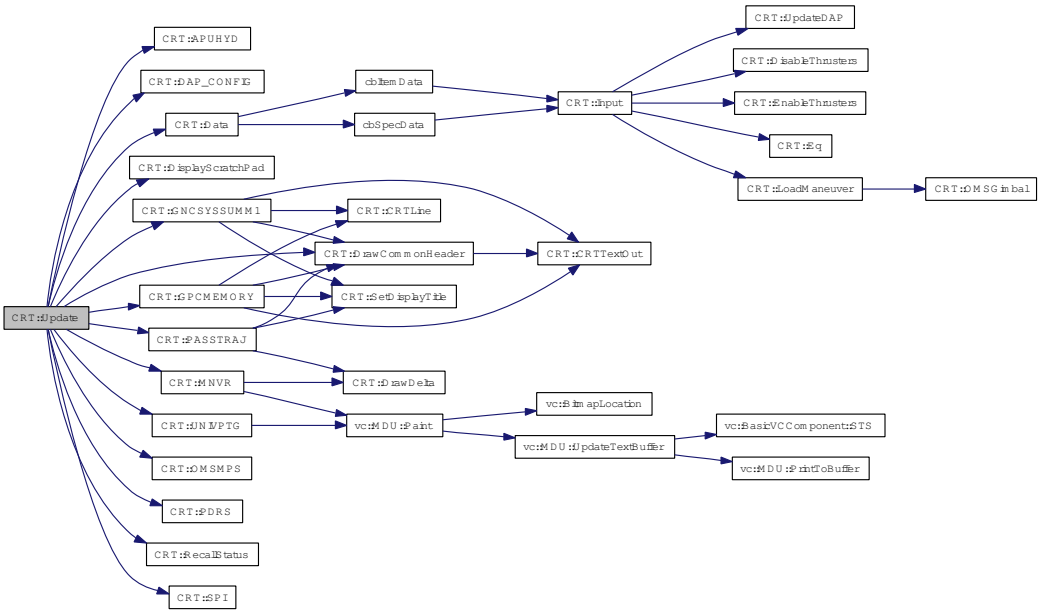
7.21.2.4 bool CRT::ConsumeButton (int *bt*, int *event*)

Here is the call graph for this function:



7.21.2.5 void CRT::Update (HDC *hDC*)

Here is the call graph for this function:

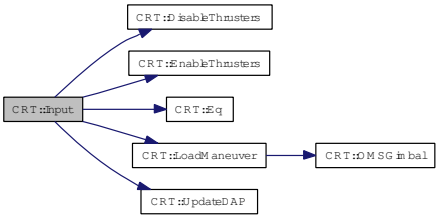


7.21.2.6 int CRT::MsgProc (UINT *msg*, UINT *mfd*, WPARAM *wparam*, LPARAM *lparam*) [static]

Here is the call graph for this function:

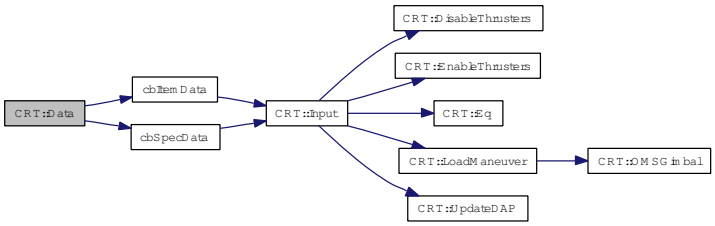
7.21.2.7 bool CRT::Input (int *change*, char * *Name*)

Here is the call graph for this function:



7.21.2.8 void CRT::Data (int *id*)

Here is the call graph for this function:



7.21.2.9 void CRT::WriteStatus (FILEHANDLE *scn*) const

7.21.2.10 void CRT::ReadStatus (FILEHANDLE *scn*)

7.21.2.11 void CRT::StoreStatus () const

7.21.2.12 void CRT::RecallStatus ()

7.21.2.13 void CRT::SetDisplayTitle (const char * *pszTitle*) [private]

7.21.2.14 VECTOR3 CRT::ToRad (const VECTOR3 & *Input*) [inline, private]

7.21.2.15 VECTOR3 CRT::ToDeg (const VECTOR3 & *Input*) [inline, private]

7.21.2.16 bool CRT::Eq (const double *d1*, const double *d2*) [inline, private]

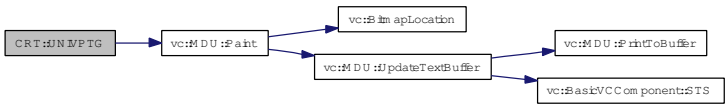
7.21.2.17 void CRT::OMSMPS (HDC *hDC*) [private]

7.21.2.18 void CRT::SPI (HDC *hDC*) [private]

7.21.2.19 void CRT::APUHYD (HDC *hDC*) [private]

7.21.2.20 void CRT::UNIVPTG (HDC *hDC*) [private]

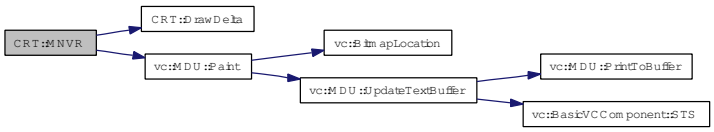
Here is the call graph for this function:



7.21.2.21 void CRT::DAP_CONFIG (HDC *hDC*) [private]

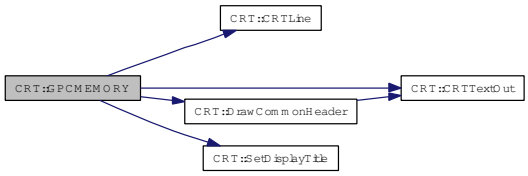
7.21.2.22 void CRT::MNVR (HDC *hDC*) [private]

Here is the call graph for this function:



7.21.2.23 void CRT::GPCMEMORY (HDC *hdc*) [private]

Here is the call graph for this function:



7.21.2.24 void CRT::PDRS (HDC *hDC*) [private]

7.21.2.25 void CRT::DisplayScratchPad (HDC *hDC*) [private]

7.21.2.26 void CRT::DrawDelta (HDC *hDC*, int *TopX*, int *TopY*, int *LBottomX*, int *RBottomX*, int *BottomY*) [private]

7.21.2.27 void CRT::EnableThrusters (const int *Thrusters*[ ], int *nThrusters*) [private]

7.21.2.28 void CRT::DisableThrusters (const int *Thrusters*[ ], int *nThrusters*) [private]

7.21.2.29 void CRT::OMSGimbal (VECTOR3 *Targets*) [private]

7.21.2.30 void CRT::LoadManeuver () [private]

Here is the call graph for this function:

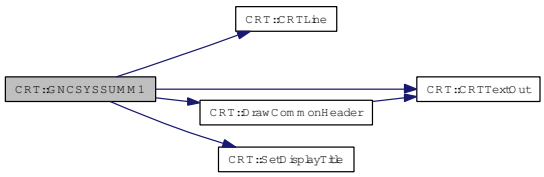
7.21.2.31 void CRT::UpdateDAP () [private]

7.21.2.32 bool CRT::CRTLine (HDC *hdc*, short *sX1*, short *sY1*, short *sX2*, short *sY2*) [protected]

7.21.2.33 bool CRT::CRTTextOut (HDC *hdc*, short *sCol*, short *sLine*, char * *text*) [protected]

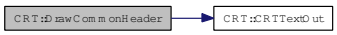
7.21.2.34 void CRT::GNCSYSSUMM1 (HDC *hdc*) [protected]

Here is the call graph for this function:



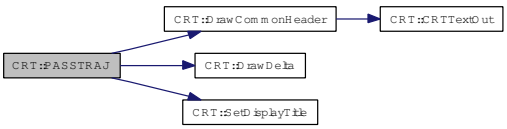
7.21.2.35 void CRT::DrawCommonHeader (HDC *hdc*) [protected]

Here is the call graph for this function:



7.21.2.36 void CRT::PASSTRAJ (HDC *hdc*) [protected]

Here is the call graph for this function:



7.21.2.37 void CRT::RecallCriticalFormat (int *iFormatCode*) [protected]

7.21.3 Friends And Related Function Documentation

7.21.3.1 friend class Atlantis [friend]

7.21.3.2 friend class Keyboard [friend]

7.21.4 Member Data Documentation

7.21.4.1 Atlantis* CRT::sts

7.21.4.2 unsigned short CRT::usPageNumber [private]

7.21.4.3 unsigned short CRT::usDisp [private]

7.21.4.4 unsigned short CRT::usGPCDriver [private]

7.21.4.5 unsigned short CRT::usMDU [private]

7.21.4.6 VESSEL* CRT::vessel [private]

7.21.4.7 VECTOR3 CRT::GVesselPos [private]

7.21.4.8 VECTOR3 CRT::GVesselVel [private]

7.21.4.9 VESSELSTATUS CRT::Status [private]

7.21.4.10 int CRT::width [private]

7.21.4.11 int CRT::height [private]

7.21.4.12 int CRT::ops [private]

7.21.4.13 int CRT::item [private]

7.21.4.14 int CRT::spec [private]

7.21.4.15 int CRT::mode [private]

7.21.4.16 int CRT::display [private]

7.21.4.17 int CRT::data [private]

7.21.4.18 int CRT::id [private]

7.21.4.19 int CRT::MET[4] [private]

7.21.4.20 int CRT::Launch_time[4] [private]

7.21.4.21 int CRT::MET_Add[4] [private]

7.21.4.22 double CRT::dT [private]

7.21.4.23 bool CRT::landed [private]

7.21.4.24 int CRT::MPS_Manifold_Press[2] [private]

7.21.4.25 int CRT::MPS_He_Reg_Press[4] [private]

7.21.4.26 int CRT::MPS_He_Tank_Press[4] [private]

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/CRT.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/CRT.cpp

7.22 CRT::SavePrm Struct Reference

Public Attributes

- int spec [11]
- int mode [11]
- int display [11]
- bool bValid [11]

7.22.1 Member Data Documentation

7.22.1.1 int CRT::SavePrm::spec[11]

7.22.1.2 int CRT::SavePrm::mode[11]

7.22.1.3 int CRT::SavePrm::display[11]

7.22.1.4 bool CRT::SavePrm::bValid[11]

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

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/CRT.h

7.23 DAPConfig Struct Reference

```
#include <Atlantis.h>
```

Public Attributes

- double PRI_ROT_RATE
- double PRI_ATT_DB
- double PRI_RATE_DB
- double PRI_ROT_PLS
- double PRI_COMP
- double PRI_TRAN_PLS
- int PRI_P_OPTION
- int PRI_Y_OPTION
- double ALT_RATE_DB
- double ALT_ON_TIME
- double ALT_DELAY
- int ALT_JET_OPT
- int ALT_JETS
- double VERN_ROT_RATE
- double VERN_ATT_DB
- double VERN_RATE_DB
- double VERN_ROT_PLS
- double VERN_COMP
- int VERN_CNTL_ACC

7.23.1 Member Data Documentation

7.23.1.1 double DAPConfig::PRI_ROT_RATE

7.23.1.2 double DAPConfig::PRI_ATT_DB

7.23.1.3 double DAPConfig::PRI_RATE_DB

7.23.1.4 double DAPConfig::PRI_ROT_PLS

7.23.1.5 double DAPConfig::PRI_COMP

7.23.1.6 double DAPConfig::PRI_TRAN_PLS

7.23.1.7 int DAPConfig::PRI_P_OPTION

7.23.1.8 int DAPConfig::PRI_Y_OPTION

7.23.1.9 double DAPConfig::ALT_RATE_DB

7.23.1.10 double DAPConfig::ALT_ON_TIME

7.23.1.11 double DAPConfig::ALT_DELAY

7.23.1.12 int DAPConfig::ALT_JET_OPT

7.23.1.13 int DAPConfig::ALT_JETS

7.23.1.14 double DAPConfig::VERN_ROT_RATE

7.23.1.15 double DAPConfig::VERN_ATT_DB

7.23.1.16 double DAPConfig::VERN_RATE_DB

7.23.1.17 double DAPConfig::VERN_ROT_PLS

7.23.1.18 double DAPConfig::VERN_COMP

7.23.1.19 int DAPConfig::VERN_CNTL_ACC

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

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Atlantis.h

7.24 dps::DEUCHAR Struct Reference

```
#include <dps_defs.h>
```

Public Attributes

- char cSymbol
- char cAttr

7.24.1 Member Data Documentation

7.24.1.1 char dps::DEUCHAR::cSymbol

7.24.1.2 char dps::DEUCHAR::cAttr

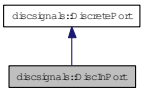
The documentation for this struct was generated from the following file:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/dps_defs.h

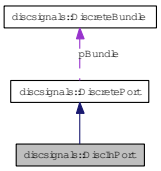
7.25 discsignals::DiscInPort Class Reference

```
#include <DiscInPort.h>
```

Inheritance diagram for discsignals::DiscInPort:



Collaboration diagram for discsignals::DiscInPort:



Public Member Functions

- DiscInPort ()
- virtual ~DiscInPort ()
- bool IsSet () const
- bool IsSet (float fThreshold) const
- float GetVoltage () const

7.25.1 Constructor & Destructor Documentation

7.25.1.1 discsignals::DiscInPort::DiscInPort ()

7.25.1.2 discsignals::DiscInPort::~DiscInPort () [virtual]

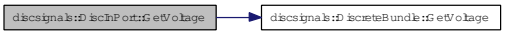
7.25.2 Member Function Documentation

7.25.2.1 bool discsignals::DiscInPort::IsSet () const

7.25.2.2 bool discsignals::DiscInPort::IsSet (float *fThreshold*) const

7.25.2.3 float discsignals::DiscInPort::GetVoltage () const

Here is the call graph for this function:



The documentation for this class was generated from the following files:

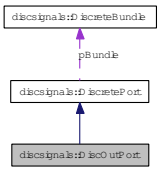
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscInPort.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscInPort.cpp

7.26 discsignals::DiscOutPort Class Reference

`#include <DiscOutPort.h>`

Inheritance diagram for discsignals::DiscOutPort:

Collaboration diagram for discsignals::DiscOutPort:



Public Member Functions

- void SetLine ()
- void SetLine (float fVoltage)
- void ResetLine ()
- DiscOutPort ()
- virtual ~DiscOutPort ()

7.26.1 Constructor & Destructor Documentation

7.26.1.1 discsignals::DiscOutPort::DiscOutPort ()

7.26.1.2 discsignals::DiscOutPort::~DiscOutPort () `[virtual]`

7.26.2 Member Function Documentation

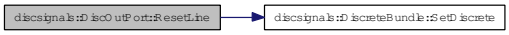
7.26.2.1 void discsignals::DiscOutPort::SetLine ()

Here is the call graph for this function:

7.26.2.2 void discsignals::DiscOutPort::SetLine (float *fVoltage*)

7.26.2.3 void discsignals::DiscOutPort::ResetLine ()

Here is the call graph for this function:



The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscOutPort.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscOutPort.cpp

7.27 discsignals::DiscreteBundle Class Reference

```
#include <DiscreteBundle.h>
```

Public Member Functions

- DiscreteBundle (const string &_ident, int nLines=16)
- virtual ~DiscreteBundle ()
- void SetLogMode (int iMode)
- float GetVoltage (int iIndex)
- const string & GetIdentity () const
- void SetDiscrete (int iIndex, float fVoltage)

Private Attributes

- string ident
- short snLines
- float fState
- float fLines [16]
- bool bBrokenLine [16]

7.27.1 Detailed Description

class for handling multiple discrete data lines or analog inputs State for each discrete can be -28V to +28V One bundle can have up to 16 lines. one line has a beginning and a end point. 0 = ground (GND)

7.27.2 Constructor & Destructor Documentation

7.27.2.1 discsignals::DiscreteBundle::DiscreteBundle (const string & *_ident*, int *nLines* = 16)

7.27.2.2 discsignals::DiscreteBundle::~DiscreteBundle () [virtual]

7.27.3 Member Function Documentation

7.27.3.1 void discsignals::DiscreteBundle::SetLogMode (int *iMode*)

Set the mode how the Discrete bundle class handles logging information

7.27.3.2 float discsignals::DiscreteBundle::GetVoltage (int *iIndex*)

7.27.3.3 const string & discsignals::DiscreteBundle::GetIdentity () const

7.27.3.4 void discsignals::DiscreteBundle::SetDiscrete (int *iIndex*, float *fVoltage*)

7.27.4 Member Data Documentation

7.27.4.1 string discsignals::DiscreteBundle::ident [private]

7.27.4.2 short discsignals::DiscreteBundle::snLines [private]

7.27.4.3 float discsignals::DiscreteBundle::fState [private]

7.27.4.4 float discsignals::DiscreteBundle::fLines[16] [private]

7.27.4.5 bool discsignals::DiscreteBundle::bBrokenLine[16] [private]

Indicate that a line is broken and does always return 0V.

The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscreteBundle.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscreteBundle.cpp

7.28 discsignals::DiscreteBundleManager Class Reference

```
#include <DiscreteBundleManager.h>
```

Public Member Functions

- DiscreteBundleManager (void)
- virtual ~DiscreteBundleManager (void)
- unsigned long CountBundles () const
- bool AddBundle (DiscreteBundle *bBundle)
- bool RemoveBundle (DiscreteBundle *pBundle)
- bool ExistsBundle (const string &_ident) const
- DiscreteBundle * FindBundle (const string &_ident) const
- DiscreteBundle * GetBundleByIndex (unsigned long ulIndex) const
- DiscreteBundle * CreateBundle (const string &_ident, unsigned short usLines)
- void DumpToLog () const

Private Attributes

- vector< DiscreteBundle * > bundles

7.28.1 Constructor & Destructor Documentation

7.28.1.1 discsignals::DiscreteBundleManager::DiscreteBundleManager (void)

7.28.1.2 discsignals::DiscreteBundleManager::~DiscreteBundleManager (void) [virtual]

7.28.2 Member Function Documentation

7.28.2.1 unsigned long discsignals::DiscreteBundleManager::CountBundles () const

7.28.2.2 bool discsignals::DiscreteBundleManager::AddBundle (discsignals::DiscreteBundle * *pBundle*)

7.28.2.3 bool discsignals::DiscreteBundleManager::RemoveBundle (DiscreteBundle * *pBundle*)

7.28.2.4 bool discsignals::DiscreteBundleManager::ExistsBundle (const string & *_ident*) const

7.28.2.5 DiscreteBundle* discsignals::DiscreteBundleManager::FindBundle (const string & *_ident*) const

7.28.2.6 DiscreteBundle* discsignals::DiscreteBundleManager::GetBundleByIndex (unsigned long *ulIndex*) const

7.28.2.7 DiscreteBundle* discsignals::DiscreteBundleManager::CreateBundle (const string & *_ident*, unsigned short *usLines*)

7.28.2.8 void discsignals::DiscreteBundleManager::DumpToLog () const

7.28.3 Member Data Documentation

7.28.3.1 vector<DiscreteBundle*> discsignals::DiscreteBundleManager::bundles [private]

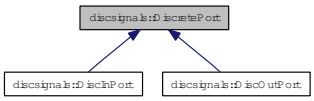
The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscreteBundleManager.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscreteBundleManager.cpp

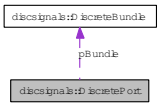
7.29 discsignals::DiscretePort Class Reference

```
#include <DiscretePort.h>
```

Inheritance diagram for discsignals::DiscretePort:



Collaboration diagram for discsignals::DiscretePort:



Public Member Functions

- bool Connect (DiscreteBundle *pBundle, int iLine)
- bool Disconnect ()
- DiscretePort ()
- DiscretePort (DiscreteBundle *pBundle, int iLine)
- virtual ~DiscretePort ()

Protected Attributes

- DiscreteBundle * pBundle
- unsigned short usLine

7.29.1 Constructor & Destructor Documentation

7.29.1.1 discsignals::DiscretePort::DiscretePort ()

7.29.1.2 discsignals::DiscretePort::DiscretePort (DiscreteBundle * *pBundle*, int *iLine*)

Here is the call graph for this function:

7.29.1.3 discsignals::DiscretePort::~DiscretePort () [virtual]

7.29.2 Member Function Documentation

7.29.2.1 bool discsignals::DiscretePort::Connect (DiscreteBundle * *pBundle*, int *iLine*)

7.29.2.2 bool discsignals::DiscretePort::Disconnect ()

7.29.3 Member Data Documentation

7.29.3.1 DiscreteBundle* discsignals::DiscretePort::pBundle [protected]

7.29.3.2 unsigned short discsignals::DiscretePort::usLine [protected]

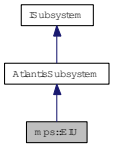
The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscretePort.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscretePort.cpp

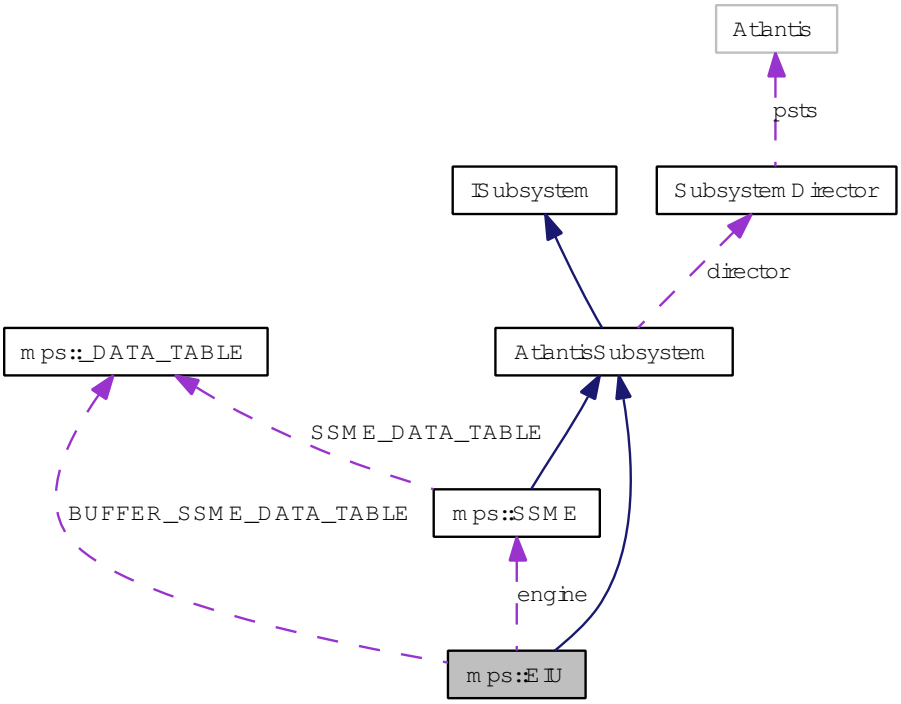
7.30 mps::EIU Class Reference

`#include <EIU.h>`

Inheritance diagram for mps::EIU:



Collaboration diagram for mps::EIU:



Public Member Functions

- EIU (SubsystemDirector *, const string &, int, SSME *)
- ~EIU (void)
- int RequestDATA (DATA_TABLE *)
- int RequestCMD (int, double)
- void OnPostStep (double, double, double)

Private Attributes

- int ID
- SSME * engine
- DATA_TABLE * BUFFER_SSME_DATA_TABLE

7.30.1 Constructor & Destructor Documentation

7.30.1.1 mps::EIU::EIU (SubsystemDirector * *_director*, const string & *_ident*, int *nID*, SSME * *ptr*)

Here is the call graph for this function:

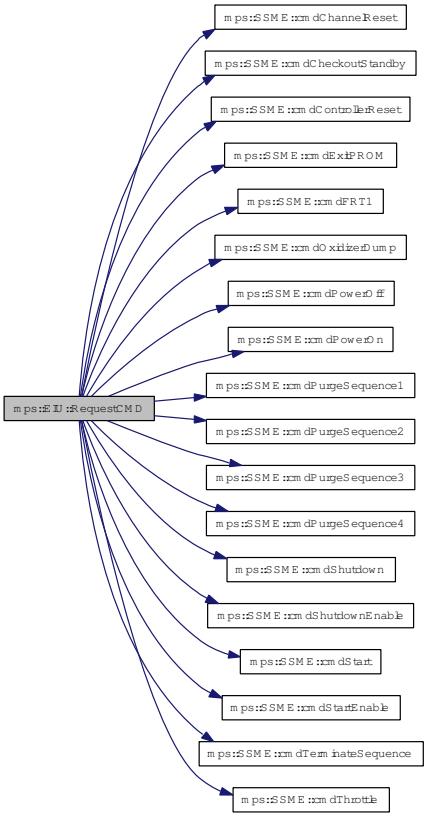
7.30.1.2 mps::EIU::~EIU (void)

7.30.2 Member Function Documentation

7.30.2.1 int mps::EIU::RequestDATA (DATA_TABLE * *ptrDataTable*)

7.30.2.2 int mps::EIU::RequestCMD (int *cmd*, double *data*)

Here is the call graph for this function:



7.30.2.3 void mps::EIU::OnPostStep (double *fSimT*, double *fDeltaT*, double *fMJD*) [virtual]

Finish calculations after the propagation of the state vector

Parameters:

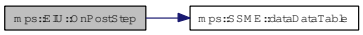
fSimT current simulation time

fDeltaT Simulation time step length

fMJD current modified julian date

Reimplemented from AtlantisSubsystem.

Here is the call graph for this function:



7.30.3 Member Data Documentation

7.30.3.1 int mps::EIU::ID [private]

7.30.3.2 SSME* mps::EIU::engine [private]

7.30.3.3 DATA_TABLE* mps::EIU::BUFFER_SSME_DATA_TABLE [private]

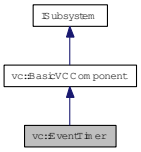
The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/EIU.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/EIU.cpp

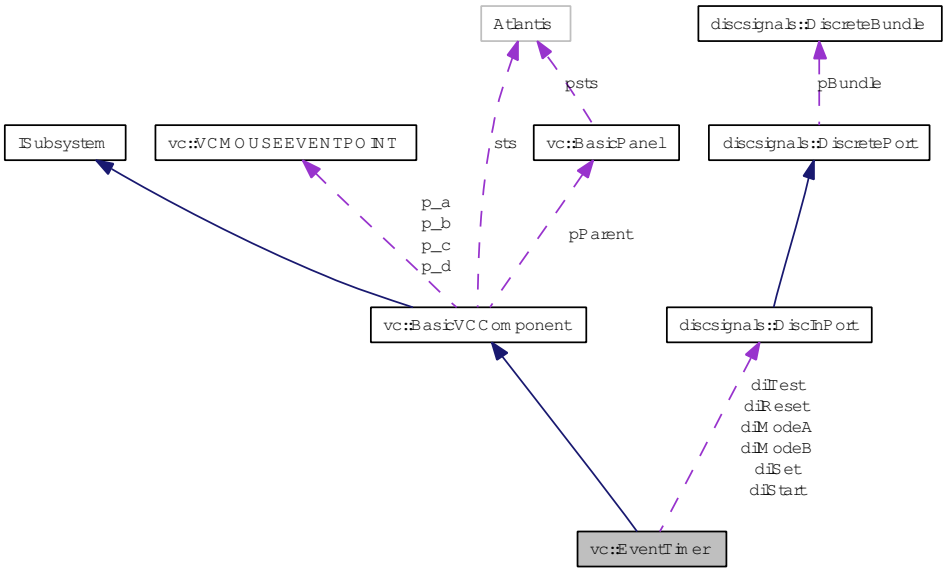
7.31 vc::EventTimer Class Reference

`#include <EventTimer.h>`

Inheritance diagram for vc::EventTimer:



Collaboration diagram for vc::EventTimer:



Public Member Functions

- EventTimer (Atlantis *_sts, const string &_ident)
- virtual ~EventTimer ()

Private Attributes

- short sDigit [4]
- bool fRunning
- double fEventTime
- unsigned short usMode

- DiscInPort dilModeA
- DiscInPort dilModeB
- DiscInPort dilStart
- DiscInPort dilTest
- DiscInPort dilSet
- DiscInPort dilReset

7.31.1 Constructor & Destructor Documentation

7.31.1.1 vc::EventTimer::EventTimer (Atlantis * *_sts*, const string & *_ident*)

7.31.1.2 virtual vc::EventTimer::~EventTimer () [virtual]

7.31.2 Member Data Documentation

7.31.2.1 short vc::EventTimer::sDigit[4] [private]

7.31.2.2 bool vc::EventTimer::fRunning [private]

7.31.2.3 double vc::EventTimer::fEventTime [private]

7.31.2.4 unsigned short vc::EventTimer::usMode [private]

7.31.2.5 DiscInPort vc::EventTimer::dilModeA [private]

7.31.2.6 DiscInPort vc::EventTimer::dilModeB [private]

7.31.2.7 DiscInPort vc::EventTimer::dilStart [private]

7.31.2.8 DiscInPort vc::EventTimer::dilTest [private]

7.31.2.9 DiscInPort vc::EventTimer::dilSet [private]

7.31.2.10 DiscInPort vc::EventTimer::dilReset [private]

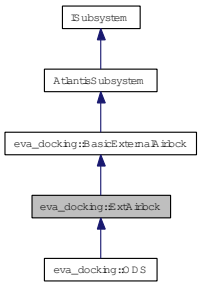
The documentation for this class was generated from the following file:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/EventTimer.h

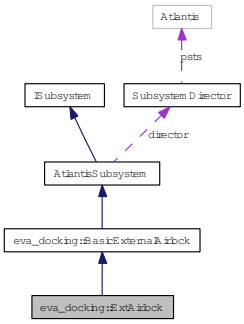
7.32 eva_docking::ExtAirlock Class Reference

`#include <ExtAirlock.h>`

Inheritance diagram for eva_docking::ExtAirlock:



Collaboration diagram for eva_docking::ExtAirlock:



Public Member Functions

- ExtAirlock (SubsystemDirector *pdirect, const string &_ident="External Airlock")
- virtual ~ExtAirlock ()
- virtual void AddMeshes (const VECTOR3 &ofs)

Protected Attributes

- double fHatchState
- double fExtALPress [2]

7.32.1 Constructor & Destructor Documentation

7.32.1.1 eva_docking::ExtAirlock::ExtAirlock (SubsystemDirector * *pdirect*, const string & *_ident* = "External Airlock")

7.32.1.2 eva_docking::ExtAirlock::~ExtAirlock () [virtual]

7.32.2 Member Function Documentation

7.32.2.1 void eva_docking::ExtAirlock::AddMeshes (const VECTOR3 & *ofs*) [virtual]

Add new meshes to the visual.

Deprecated

Don't use, instead, use Realize() to tell the Atlantis class which meshes it should use for special slots.

Reimplemented from eva_docking::BasicExternalAirlock.

Reimplemented in eva_docking::ODS.

7.32.3 Member Data Documentation

7.32.3.1 double eva_docking::ExtAirlock::fHatchState [protected]

7.32.3.2 double eva_docking::ExtAirlock::fExtALPress[2] [protected]

The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/eva_docking/ExtAirlock.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/eva_docking/ExtAirlock.cpp

7.33 FLOWSTATE Struct Reference

```
#include <Atlantis.h>
```

Public Attributes

- double P
- double T
- double mdot

7.33.1 Member Data Documentation

7.33.1.1 double FLOWSTATE::P

7.33.1.2 double FLOWSTATE::T

7.33.1.3 double FLOWSTATE::mdot

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

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Atlantis.h

7.34 GDIParams Struct Reference

```
#include <Atlantis.h>
```

Public Attributes

- HINSTANCE hDLL
- SURFHANDLE tkbk_label
- SURFHANDLE clock_digits
- SURFHANDLE digits_7seg
- HBITMAP deu_characters
- HDC DeuCharBitmapDC
- HFONT font [1]

7.34.1 Member Data Documentation

7.34.1.1 HINSTANCE GDIParams::hDLL

7.34.1.2 SURFHANDLE GDIParams::tkbk_label

7.34.1.3 SURFHANDLE GDIParams::clock_digits

7.34.1.4 SURFHANDLE GDIParams::digits_7seg

7.34.1.5 HBITMAP GDIParams::deu_characters

7.34.1.6 HDC GDIParams::DeuCharBitmapDC

7.34.1.7 HFONT GDIParams::font[1]

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

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Atlantis.h

7.35 GearOp Class Reference

`#include <GearOp.h>`

Collaboration diagram for GearOp:

Public Member Functions

- GearOp (Atlantis *_sts)
- void OpenDialog ()
- void DefineAnimations ()
- void DefineVCAnimations (UINT vcidx)
- void RegisterVC ()
- void UpdateVC ()
- void UpdateMesh ()
- bool VCMouseEvent (int id, int event, VECTOR3 &p)
- bool VCRedrawEvent (int id, int event, SURFHANDLE surf)
- void Step (double t, double dt)
- AnimState::Action GetGearAction () const
- void OperateLandingGear (AnimState::Action action)
- void RevertLandingGear ()
- void ArmGear ()
- void DamageGear ()
- void SetGearParameters (double state)
- bool GearArmed ()
- bool ParseScenarioLine (char *line)
- void SaveState (FILEHANDLE scn)

Private Types

- enum { SWC_CLOSE, SWC_OPEN }
- enum { SW_NOT_PUSH, SW_PUSH }

Private Member Functions

- bool VCDrawTalkback (SURFHANDLE surf, int idx, int label)

Private Attributes

- Atlantis * sts
- enum GearOp:: { ... } SwitchCover [2]
- enum GearOp:: { ... } SwitchPush [2]

- UINT anim_VC_F6 [2]
- int tkbk_state [3]
- AnimState::Action gear_status
- double gear_proc
- UINT anim_gear
- bool GearDamaged

Friends

- class Atlantis

7.35.1 Member Enumeration Documentation

7.35.1.1 anonymous enum [private]

Enumerator:

SWC_CLOSE

SWC_OPEN

7.35.1.2 anonymous enum [private]

Enumerator:

SW_NOT_PUSH

SW_PUSH

7.35.2 Constructor & Destructor Documentation

7.35.2.1 GearOp::GearOp (Atlantis * *_sts*)

7.35.3 Member Function Documentation

7.35.3.1 void GearOp::OpenDialog ()

7.35.3.2 void GearOp::DefineAnimations (void)

7.35.3.3 void GearOp::DefineVCAnimations (UINT *vcidx*)

7.35.3.4 void GearOp::RegisterVC ()

7.35.3.5 void GearOp::UpdateVC ()

7.35.3.6 void GearOp::UpdateMesh ()

7.35.3.7 bool GearOp::VCMouseEvent (int *id*, int *event*, VECTOR3 & *p*)

Here is the call graph for this function:

7.35.3.8 bool GearOp::VCRedrawEvent (int *id*, int *event*, SURFHANDLE *surf*)

Here is the call graph for this function:

7.35.3.9 void GearOp::Step (double *t*, double *dt*)

Here is the call graph for this function:

7.35.3.10 AnimState::Action GearOp::GetGearAction () const [inline]

7.35.3.11 void GearOp::OperateLandingGear (AnimState::Action *action*)

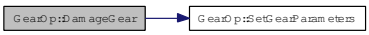
7.35.3.12 void GearOp::RevertLandingGear ()

Here is the call graph for this function:

7.35.3.13 void GearOp::ArmGear ()

7.35.3.14 void GearOp::DamageGear ()

Here is the call graph for this function:

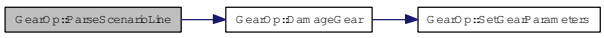


7.35.3.15 void GearOp::SetGearParameters (double *state*)

7.35.3.16 bool GearOp::GearArmed ()

7.35.3.17 bool GearOp::ParseScenarioLine (char * *line*)

Here is the call graph for this function:



7.35.3.18 void GearOp::SaveState (FILEHANDLE *scn*)

7.35.3.19 bool GearOp::VCDrawTalkback (SURFHANDLE *surf*, int *idx*, int *label*) [private]

7.35.4 Friends And Related Function Documentation

7.35.4.1 friend class Atlantis [friend]

7.35.5 Member Data Documentation

7.35.5.1 Atlantis* GearOp::sts [private]

7.35.5.2 enum { ... } GearOp::SwitchCover[2] [private]

7.35.5.3 enum { ... } GearOp::SwitchPush[2] [private]

7.35.5.4 UINT GearOp::anim_VC_F6[2] [private]

7.35.5.5 int GearOp::tkbk_state[3] [private]

7.35.5.6 AnimState::Action GearOp::gear_status [private]

7.35.5.7 double GearOp::gear_proc [private]

7.35.5.8 UINT GearOp::anim_gear [private]

7.35.5.9 bool GearOp::GearDamaged [private]

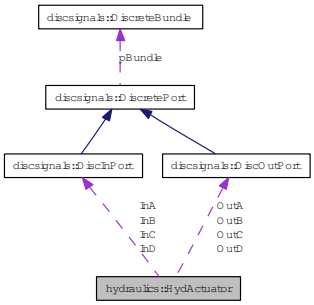
The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/GearOp.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/GearOp.cpp

7.36 hydraulics::HydActuator Class Reference

`#include <HydActuator.h>`

Collaboration diagram for hydraulics::HydActuator:



Public Member Functions

- HydActuator ()
- virtual ~HydActuator ()
- double GetActPos () const

Public Attributes

- DiscInPort InA
- DiscInPort InB
- DiscInPort InC
- DiscInPort InD
- DiscOutPort OutA
- DiscOutPort OutB
- DiscOutPort OutC
- DiscOutPort OutD

7.36.1 Constructor & Destructor Documentation

7.36.1.1 hydraulics::HydActuator::HydActuator ()

7.36.1.2 virtual hydraulics::HydActuator::~HydActuator () [virtual]

7.36.2 Member Function Documentation

7.36.2.1 double hydraulics::HydActuator::GetActPos () const

7.36.3 Member Data Documentation

7.36.3.1 DiscInPort hydraulics::HydActuator::InA

Command input port A

7.36.3.2 DiscInPort hydraulics::HydActuator::InB

7.36.3.3 DiscInPort hydraulics::HydActuator::InC

7.36.3.4 DiscInPort hydraulics::HydActuator::InD

7.36.3.5 DiscOutPort hydraulics::HydActuator::OutA

7.36.3.6 DiscOutPort hydraulics::HydActuator::OutB

7.36.3.7 DiscOutPort hydraulics::HydActuator::OutC

7.36.3.8 DiscOutPort hydraulics::HydActuator::OutD

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

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/hydraulics/HydActuator.h

7.37 eps::IACElement Class Reference

```
#include <eps_defs.h>
```

Public Member Functions

- virtual double GetVoltage (ACPHASE phase) const =0
- virtual double GetAmps (ACPHASE phase) const =0
- virtual bool DrawPower (ACPHASE phase, double fPower) const =0

7.37.1 Member Function Documentation

7.37.1.1 virtual double eps::IACElement::GetVoltage (ACPHASE *phase*) const [pure virtual]

electrical potential at the node

7.37.1.2 virtual double eps::IACElement::GetAmps (ACPHASE *phase*) const [pure virtual]

either amps produced by a power source or amps flowing through a node

7.37.1.3 virtual bool eps::IACElement::DrawPower (ACPHASE *phase*, double *fPower*) const [pure virtual]

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

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/eps/eps_defs.h

7.38 eps::IDCElement Class Reference

```
#include <eps_defs.h>
```

Public Member Functions

- virtual double GetVoltage () const =0
- virtual double GetAmps () const =0
- virtual double GetPower () const =0
- virtual bool DrawPower (double fPower) const =0

7.38.1 Member Function Documentation

7.38.1.1 virtual double eps::IDCElement::GetVoltage () const [pure virtual]

electrical potential at the node

7.38.1.2 virtual double eps::IDCElement::GetAmps () const [pure virtual]

either amps produced by a power source or amps flowing through a node

7.38.1.3 virtual double eps::IDCElement::GetPower () const [pure virtual]

7.38.1.4 virtual bool eps::IDCElement::DrawPower (double *fPower*) const [pure virtual]

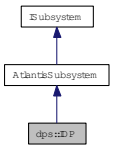
The documentation for this class was generated from the following file:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/eps/eps_defs.h

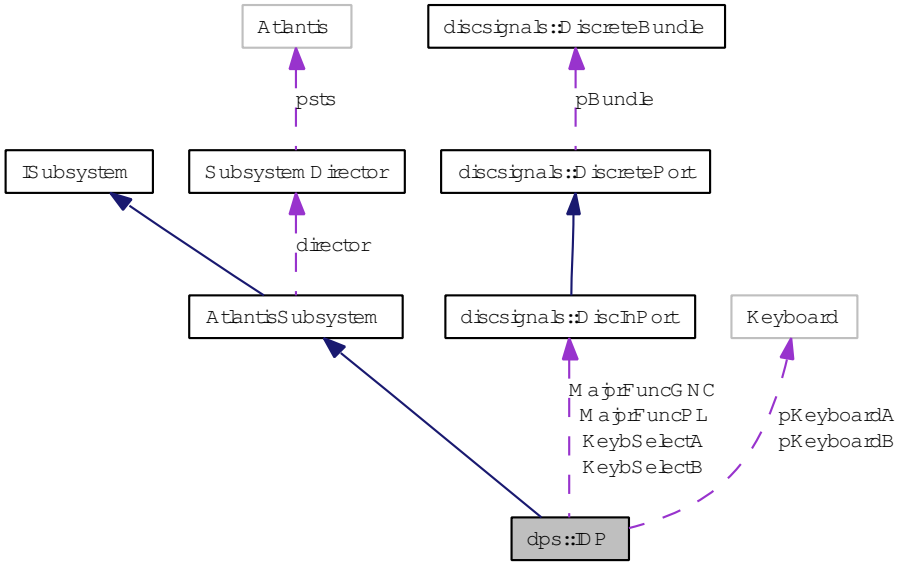
7.39 dps::IDP Class Reference

`#include <IDP.h>`

Inheritance diagram for dps::IDP:



Collaboration diagram for dps::IDP:



Public Member Functions

- IDP (SubsystemDirector *pDirect, const string &_ident, unsigned short _usIDPID)
- virtual ~IDP ()
- void IPL ()
- void ConnectToMDU (vc::PMDU pMDU, bool bPrimary=true)
- void ConnectToKeyboard (Keyboard *pKeyboardA, Keyboard *pKeyboardB)
- unsigned short GetIDPID () const

Private Attributes

- unsigned short usIDPID
- vc::PMDU mdu_list [7]
- MAJORFUNCTION majfunc
- unsigned short usOPS
- unsigned short usSPEC
- unsigned short usDISP
- Keyboard * pKeyboardA
- Keyboard * pKeyboardB
- bool bUseKeyboardA
- char cScratchPadLine [120]
- short sScratchPadLength
- unsigned short usGPCDriver
- unsigned short usSelectedFC
- DiscInPort KeybSelectA
- DiscInPort KeybSelectB
- DiscInPort MajorFuncGNC
- DiscInPort MajorFuncPL
- map< unsigned short, IDPSoftware * > ipl_software
- map< unsigned short, IDPSoftware * > gnc_ops
- map< unsigned short, IDPSoftware * > gnc_specdisp
- map< unsigned short, IDPSoftware * > sm_ops
- map< unsigned short, IDPSoftware * > sm_specdisp
- map< unsigned short, IDPSoftware * > pl_ops
- map< unsigned short, IDPSoftware * > pl_specdisp
- map< unsigned short, IDPSoftware * > bfs_ops
- map< unsigned short, IDPSoftware * > bfs_specdisp

7.39.1 Detailed Description

Implementation of the Integrated display processor. Each can deal with a infinite number of MDUs.

7.39.2 Constructor & Destructor Documentation

7.39.2.1 dps::IDP::IDP (SubsystemDirector * *pDirect*, const string & *_ident*, unsigned short *_usIDPID*)

7.39.2.2 dps::IDP::~IDP () [virtual]

7.39.3 Member Function Documentation

7.39.3.1 void dps::IDP::IPL ()

Perform a initial software load. Basically just reset software configuration to basic and request critical format data from assigned GPC.

7.39.3.2 void dps::IDP::ConnectToMDU (vc::PMDU *pMDU*, bool *bPrimary* = `true`)

7.39.3.3 void dps::IDP::ConnectToKeyboard (Keyboard ∗ *pKeyboardA*, Keyboard ∗ *pKeyboardB*)

7.39.3.4 unsigned short dps::IDP::GetIDPID () const

7.39.4 Member Data Documentation

7.39.4.1 unsigned short dps::IDP::usIDPID `[private]`

7.39.4.2 vc::PMDU dps::IDP::mdu_list[7] `[private]`

7.39.4.3 MAJORFUNCTION dps::IDP::majfunc `[private]`

7.39.4.4 unsigned short dps::IDP::usOPS `[private]`

7.39.4.5 unsigned short dps::IDP::usSPEC `[private]`

7.39.4.6 unsigned short dps::IDP::usDISP `[private]`

7.39.4.7 Keyboard∗ dps::IDP::pKeyboardA `[private]`

7.39.4.8 Keyboard∗ dps::IDP::pKeyboardB `[private]`

7.39.4.9 bool dps::IDP::bUseKeyboardA `[private]`

7.39.4.10 char dps::IDP::cScratchPadLine[120] `[private]`

7.39.4.11 short dps::IDP::sScratchPadLength `[private]`

7.39.4.12 unsigned short dps::IDP::usGPCDriver `[private]`

7.39.4.13 unsigned short dps::IDP::usSelectedFC `[private]`

7.39.4.14 DiscInPort dps::IDP::KeybSelectA `[private]`

7.39.4.15 DiscInPort dps::IDP::KeybSelectB `[private]`

7.39.4.16 DiscInPort dps::IDP::MajorFuncGNC `[private]`

7.39.4.17 DiscInPort dps::IDP::MajorFuncPL `[private]`

7.39.4.18 map<unsigned short, IDPSoftware∗> dps::IDP::ipl_software `[private]`

7.39.4.19 map<unsigned short, IDPSoftware∗> dps::IDP::gnc_ops `[private]`

7.39.4.20 map<unsigned short, IDPSoftware∗> dps::IDP::gnc_specdisp `[private]`

7.39.4.21 map<unsigned short, IDPSoftware∗> dps::IDP::sm_ops `[private]`

7.39.4.22 map<unsigned short, IDPSoftware∗> dps::IDP::sm_specdisp `[private]`

7.39.4.23 map<unsigned short, IDPSoftware∗> dps::IDP::pl_ops `[private]`

7.39.4.24 map<unsigned short, IDPSoftware∗> dps::IDP::pl_specdisp `[private]`

7.39.4.25 map<unsigned short, IDPSoftware∗> dps::IDP::bfs_ops `[private]`

7.39.4.26 map<unsigned short, IDPSoftware∗> dps::IDP::bfs_specdisp `[private]`

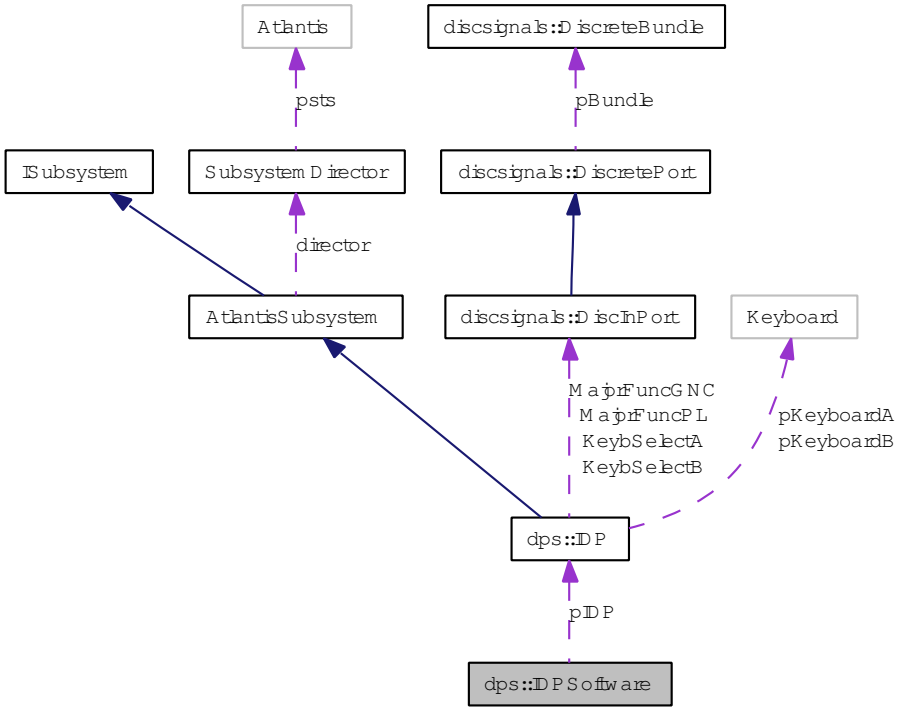
The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/IDP.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/IDP.cpp

7.40 dps::IDPSoftware Class Reference

`#include <IDP.h>`

Collaboration diagram for dps::IDPSoftware:



Public Member Functions

- IDPSoftware (IDP *hardware)
- virtual ~IDPSoftware ()
- virtual void Run (double fSimT, double fDeltaT)

Private Attributes

- IDP * pIDP

7.40.1 Constructor & Destructor Documentation

7.40.1.1 dps::IDPSoftware::IDPSoftware (IDP ∗ *hardware*)

7.40.1.2 virtual dps::IDPSoftware::∼IDPSoftware () [virtual]

7.40.2 Member Function Documentation

7.40.2.1 virtual void dps::IDPSoftware::Run (double *fSimT*, double *fDeltaT*) [virtual]

7.40.3 Member Data Documentation

7.40.3.1 IDP∗ dps::IDPSoftware::pIDP [private]

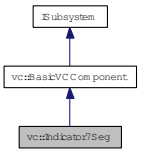
The documentation for this class was generated from the following file:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/IDP.h

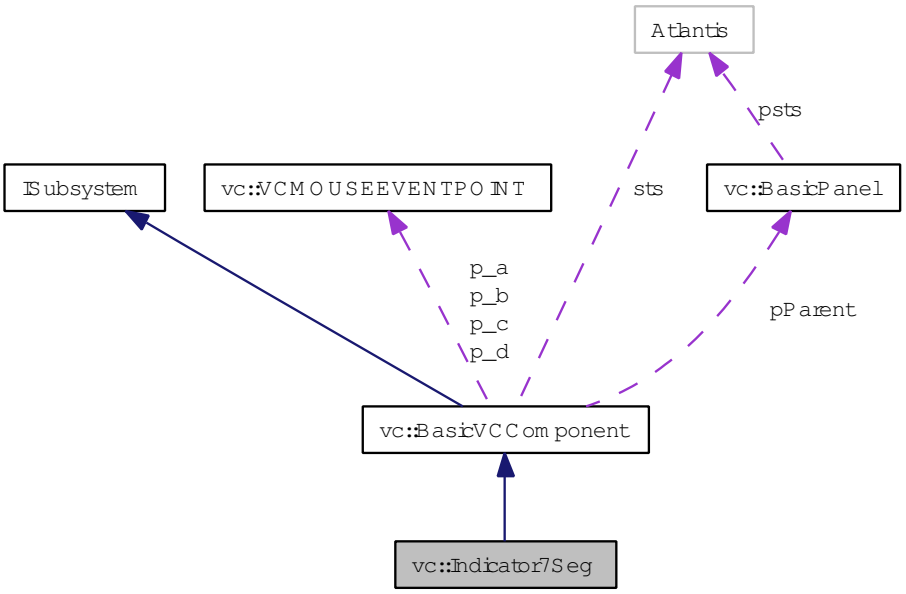
7.41 vc::Indicator7Seg Class Reference

`#include <Indicator7Seg.h>`

Inheritance diagram for vc::Indicator7Seg:



Collaboration diagram for vc::Indicator7Seg:



Public Member Functions

- Indicator7Seg (Atlantis *_sts)
- virtual ~Indicator7Seg (void)

7.41.1 Detailed Description

Implementation of a seven segment indicator. Each indicator does a single digit.

7.41.2 Constructor & Destructor Documentation

7.41.2.1 vc::Indicator7Seg::Indicator7Seg (Atlantis * *_sts*)

7.41.2.2 vc::Indicator7Seg::~Indicator7Seg (void) [virtual]

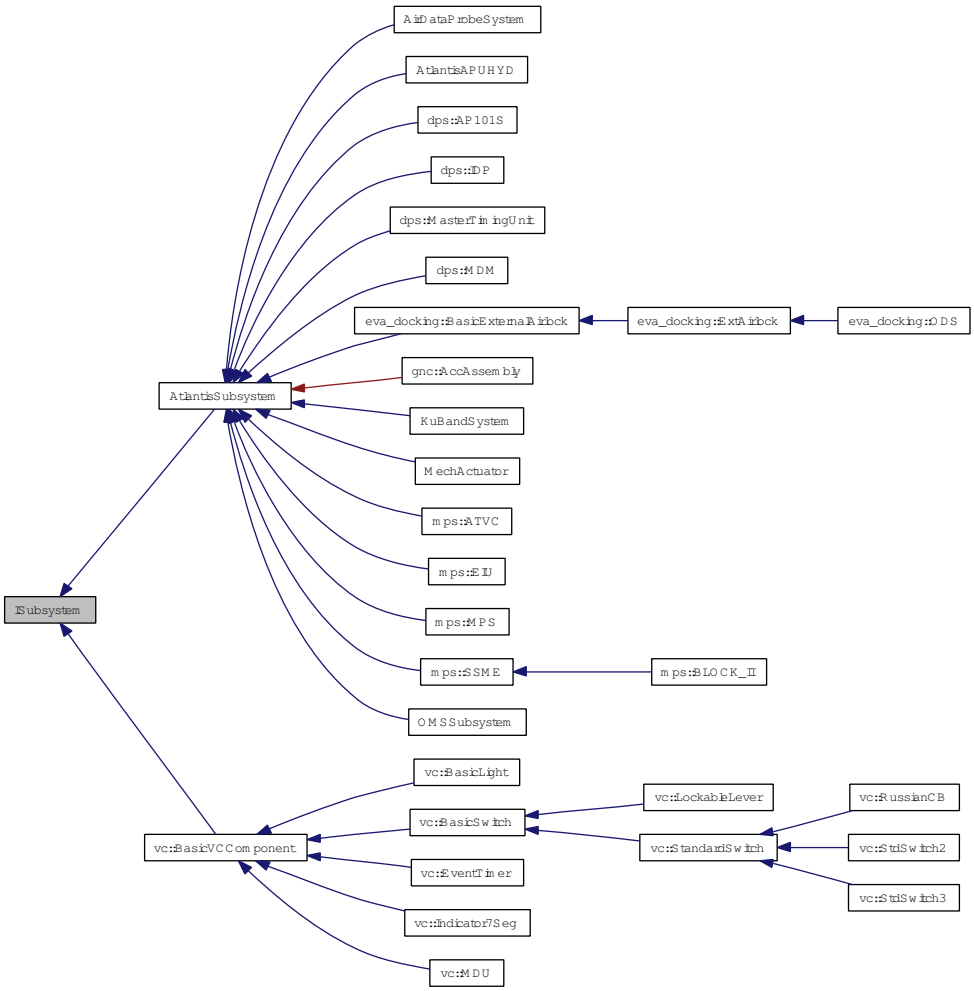
The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/Indicator7Seg.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/Indicator7Seg.cpp

7.42 ISubsystem Class Reference

```
#include <ISubsystem.h>
```

Inheritance diagram for ISubsystem:



Public Member Functions

- virtual void Realize ()=0
- virtual bool RegisterComponent (SubsystemComponent *pComponent)=0
- virtual bool DelComponent (SubsystemComponent *pComponent)=0
- virtual unsigned long CountComponents () const =0
- virtual void OnPreStep (double fSimT, double fDeltaT, double fMJD)=0
- virtual void OnPostStep (double fSimT, double fDeltaT, double fMJD)=0
- virtual void OnPropagate (double fSimT, double fDeltaT, double fMJD)=0
- virtual void OnPlaybackEvent (double fSimT, double fEventT, const char *event_t, const char *event)=0

- virtual bool OnParseLine (const char *line)=0
- virtual void OnSaveState (FILEHANDLE scn) const =0
- virtual void OnDumpToLog () const =0
- virtual const string & GetQualifiedIdentifier () const =0
- virtual const string & GetIdentifier () const =0
- virtual bool OnFail (const string &strFailureCode)=0

7.42.1 Member Function Documentation

7.42.1.1 virtual void ISubsystem::Realize () [pure virtual]

Override this function to create all Orbiter objects required: Thrusters, animations, meshes.

Implemented in AtlantisSubsystem, eva_docking::BasicExternalAirlock, mps::SSME, vc::BasicVCComponent, and vc::StandardSwitch.

7.42.1.2 virtual bool ISubsystem::RegisterComponent (SubsystemComponent * *pComponent*) [pure virtual]

Implemented in AtlantisSubsystem, and vc::BasicVCComponent.

7.42.1.3 virtual bool ISubsystem::DelComponent (SubsystemComponent * *pComponent*) [pure virtual]

Implemented in AtlantisSubsystem, and vc::BasicVCComponent.

7.42.1.4 virtual unsigned long ISubsystem::CountComponents () const [pure virtual]

Implemented in AtlantisSubsystem, and vc::BasicVCComponent.

7.42.1.5 virtual void ISubsystem::OnPreStep (double *fSimT*, double *fDeltaT*, double *fMJD*) [pure virtual]

Implemented in AirDataProbeSystem, AtlantisSubsystem, dps::MasterTimingUnit, eva_docking::BasicExternalAirlock, MechActuator, and vc::BasicVCComponent.

7.42.1.6 virtual void ISubsystem::OnPostStep (double *fSimT*, double *fDeltaT*, double *fMJD*) [pure virtual]

Implemented in AirDataProbeSystem, AtlantisSubsystem, eva_docking::BasicExternalAirlock, MechActuator, mps::ATVC, mps::EIU, mps::MPS, mps::SSME, and vc::BasicVCComponent.

7.42.1.7 virtual void ISubsystem::OnPropagate (double *fSimT*, double *fDeltaT*, double *fMJD*) [pure virtual]

Implemented in AirDataProbeSystem, AtlantisSubsystem, dps::MasterTimingUnit, eva_docking::BasicExternalAirlock, MechActuator, and vc::BasicVCComponent.

7.42.1.8 virtual void ISubsystem::OnPlaybackEvent (double *fSimT*, double *fEventT*, const char * *event_t*, const char * *event*) [pure virtual]

Implemented in AtlantisSubsystem, eva_docking::BasicExternalAirlock, and vc::BasicVCComponent.

7.42.1.9 virtual bool ISubsystem::OnParseLine (const char * *line*) [pure virtual]

Implemented in AtlantisSubsystem, dps::MasterTimingUnit, eva_docking::BasicExternalAirlock, mps::BLOCK_II, mps::MPS, mps::SSME, and vc::BasicVCComponent.

7.42.1.10 virtual void ISubsystem::OnSaveState (FILEHANDLE *scn*) const [pure virtual]

Implemented in AtlantisSubsystem, dps::MasterTimingUnit, eva_docking::BasicExternalAirlock, mps::BLOCK_II, mps::MPS, mps::SSME, vc::BasicSwitch, and vc::BasicVCComponent.

7.42.1.11 virtual void ISubsystem::OnDumpToLog () const [pure virtual]

Implemented in AtlantisSubsystem, eva_docking::BasicExternalAirlock, and vc::BasicVCComponent.

7.42.1.12 virtual const string& ISubsystem::GetQualifiedIdentifier () const [pure virtual]

Implemented in AtlantisSubsystem, and vc::BasicVCComponent.

7.42.1.13 virtual const string& ISubsystem::GetIdentifier () const [pure virtual]

Implemented in AtlantisSubsystem, and vc::BasicVCComponent.

7.42.1.14 virtual bool ISubsystem::OnFail (const string & *strFailureCode*) [pure virtual]

Implemented in AtlantisSubsystem, and vc::BasicVCComponent.

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

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/ISubsystem.h

7.43 Keyboard Class Reference

`#include <Keyboard.h>`

Collaboration diagram for Keyboard:

Public Member Functions

- Keyboard (Atlantis *_sts, int id)
- void RegisterVC ()
- bool VCMouseEvent (int id, int event, VECTOR3 &p)
- void ClearInput ()

Private Attributes

- Atlantis * sts
- int keyboard

Friends

- class Atlantis
- class CRT

7.43.1 Constructor & Destructor Documentation

7.43.1.1 Keyboard::Keyboard (Atlantis * *_sts*, int *id*)

7.43.2 Member Function Documentation

7.43.2.1 void Keyboard::RegisterVC ()

7.43.2.2 bool Keyboard::VCMouseEvent (int *id*, int *event*, VECTOR3 & *p*)

7.43.2.3 void Keyboard::ClearInput ()

7.43.3 Friends And Related Function Documentation

7.43.3.1 friend class Atlantis [friend]

7.43.3.2 friend class CRT [friend]

7.43.4 Member Data Documentation

7.43.4.1 Atlantis* Keyboard::sts [private]

7.43.4.2 int Keyboard::keyboard [private]

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

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Keyboard.h

7.44 KeyboardInput Struct Reference

```
#include <Atlantis.h>
```

Public Attributes

- bool OPS
- bool ITEM
- bool SPEC
- bool EXEC
- bool PRO
- bool NewEntry
- char input [255]
- int InputSize

7.44.1 Member Data Documentation

7.44.1.1 bool KeyboardInput::OPS

7.44.1.2 bool KeyboardInput::ITEM

7.44.1.3 bool KeyboardInput::SPEC

7.44.1.4 bool KeyboardInput::EXEC

7.44.1.5 bool KeyboardInput::PRO

7.44.1.6 bool KeyboardInput::NewEntry

7.44.1.7 char KeyboardInput::input[255]

7.44.1.8 int KeyboardInput::InputSize

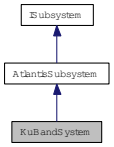
The documentation for this struct was generated from the following file:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Atlantis.h

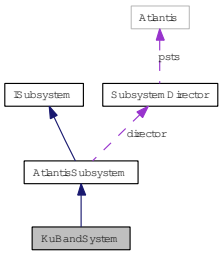
7.45 KuBandSystem Class Reference

`#include <KuBandSystem.h>`

Inheritance diagram for KuBandSystem:



Collaboration diagram for KuBandSystem:



Public Member Functions

- virtual double GetAzimuth ()
- virtual bool SetSlew (double fAzimuth, double fElevation)
- virtual double GetSignalStrength ()
- KuBandSystem (SubsystemDirector *_director)
- virtual ~KuBandSystem ()
- virtual bool ParseLine (const char *line)
- virtual void SaveState (FILEHANDLE scn)
- virtual void PreStep (double fSimT, double fDeltaT)
- virtual void PostStep (double fSimT, double fDeltaT)
- virtual void Propagate (double fSimT, double fDeltaT)

Private Attributes

- unsigned short usGPCMode [2]
- double alpha_angle [2]
- double beta_angle [2]
- double fDeployPos [2]
- AnimState DeployState [2]
- AnimState GimbalLockState [2]
- double fGimbalLockState [2]
- bool bHeatersOn [2]

7.45.1 Constructor & Destructor Documentation

7.45.1.1 KuBandSystem::KuBandSystem (SubsystemDirector * *_director*)

7.45.1.2 KuBandSystem::~KuBandSystem () [virtual]

7.45.2 Member Function Documentation

7.45.2.1 double KuBandSystem::GetAzimuth () [virtual]

7.45.2.2 bool KuBandSystem::SetSlew (double *fAzimuth*, double *fElevation*) [virtual]

7.45.2.3 double KuBandSystem::GetSignalStrength () [virtual]

7.45.2.4 bool KuBandSystem::ParseLine (const char * *line*) [virtual]

7.45.2.5 void KuBandSystem::SaveState (FILEHANDLE *scn*) [virtual]

7.45.2.6 void KuBandSystem::PreStep (double *fSimT*, double *fDeltaT*) [virtual]

7.45.2.7 void KuBandSystem::PostStep (double *fSimT*, double *fDeltaT*) [virtual]

7.45.2.8 void KuBandSystem::Propagate (double *fSimT*, double *fDeltaT*) [virtual]

7.45.3 Member Data Documentation

7.45.3.1 unsigned short KuBandSystem::usGPCMode[2] [private]

7.45.3.2 double KuBandSystem::alpha_angle[2] [private]

7.45.3.3 double KuBandSystem::beta_angle[2] [private]

7.45.3.4 double KuBandSystem::fDeployPos[2] [private]

7.45.3.5 AnimState KuBandSystem::DeployState[2] [private]

7.45.3.6 AnimState KuBandSystem::GimbalLockState[2] [private]

7.45.3.7 double KuBandSystem::fGimbalLockState[2] [private]

7.45.3.8 bool KuBandSystem::bHeatersOn[2] [private]

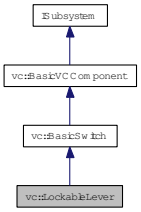
The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/KuBandSystem.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/KuBandSystem.cpp

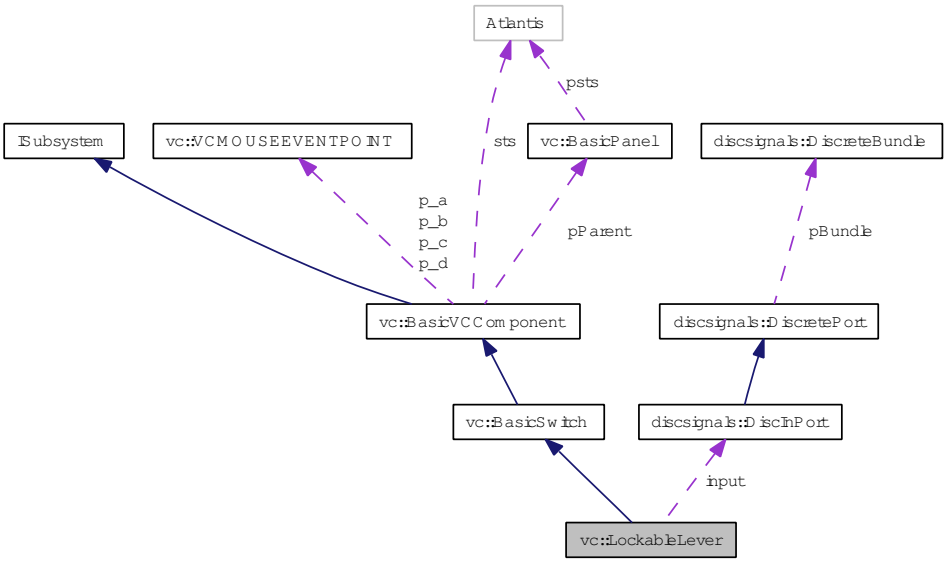
7.46 vc::LockableLever Class Reference

`#include <StandardSwitch.h>`

Inheritance diagram for vc::LockableLever:



Collaboration diagram for vc::LockableLever:



Public Member Functions

- LockableLever (Atlantis *psts, unsigned short usNumPositions, const string &_ident)
- virtual ~LockableLever ()

Private Attributes

- UINT grpIndex
- MGROUP_ROTATE * pswitchrot
- MGROUP_TRANSLATE * pswitchpull

- bool bIsPulled
- double fInitialMousePos
- UINT anim_switch
- DiscInPort input

7.46.1 Constructor & Destructor Documentation

7.46.1.1 vc::LockableLever::LockableLever (Atlantis * *psts*, unsigned short *usNumPositions*, const string & *_ident*)

7.46.1.2 virtual vc::LockableLever::~LockableLever () [virtual]

7.46.2 Member Data Documentation

7.46.2.1 UINT vc::LockableLever::grpIndex [private]

7.46.2.2 MGROUP_ROTATE* vc::LockableLever::pswitchrot [private]

7.46.2.3 MGROUP_TRANSLATE* vc::LockableLever::pswitchpull [private]

7.46.2.4 bool vc::LockableLever::bIsPulled [private]

7.46.2.5 double vc::LockableLever::fInitialMousePos [private]

7.46.2.6 UINT vc::LockableLever::anim_switch [private]

7.46.2.7 DiscInPort vc::LockableLever::input [private]

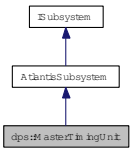
The documentation for this class was generated from the following file:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/StandardSwitch.h

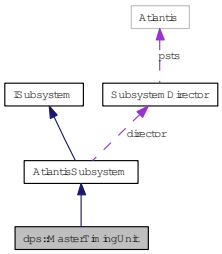
7.47 dps::MasterTimingUnit Class Reference

`#include <MasterTimingUnit.h>`

Inheritance diagram for dps::MasterTimingUnit:



Collaboration diagram for dps::MasterTimingUnit:



Public Member Functions

- MasterTimingUnit (SubsystemDirector *_director)
- virtual ~MasterTimingUnit ()
- short GetEventTimerSec (MTU_EVTTMR_INDEX timer) const
- short GetEventTimerMin (MTU_EVTTMR_INDEX timer) const
- void SetEventTimer (MTU_EVTTMR_INDEX timer, short Minutes, short Seconds)
- void StartEventTimer (MTU_EVTTMR_INDEX timer)
- void StopEventTimer (MTU_EVTTMR_INDEX timer)
- void SetEventTimerMode (MTU_EVTTMR_INDEX timer, EVENT_TIMER_MODE mode)
- void ResetEventTimer (MTU_EVTTMR_INDEX timer)
- short GetMETMilli (MTU_ACCU_INDEX accu_index) const
- short GetMETSec (MTU_ACCU_INDEX accu_index) const
- short GetMETMin (MTU_ACCU_INDEX accu_index) const
- short GetMETHour (MTU_ACCU_INDEX accu_index) const
- short GetMETDay (MTU_ACCU_INDEX accu_index) const
- void ResetMET ()
- void StartMET ()
- short GetGMTMilli (MTU_ACCU_INDEX accu_index) const
- short GetGMTSec (MTU_ACCU_INDEX accu_index) const
- short GetGMTMin (MTU_ACCU_INDEX accu_index) const
- short GetGMTHour (MTU_ACCU_INDEX accu_index) const

- short GetGMTDay (MTU_ACCU_INDEX accu_index) const
- virtual void OnPreStep (double fSimT, double fDeltaT, double fMJD)
- virtual void OnPropagate (double fSimT, double fDeltaT, double fMJD)
- virtual void OnSaveState (FILEHANDLE scn) const
- virtual bool OnParseLine (const char *line)

Private Attributes

- double fGMT [3][2]
- double fMET [3][2]
- double fEvent [2][2]
- short sEventMinutes [2]
- short sEventSeconds [2]
- short sGMTMillis [3]
- short sGMTSeconds [3]
- short sGMTMinutes [3]
- short sGMTHours [3]
- short sGMTDays [3]
- short sMETMillis [3]
- short sMETSeconds [3]
- short sMETMinutes [3]
- short sMETHours [3]
- short sMETDays [3]
- EVENT_TIMER_MODE event_mode [2][2]
- EVENT_TIMER_CONTROL event_control [2][2]
- bool bMETCounting [2]

7.47.1 Constructor & Destructor Documentation

7.47.1.1 dps::MasterTimingUnit::MasterTimingUnit (SubsystemDirector * *_director*)

7.47.1.2 dps::MasterTimingUnit::~MasterTimingUnit () [virtual]

7.47.2 Member Function Documentation

7.47.2.1 short dps::MasterTimingUnit::GetEventTimerSec (MTU_EVTTMR_INDEX *timer*) const

7.47.2.2 short dps::MasterTimingUnit::GetEventTimerMin (MTU_EVTTMR_INDEX *timer*) const

7.47.2.3 void dps::MasterTimingUnit::SetEventTimer (MTU_EVTTMR_INDEX *timer*, short *Minutes*, short *Seconds*)

7.47.2.4 void dps::MasterTimingUnit::StartEventTimer (MTU_EVTTMR_INDEX *timer*)

7.47.2.5 void dps::MasterTimingUnit::StopEventTimer (MTU_EVTTMR_INDEX *timer*)

7.47.2.6 void dps::MasterTimingUnit::SetEventTimerMode (MTU_EVTTMR_INDEX *timer*, EVENT_TIMER_MODE *mode*)

7.47.2.7 void dps::MasterTimingUnit::ResetEventTimer (MTU_EVTTMR_INDEX *timer*)

7.47.2.8 short dps::MasterTimingUnit::GetMETMilli (MTU_ACCU_INDEX *accu_index*) const

7.47.2.9 short dps::MasterTimingUnit::GetMETSec (MTU_ACCU_INDEX *accu_index*) const

7.47.2.10 short dps::MasterTimingUnit::GetMETMin (MTU_ACCU_INDEX *accu_index*) const

7.47.2.11 short dps::MasterTimingUnit::GetMETHour (MTU_ACCU_INDEX *accu_index*) const

7.47.2.12 short dps::MasterTimingUnit::GetMETDay (MTU_ACCU_INDEX *accu_index*) const

7.47.2.13 void dps::MasterTimingUnit::ResetMET ()

7.47.2.14 void dps::MasterTimingUnit::StartMET ()

7.47.2.15 short dps::MasterTimingUnit::GetGMTMilli (MTU_ACCU_INDEX *accu_index*) const

7.47.2.16 short dps::MasterTimingUnit::GetGMTSec (MTU_ACCU_INDEX *accu_index*) const

7.47.2.17 short dps::MasterTimingUnit::GetGMTMin (MTU_ACCU_INDEX *accu_index*) const

7.47.2.18 short dps::MasterTimingUnit::GetGMTHour (MTU_ACCU_INDEX *accu_index*) const

7.47.2.19 short dps::MasterTimingUnit::GetGMTDay (MTU_ACCU_INDEX *accu_index*) const

7.47.2.20 void dps::MasterTimingUnit::OnPreStep (double *fSimT*, double *fDeltaT*, double *fMJD*) [virtual]

Prepare and execute calculations before the propagation of the state vector

Parameters:

fSimT current simulation time

fDeltaT Simulation time step length

fMJD current modified julian date

Reimplemented from AtlantisSubsystem.

7.47.2.21 void dps::MasterTimingUnit::OnPropagate (double *fSimT*, double *fDeltaT*, double *fMJD*) [virtual]

Propagate the state vector from old state to new state.

Parameters:

fSimT current simulation time

fDeltaT Simulation time step length

fMJD current modified julian date

Reimplemented from AtlantisSubsystem.

7.47.2.22 void dps::MasterTimingUnit::OnSaveState (FILEHANDLE *scn*) const [virtual]

Save the state of the subsystem

Reimplemented from AtlantisSubsystem.

7.47.2.23 bool dps::MasterTimingUnit::OnParseLine (const char * *line*) [virtual]

Parse a line of the scenario file.

Warning:

Should get replaced by a block oriented scenario file format, like NASSP, making this function deprecated.

Reimplemented from AtlantisSubsystem.

7.47.3 Member Data Documentation

7.47.3.1 double dps::MasterTimingUnit::fGMT[3][2] [private]

7.47.3.2 double dps::MasterTimingUnit::fMET[3][2] [private]

7.47.3.3 double dps::MasterTimingUnit::fEvent[2][2] [private]

7.47.3.4 short dps::MasterTimingUnit::sEventMinutes[2] [private]

7.47.3.5 short dps::MasterTimingUnit::sEventSeconds[2] [private]

7.47.3.6 short dps::MasterTimingUnit::sGMTMillis[3] [private]

7.47.3.7 short dps::MasterTimingUnit::sGMTSeconds[3] [private]

7.47.3.8 short dps::MasterTimingUnit::sGMTMinutes[3] [private]

7.47.3.9 short dps::MasterTimingUnit::sGMTHours[3] [private]

7.47.3.10 short dps::MasterTimingUnit::sGMTDays[3] [private]

7.47.3.11 short dps::MasterTimingUnit::sMETMillis[3] [private]

7.47.3.12 short dps::MasterTimingUnit::sMETSeconds[3] [private]

7.47.3.13 short dps::MasterTimingUnit::sMETMinutes[3] [private]

7.47.3.14 short dps::MasterTimingUnit::sMETHours[3] [private]

7.47.3.15 short dps::MasterTimingUnit::sMETDays[3] [private]

7.47.3.16 EVENT_TIMER_MODE dps::MasterTimingUnit::event_mode[2][2] [private]

7.47.3.17 EVENT_TIMER_CONTROL dps::MasterTimingUnit::event_control[2][2] [private]

7.47.3.18 bool dps::MasterTimingUnit::bMETCounting[2] [private]

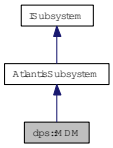
The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/MasterTimingUnit.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/MasterTimingUnit.cpp

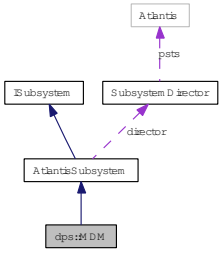
7.48 dps::MDM Class Reference

`#include <MDM.h>`

Inheritance diagram for dps::MDM:



Collaboration diagram for dps::MDM:



Public Member Functions

- MDM (SubsystemDirector *_director, const string &_ident)
- virtual ~MDM ()

Protected Attributes

- short SCU_PROM [2][16384]
- MDMIOModule * pmodules [16]

7.48.1 Detailed Description

Possible modules per MDM (maximal 16 modules):

Discrete output high (DOH), 28 volts, three 16-bit channels Discrete output low (DOL), 5 volts, three 16-bit channels Discrete input high (DIH), 28 volts, three 16-bit channels Discrete input low (DIL), 5 volts, three 16-bit channels Analog output differential (AOD), 16 channels Analog input differential (AID), 16 channels Analog input single-ended (AIS), 32 channels Serial input/output (SIO), four channels Tacan/radar altimeter (special-purpose I/O)

FF MDM: Modules 5 and 13 connect to the RJDF - likely DOH/DIL modules

7.48.2 Constructor & Destructor Documentation

7.48.2.1 dps::MDM::MDM (SubsystemDirector * *_director*, const string & *_ident*)

7.48.2.2 dps::MDM::~MDM () [virtual]

7.48.3 Member Data Documentation

7.48.3.1 short dps::MDM::SCU_PROM[2][16384] [protected]

7.48.3.2 MDMIOModule* dps::MDM::pmodules[16] [protected]

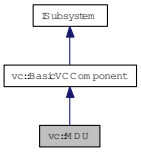
The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/MDM.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/MDM.cpp

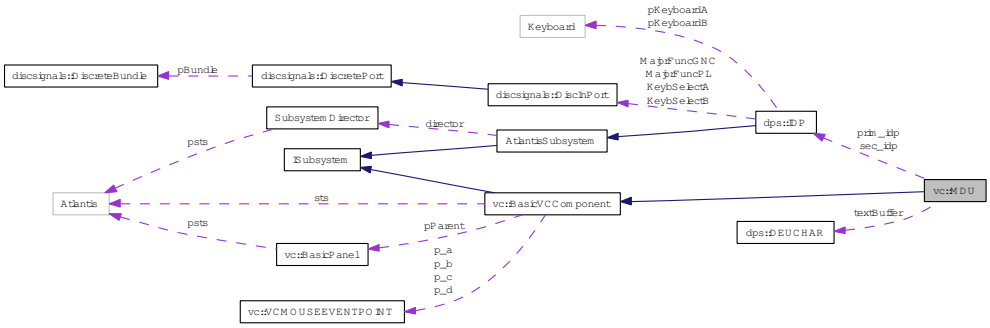
7.49 vc::MDU Class Reference

`#include <MDU.h>`

Inheritance diagram for vc::MDU:



Collaboration diagram for vc::MDU:



Public Types

- enum MDU_MODE {
  MDU_ADI_AVVI, MDU_HSI_AMI, MDU_COMP_ADI_HSI, MDU_ADI, MDU_OMS_MPS, MDU_HYD_APU, MDU_SPI, MDU_PORT_SELECT, MDU_DPS, MDU_FAULTSUMM, MDU_CONFIGSTATUS, MDU_CST, MDU_MEMORYMGMT }

Public Member Functions

- MDU (Atlantis *_sts, const string &_ident, unsigned short usMDUID, bool _bUseCRTMFD=true)
- virtual ~MDU ()
- virtual char * GetEdgeKeyMenuLabel (int iButton)
- bool SetPrimaryIDP (dps::IDP *idp)
- bool SetSecondaryIDP (dps::IDP *idp)
- virtual void DefineVCAnimations (UINT vc_idx)
- virtual bool OnMouseEvent (int _event, float x, float y)
- virtual bool Paint (HDC hdc)

- virtual bool PaintEdgeMenu (HDC hdc)
- virtual bool RealizeMFD ()
- virtual void RegisterVC ()
- virtual bool DefineRegionAID (UINT aid)
- virtual bool DefineVCGroup (UINT mgrp)
- virtual bool DefineVCTexture (SURFHANDLE tex)
- virtual bool IsCRTBufferEnabled () const
- virtual void UpdateTextBuffer ()
- virtual void PrintToBuffer (char *string, int length, int col, int row, char attributes)
- virtual bool GetViewAngle () const
- virtual const string & GetEdgekeyMenu () const
- virtual short GetPortConfig () const
- virtual bool GetSelectedPort () const
- virtual unsigned short GetDrivingIDP () const
- virtual void ConnectToCRTMFD ()

Protected Member Functions

- virtual void RegisterMFDContext ()
- virtual void SwitchMFDMode ()

Protected Attributes

- unsigned short usMDUID
- bool bUseCRTMFD
- SURFHANDLE shBackGround
- SURFHANDLE shLabelTex
- dps::DEUCHAR textBuffer [51][26]
- dps::IDP * prim_idp
- dps::IDP * sec_idp
- bool bInverseX
- bool bUseSecondaryPort
- UINT myGroup
- EXTMFDSPEC mfdspec
- double fBrightness
- bool bPower
- bool bIsConnectedToCRTMFD

Private Attributes

- double t0
- bool counting

7.49.1 Member Enumeration Documentation

7.49.1.1 enum vc::MDU::MDU_MODE

Enumerator:

MDU_ADI_AVVI

MDU_HSI_AMI

MDU_COMP_ADI_HSI

MDU_ADI

MDU_OMS_MPS

MDU_HYD_APU

MDU_SPI

MDU_PORT_SELECT

MDU_DPS

MDU_FAULTSUMM

MDU_CONFIGSTATUS

MDU_CST

MDU_MEMORYMGMT

7.49.2 Constructor & Destructor Documentation

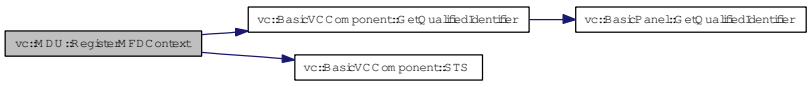
7.49.2.1 vc::MDU::MDU (Atlantis * *_sts*, const string & *_ident*, unsigned short *usMDUID*, bool *_bUseCRTMFD* = `true`)

7.49.2.2 vc::MDU::~MDU () `[virtual]`

7.49.3 Member Function Documentation

7.49.3.1 void vc::MDU::RegisterMFDContext () `[protected, virtual]`

Here is the call graph for this function:



7.49.3.2 void vc::MDU::SwitchMFDMode () [protected, virtual]

7.49.3.3 char * vc::MDU::GetEdgeKeyMenuLabel (int *iButton*) [virtual]

7.49.3.4 bool vc::MDU::SetPrimaryIDP (dps::IDP * *idp*)

7.49.3.5 bool vc::MDU::SetSecondaryIDP (dps::IDP * *idp*)

7.49.3.6 void vc::MDU::DefineVCAnimations (UINT *vc_idx*) [virtual]

Does nothing as it currently has no animations.

See also:

BasicVCComponent::DefineVCAnimations

Reimplemented from vc::BasicVCComponent.

7.49.3.7 bool vc::MDU::OnMouseEvent (int *_event*, float *x*, float *y*) [virtual]

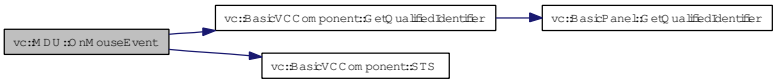
handle mouse events

See also:

BasicVCComponent::OnMouseEvent

Reimplemented from vc::BasicVCComponent.

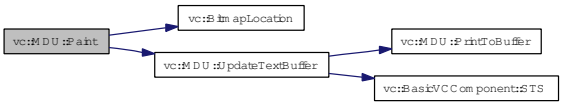
Here is the call graph for this function:



7.49.3.8 bool vc::MDU::Paint (HDC *hdc*) [virtual]

Paint the contents of the MDU on a device context.

Here is the call graph for this function:



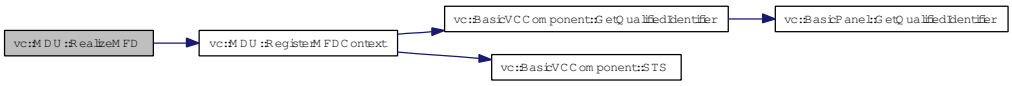
7.49.3.9 bool vc::MDU::PaintEdgeMenu (HDC *hdc*) [virtual]

Paint the edge menu area on this DC.

7.49.3.10 bool vc::MDU::RealizeMFD () [virtual]

Register the MFD area for Orbiter. Does nothing when MFD rendering is bypassed.

Here is the call graph for this function:



7.49.3.11 void vc::MDU::RegisterVC () [virtual]

Reimplemented from vc::BasicVCComponent.

7.49.3.12 bool vc::MDU::DefineRegionAID (UINT *aid*) [virtual]

define the Area ID of the MDU screen inside the VC. when in CRTMFD mode, it paints only the edge key menu area.

Here is the call graph for this function:

7.49.3.13 bool vc::MDU::DefineVCGroup (UINT *mgrp*) [virtual]

7.49.3.14 bool vc::MDU::DefineVCTexture (SURFHANDLE *tex*) [virtual]

Links to label texture, when used in conjunction with CRTMFD, expects texture for drawing full MDU when not.

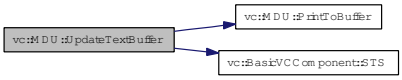
Returns:

false, if failed.

7.49.3.15 bool vc::MDU::IsCRTBufferEnabled () const [virtual]

7.49.3.16 void vc::MDU::UpdateTextBuffer () [virtual]

Here is the call graph for this function:



7.49.3.17 void vc::MDU::PrintToBuffer (char * *string*, int *length*, int *col*, int *row*, char *attributes*) [virtual]

7.49.3.18 bool vc::MDU::GetViewAngle () const [virtual]

7.49.3.19 const string & vc::MDU::GetEdgekeyMenu () const [virtual]

7.49.3.20 short vc::MDU::GetPortConfig () const [virtual]

7.49.3.21 bool vc::MDU::GetSelectedPort () const [virtual]

7.49.3.22 unsigned short vc::MDU::GetDrivingIDP () const [virtual]

Return the identification of the IDP driving this display.

Returns:

0 if not connected to IDP or IDP failed

Here is the call graph for this function:

7.49.3.23 void vc::MDU::ConnectToCRTMFD () [virtual]

Called by CRTMFD or compatible, when registering special functions. Disables the old MFD menu and replaces it by the MDU menus. Gets disabled when the MDU gets reset.

7.49.4 Member Data Documentation

7.49.4.1 double vc::MDU::t0 [private]

7.49.4.2 bool vc::MDU::counting [private]

7.49.4.3 unsigned short vc::MDU::usMDUID [protected]

7.49.4.4 bool vc::MDU::bUseCRTMFD [protected]

7.49.4.5 SURFHANDLE vc::MDU::shBackGround [protected]

7.49.4.6 SURFHANDLE vc::MDU::shLabelTex [protected]

7.49.4.7 dps::DEUCHAR vc::MDU::textBuffer[51][26] [protected]

7.49.4.8 dps::IDP* vc::MDU::prim_idp [protected]

7.49.4.9 dps::IDP* vc::MDU::sec_idp [protected]

7.49.4.10 bool vc::MDU::bInverseX [protected]

7.49.4.11 bool vc::MDU::bUseSecondaryPort [protected]

7.49.4.12 UINT vc::MDU::myGroup [protected]

7.49.4.13 EXTMFDSPEC vc::MDU::mfdspec [protected]

7.49.4.14 double vc::MDU::fBrightness [protected]

7.49.4.15 bool vc::MDU::bPower [protected]

7.49.4.16 bool vc::MDU::bIsConnectedToCRTMFD [protected]

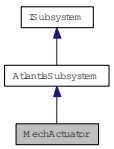
The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/MDU.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/MDU.cpp

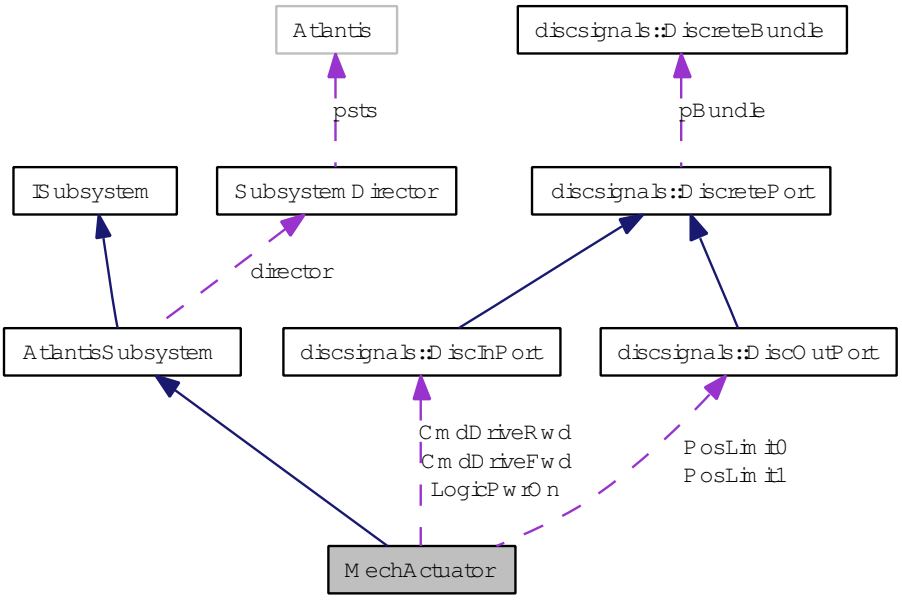
7.50 MechActuator Class Reference

```
#include <MechActuator.h>
```

Inheritance diagram for MechActuator:



Collaboration diagram for MechActuator:



Public Member Functions

- MechActuator (SubsystemDirector *_pDirect, const string &_ident)
- virtual ~MechActuator ()
- virtual void OnPreStep (double fSimT, double fDeltaT, double fMJD)
- virtual void OnPostStep (double fSimT, double fDeltaT, double fMJD)
- virtual void OnPropagate (double fSimT, double fDeltaT, double fMJD)
- void SetMotorParams (double fNormTorque, double fNormAmps, double fNormSpeed, double fNormVoltage)
- void SetObjectParams (double fMinTorque, double fFrictionCoeff, double fMOICoeff)

- void SetObjectAnim (UINT anim)

Private Attributes

- double fOutPos
- discsignals::DiscOutPort PosLimit0
- discsignals::DiscOutPort PosLimit1
- discsignals::DiscInPort CmdDriveFwd
- discsignals::DiscInPort CmdDriveRwd
- discsignals::DiscInPort LogicPwrOn
- double fObjectTorque
- UINT object_anim

7.50.1 Constructor & Destructor Documentation

7.50.1.1 MechActuator::MechActuator (SubsystemDirector * *_pDirect*, const string & *_ident*)

7.50.1.2 virtual MechActuator::~MechActuator () [virtual]

7.50.2 Member Function Documentation

7.50.2.1 virtual void MechActuator::OnPreStep (double *fSimT*, double *fDeltaT*, double *fMJD*) [virtual]

Prepare and execute calculations before the propagation of the state vector

Parameters:

fSimT current simulation time

fDeltaT Simulation time step length

fMJD current modified julian date

Reimplemented from AtlantisSubsystem.

7.50.2.2 virtual void MechActuator::OnPostStep (double *fSimT*, double *fDeltaT*, double *fMJD*) [virtual]

Finish calculations after the propagation of the state vector

Parameters:

fSimT current simulation time

fDeltaT Simulation time step length

fMJD current modified julian date

Reimplemented from AtlantisSubsystem.

7.50.2.3 virtual void MechActuator::OnPropagate (double *fSimT*, double *fDeltaT*, double *fMJD*) [virtual]

Propagate the state vector from old state to new state.

Parameters:

fSimT current simulation time

fDeltaT Simulation time step length

fMJD current modified julian date

Reimplemented from AtlantisSubsystem.

7.50.2.4 void MechActuator::SetMotorParams (double *fNormTorque*, double *fNormAmps*, double *fNormSpeed*, double *fNormVoltage*)

7.50.2.5 void MechActuator::SetObjectParams (double *fMinTorque*, double *fFrictionCoeff*, double *fMOICoeff*)

7.50.2.6 void MechActuator::SetObjectAnim (UINT *anim*)

7.50.3 Member Data Documentation

7.50.3.1 double MechActuator::fOutPos [private]

7.50.3.2 discsignals::DiscOutPort MechActuator::PosLimit0 [private]

7.50.3.3 discsignals::DiscOutPort MechActuator::PosLimit1 [private]

7.50.3.4 discsignals::DiscInPort MechActuator::CmdDriveFwd [private]

7.50.3.5 discsignals::DiscInPort MechActuator::CmdDriveRwd [private]

7.50.3.6 discsignals::DiscInPort MechActuator::LogicPwrOn [private]

7.50.3.7 double MechActuator::fObjectTorque [private]

7.50.3.8 UINT MechActuator::object_anim [private]

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

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/MechActuator.h

7.51 MFD_GDIPARAM Struct Reference

```
#include <CRT.h>
```

Public Attributes

- HINSTANCE hDLL
- HFONT hCRTFont

7.51.1 Member Data Documentation

7.51.1.1 HINSTANCE MFD_GDIPARAM::hDLL

7.51.1.2 HFONT MFD_GDIPARAM::hCRTFont

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

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/CRT.h

7.52 mission::Mission Class Reference

```
#include <Mission.h>
```

Public Member Functions

- Mission ()
- virtual ~Mission ()
- virtual bool LoadMission (const string &strFilename)
- virtual bool SaveMission (const string &strFilename) const
- virtual bool SaveMission () const
- virtual Mission * CloneNewBranch () const
- virtual Mission * CloneRevision () const
- virtual const string & GetOrbiter () const
- virtual const string & GetMissionName () const
- virtual bool CheckValidity () const

Protected Attributes

- string strMissionName
- string strOrbiter
- int iETVersion
- int iBranchCode
- int iRevisionCode
- double fLTimeMJD
- bool bUseRMS

7.52.1 Constructor & Destructor Documentation

7.52.1.1 mission::Mission::Mission ()

7.52.1.2 virtual mission::Mission::~Mission () [virtual]

7.52.2 Member Function Documentation

7.52.2.1 virtual bool mission::Mission::LoadMission (const string & *strFilename*) [virtual]

7.52.2.2 virtual bool mission::Mission::SaveMission (const string & *strFilename*) const [virtual]

7.52.2.3 virtual bool mission::Mission::SaveMission () const [virtual]

7.52.2.4 virtual Mission* mission::Mission::CloneNewBranch () const [virtual]

7.52.2.5 virtual Mission* mission::Mission::CloneRevision () const [virtual]

7.52.2.6 virtual const string& mission::Mission::GetOrbiter () const [virtual]

7.52.2.7 virtual const string& mission::Mission::GetMissionName () const [virtual]

7.52.2.8 virtual bool mission::Mission::CheckValidity () const [virtual]

7.52.3 Member Data Documentation

7.52.3.1 string mission::Mission::strMissionName [protected]

7.52.3.2 string mission::Mission::strOrbiter [protected]

7.52.3.3 int mission::Mission::iETVersion [protected]

7.52.3.4 int mission::Mission::iBranchCode [protected]

7.52.3.5 int mission::Mission::iRevisionCode [protected]

7.52.3.6 double mission::Mission::fLTimeMJD [protected]

7.52.3.7 bool mission::Mission::bUseRMS [protected]

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

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mission/Mission.h

7.53 MLP Class Reference

```
#include <MLP.h>
```

Public Member Functions

- virtual void SignalGSEStart ()
- virtual void TriggerHDP ()
- virtual void ActivateSSS ()
- MLP (OBJHANDLE hVessel, int iFlightModel)
- virtual ~MLP ()
- virtual void clbkSetClassCaps (FILEHANDLE cfg)
- virtual void clbkLoadStateEx (FILEHANDLE scn, void *vs)
- virtual bool clbkLoadVC (int id)
- virtual void clbkSaveState (FILEHANDLE scn)
- virtual void clbkPreStep (double fSimT, double fDeltaT, double mjd)

Private Member Functions

- void CalculateSteamProduction (double fSimT, double fDeltaT)

Private Attributes

- MESHHANDLE mshMLP
- double fCountdown
- bool bStartSequence
- bool bSSS_Active
- double fT_SSSActive
- ATTACHMENTHANDLE ahHDP
- double fSRBSteam
- double fSSMESteam

7.53.1 Constructor & Destructor Documentation

7.53.1.1 MLP::MLP (OBJHANDLE *hVessel*, int *iFlightModel*)

7.53.1.2 MLP::~MLP () [virtual]

7.53.2 Member Function Documentation

7.53.2.1 void MLP::SignalGSEStart () [virtual]

7.53.2.2 void MLP::TriggerHDP () [virtual]

7.53.2.3 void MLP::ActivateSSS () [virtual]

7.53.2.4 void MLP::clbkSetClassCaps (FILEHANDLE *cfg*) [virtual]

7.53.2.5 void MLP::clbkLoadStateEx (FILEHANDLE *scn*, void * *vs*) [virtual]

7.53.2.6 bool MLP::clbkLoadVC (int *id*) [virtual]

7.53.2.7 void MLP::clbkSaveState (FILEHANDLE *scn*) [virtual]

7.53.2.8 void MLP::clbkPreStep (double *fSimT*, double *fDeltaT*, double *mjd*) [virtual]

Here is the call graph for this function:

7.53.2.9 void MLP::CalculateSteamProduction (double *fSimT*, double *fDeltaT*) [private]

7.53.3 Member Data Documentation

7.53.3.1 MESHHANDLE MLP::mshMLP [private]

7.53.3.2 double MLP::fCountdown [private]

7.53.3.3 bool MLP::bStartSequence [private]

7.53.3.4 bool MLP::bSSS_Active [private]

7.53.3.5 double MLP::fT_SSSActive [private]

7.53.3.6 ATTACHMENTHANDLE MLP::ahHDP [private]

7.53.3.7 double MLP::fSRBSteam [private]

7.53.3.8 double MLP::fSSMESteam [private]

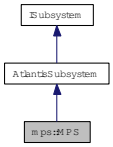
The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/MLP/MLP.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/MLP/MLP.cpp

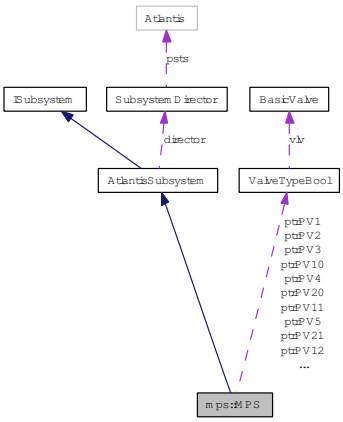
7.54 mps::MPS Class Reference

`#include <MPS.h>`

Inheritance diagram for mps::MPS:



Collaboration diagram for mps::MPS:



Public Member Functions

- MPS (SubsystemDirector *)
- ~MPS (void)
- bool MPSValveOpen (MPS_VALVE)
- bool MPSValveClose (MPS_VALVE)
- double MPSValveStatus (MPS_VALVE)
- virtual void OnPostStep (double, double, double)
- void OnSaveState (FILEHANDLE) const
- bool OnParseLine (const char *)

Private Attributes

- ValveTypeBool * ptrPV1
- ValveTypeBool * ptrPV2
- ValveTypeBool * ptrPV3
- ValveTypeBool * ptrPV4

- ValveTypeBool * ptrPV5
- ValveTypeBool * ptrPV6
- ValveTypeBool * ptrPV7
- ValveTypeBool * ptrPV8
- ValveTypeBool * ptrPV9
- ValveTypeBool * ptrPV10
- ValveTypeBool * ptrPV11
- ValveTypeBool * ptrPV12
- ValveTypeBool * ptrPV13
- ValveTypeBool * ptrPV17
- ValveTypeBool * ptrPV18
- ValveTypeBool * ptrPV19
- ValveTypeBool * ptrPV20
- ValveTypeBool * ptrPV21
- ValveTypeBool * ptrPV22
- bool posPD1
- bool posPD2
- bool posPD3

7.54.1 Constructor & Destructor Documentation

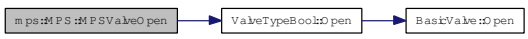
7.54.1.1 mps::MPS::MPS (SubsystemDirector * *_director*)

7.54.1.2 mps::MPS::~MPS (void)

7.54.2 Member Function Documentation

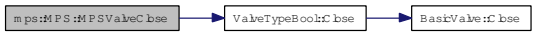
7.54.2.1 bool mps::MPS::MPSValveOpen (MPS_VALVE *vlv*)

Here is the call graph for this function:



7.54.2.2 bool mps::MPS::MPSValveClose (MPS_VALVE *vlv*)

Here is the call graph for this function:



7.54.2.3 double mps::MPS::MPSValveStatus (MPS_VALVE *vlv*)

Here is the call graph for this function:

7.54.2.4 void mps::MPS::OnPostStep (double *fSimT*, double *fDeltaT*, double *fMJD*) [virtual]

Finish calculations after the propagation of the state vector

Parameters:

fSimT current simulation time

fDeltaT Simulation time step length

fMJD current modified julian date

Reimplemented from AtlantisSubsystem.

Here is the call graph for this function:

7.54.2.5 void mps::MPS::OnSaveState (FILEHANDLE *scn*) const [virtual]

Save the state of the subsystem

Reimplemented from AtlantisSubsystem.

Here is the call graph for this function:

7.54.2.6 bool mps::MPS::OnParseLine (const char * *line*) [virtual]

Parse a line of the scenario file.

Warning:

Should get replaced by a block oriented scenario file format, like NASSP, making this function deprecated.

Reimplemented from AtlantisSubsystem.

7.54.3 Member Data Documentation

7.54.3.1 ValveTypeBool* mps::MPS::ptrPV1 [private]

7.54.3.2 ValveTypeBool* mps::MPS::ptrPV2 [private]

7.54.3.3 ValveTypeBool* mps::MPS::ptrPV3 [private]

7.54.3.4 ValveTypeBool* mps::MPS::ptrPV4 [private]

7.54.3.5 ValveTypeBool* mps::MPS::ptrPV5 [private]

7.54.3.6 ValveTypeBool* mps::MPS::ptrPV6 [private]

7.54.3.7 ValveTypeBool* mps::MPS::ptrPV7 [private]

7.54.3.8 ValveTypeBool* mps::MPS::ptrPV8 [private]

7.54.3.9 ValveTypeBool* mps::MPS::ptrPV9 [private]

7.54.3.10 ValveTypeBool* mps::MPS::ptrPV10 [private]

7.54.3.11 ValveTypeBool* mps::MPS::ptrPV11 [private]

7.54.3.12 ValveTypeBool* mps::MPS::ptrPV12 [private]

7.54.3.13 ValveTypeBool* mps::MPS::ptrPV13 [private]

7.54.3.14 ValveTypeBool* mps::MPS::ptrPV17 [private]

7.54.3.15 ValveTypeBool* mps::MPS::ptrPV18 [private]

7.54.3.16 ValveTypeBool* mps::MPS::ptrPV19 [private]

7.54.3.17 ValveTypeBool* mps::MPS::ptrPV20 [private]

7.54.3.18 ValveTypeBool* mps::MPS::ptrPV21 [private]

7.54.3.19 ValveTypeBool* mps::MPS::ptrPV22 [private]

7.54.3.20 bool mps::MPS::posPD1 [private]

7.54.3.21 bool mps::MPS::posPD2 [private]

7.54.3.22 bool mps::MPS::posPD3 [private]

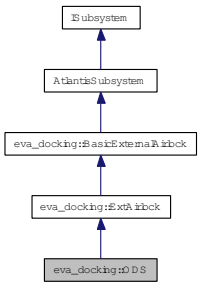
The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/MPS.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/MPS.cpp

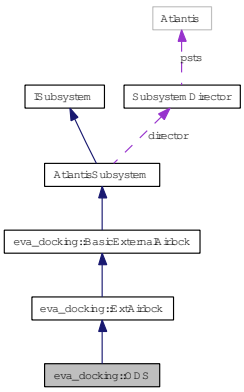
7.55 eva_docking::ODS Class Reference

`#include <ODS.h>`

Inheritance diagram for eva_docking::ODS:



Collaboration diagram for eva_docking::ODS:



Public Member Functions

- ODS (SubsystemDirector *pdirect, const string &_ident)
- virtual ~ODS ()
- virtual void AddMeshes (const VECTOR3 &ofs)
- virtual double GetSubsystemEmptyMass () const

Protected Attributes

- UINT midxODS
- double fRingState
- double fHookState
- double fLatchState
- double fVestPressure [2]

7.55.1 Constructor & Destructor Documentation

7.55.1.1 eva_docking::ODS::ODS (SubsystemDirector * *pdirect*, const string & *_ident*)

7.55.1.2 virtual eva_docking::ODS::~ODS () [virtual]

7.55.2 Member Function Documentation

7.55.2.1 virtual void eva_docking::ODS::AddMeshes (const VECTOR3 & *ofs*) [virtual]

Add new meshes to the visual.

Deprecated

Don't use, instead, use Realize() to tell the Atlantis class which meshes it should use for special slots.

Reimplemented from eva_docking::ExtAirlock.

7.55.2.2 virtual double eva_docking::ODS::GetSubsystemEmptyMass () const [virtual]

Get the dry mass of the subsystem, if this subsystem is optional and causes changes to the standard orbiter dry mass. For example RMS, external airlock/ODS, etc.

Returns:

The subsystem dry mass in kg.

Reimplemented from eva_docking::BasicExternalAirlock.

7.55.3 Member Data Documentation

7.55.3.1 UINT eva_docking::ODS::midxODS [protected]

7.55.3.2 double eva_docking::ODS::fRingState [protected]

7.55.3.3 double eva_docking::ODS::fHookState [protected]

7.55.3.4 double eva_docking::ODS::fLatchState [protected]

7.55.3.5 double eva_docking::ODS::fVestPressure[2] [protected]

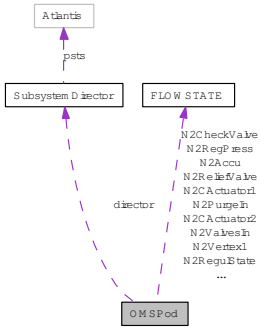
The documentation for this class was generated from the following file:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/eva_docking/ODS.h

7.56 OMSPod Class Reference

```
#include <OMSPod.h>
```

Collaboration diagram for OMSPod:



Public Member Functions

- OMSPod (SubsystemDirector *_director)
- void SetOMSThrusterPos (const VECTOR3 &rOMS)
- void CommandN2PressIsolValve (unsigned short usState)
- double SetN2TankFillState (double fFillPress, double fFillTemperature)
- double GetN2TankPressure (unsigned short sSensorID) const
- void PreStep (double fSimT, double fDeltaT)
- void Propagate (double fSimT, double fDeltaT)

Protected Member Functions

- void CalculateN2System (double fSimT, double fDeltaT)

Private Attributes

- SubsystemDirector * director
- FLOWSTATE N2Tank [2]
- double fN2TankMass [2]
- double fN2PressIsolVlv [2]
- FLOWSTATE N2PressIsolVlv [2]
- FLOWSTATE N2RegulState [2]
- FLOWSTATE N2ReliefValve [2]
- FLOWSTATE N2Vertex1 [2]
- FLOWSTATE N2CheckValve [2]
- FLOWSTATE N2RegPress [2]
- FLOWSTATE N2Accu [2]
- FLOWSTATE N2ValvesIn [2]
- FLOWSTATE N2CValve1 [2]

- FLOWSTATE N2CValve2 [2]

- FLOWSTATE N2CActuator1 [2]

- FLOWSTATE N2CActuator2 [2]

- FLOWSTATE N2PurgeIn [2]

7.56.1 Constructor & Destructor Documentation

7.56.1.1 OMSPod::OMSPod (SubsystemDirector * *_director*)

7.56.2 Member Function Documentation

7.56.2.1 void OMSPod::CalculateN2System (double *fSimT*, double *fDeltaT*) [protected]

7.56.2.2 void OMSPod::SetOMSThrusterPos (const VECTOR3 & *rOMS*)

7.56.2.3 void OMSPod::CommandN2PressIsolValve (unsigned short *usState*)

7.56.2.4 double OMSPod::SetN2TankFillState (double *fFillPress*, double *fFillTemperature*)

7.56.2.5 double OMSPod::GetN2TankPressure (unsigned short *sSensorID*) const

7.56.2.6 void OMSPod::PreStep (double *fSimT*, double *fDeltaT*)

7.56.2.7 void OMSPod::Propagate (double *fSimT*, double *fDeltaT*)

7.56.3 Member Data Documentation

7.56.3.1 SubsystemDirector* OMSPod::director [private]

7.56.3.2 FLOWSTATE OMSPod::N2Tank[2] [private]

7.56.3.3 double OMSPod::fN2TankMass[2] [private]

7.56.3.4 double OMSPod::fN2PressIsolVlv[2] [private]

7.56.3.5 FLOWSTATE OMSPod::N2PressIsolVlv[2] [private]

7.56.3.6 FLOWSTATE OMSPod::N2RegulState[2] [private]

7.56.3.7 FLOWSTATE OMSPod::N2ReliefValve[2] [private]

7.56.3.8 FLOWSTATE OMSPod::N2Vertex1[2] [private]

7.56.3.9 FLOWSTATE OMSPod::N2CheckValve[2] [private]

7.56.3.10 FLOWSTATE OMSPod::N2RegPress[2] [private]

7.56.3.11 FLOWSTATE OMSPod::N2Accu[2] [private]

7.56.3.12 FLOWSTATE OMSPod::N2ValvesIn[2] [private]

7.56.3.13 FLOWSTATE OMSPod::N2CValve1[2] [private]

7.56.3.14 FLOWSTATE OMSPod::N2CValve2[2] [private]

7.56.3.15 FLOWSTATE OMSPod::N2CActuator1[2] [private]

7.56.3.16 FLOWSTATE OMSPod::N2CActuator2[2] [private]

7.56.3.17 FLOWSTATE OMSPod::N2PurgeIn[2] [private]

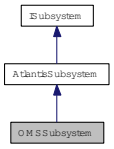
The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/OMSPod.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/OMSPod.cpp

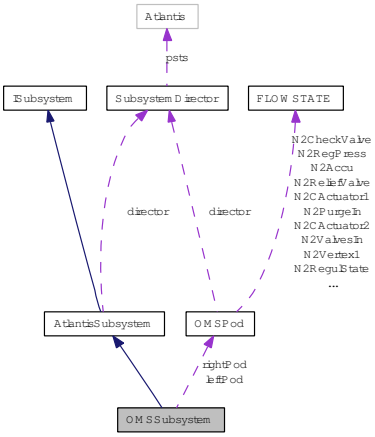
7.57 OMSSubsystem Class Reference

`#include <OMSSubsystem.h>`

Inheritance diagram for OMSSubsystem:



Collaboration diagram for OMSSubsystem:



Public Member Functions

- OMSSubsystem (SubsystemDirector *_director)
- virtual ~OMSSubsystem ()
- virtual void SetClassCaps (FILEHANDLE cfg)
- virtual bool ParseLine (const char *line)
- virtual void PreStep (double fSimT, double fDeltaT)
- virtual void Propagate (double fSimT, double fDeltaT)
- virtual void SaveState (FILEHANDLE scn)

Private Attributes

- OMSPod leftPod
- OMSPod rightPod

7.57.1 Constructor & Destructor Documentation

7.57.1.1 OMSSubsystem::OMSSubsystem (SubsystemDirector * *_director*)

7.57.1.2 OMSSubsystem::~OMSSubsystem () [virtual]

7.57.2 Member Function Documentation

7.57.2.1 void OMSSubsystem::SetClassCaps (FILEHANDLE *cfg*) [virtual]

7.57.2.2 bool OMSSubsystem::ParseLine (const char * *line*) [virtual]

7.57.2.3 void OMSSubsystem::PreStep (double *fSimT*, double *fDeltaT*) [virtual]

Here is the call graph for this function:

7.57.2.4 void OMSSubsystem::Propagate (double *fSimT*, double *fDeltaT*) [virtual]

Here is the call graph for this function:

7.57.2.5 void OMSSubsystem::SaveState (FILEHANDLE *scn*) [virtual]

7.57.3 Member Data Documentation

7.57.3.1 OMSPod OMSSubsystem::leftPod [private]

7.57.3.2 OMSPod OMSSubsystem::rightPod [private]

The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/OMSSubsystem.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/OMSSubsystem.cpp

7.58 PanelA4 Class Reference

`#include <PanelA4.h>`

Collaboration diagram for PanelA4:

Public Member Functions

- PanelA4 (Atlantis *_sts)
- void DefineAnimations ()
- void DefineVCAnimations (UINT vcidx)
- void RegisterVC ()
- void UpdateVC ()
- void UpdateMesh ()
- bool VCMouseEvent (int id, int event, VECTOR3 &p)
- bool VCRedrawEvent (int id, int event, SURFHANDLE surf)
- void Step (double t, double dt)
- bool ParseScenarioLine (char *line)
- void SaveState (FILEHANDLE scn)

Private Attributes

- Atlantis * sts
- UINT switch_state
- UINT old_switch_state
- double switch_timer
- short sTimerSeconds
- short sTimerMinutes
- short sMETSeconds
- short sMETMinutes
- short sMETHours
- short sMETDays
- UINT anim_VC_A4

Friends

- class Atlantis

7.58.1 Constructor & Destructor Documentation

7.58.1.1 PanelA4::PanelA4 (Atlantis * *_sts*)

7.58.2 Member Function Documentation

7.58.2.1 void PanelA4::DefineAnimations ()

7.58.2.2 void PanelA4::DefineVCAnimations (UINT *vcidx*)

7.58.2.3 void PanelA4::RegisterVC ()

7.58.2.4 void PanelA4::UpdateVC ()

7.58.2.5 void PanelA4::UpdateMesh ()

7.58.2.6 bool PanelA4::VCMouseEvent (int *id*, int *event*, VECTOR3 & *p*)

Here is the call graph for this function:

7.58.2.7 bool PanelA4::VCRedrawEvent (int *id*, int *event*, SURFHANDLE *surf*)

7.58.2.8 void PanelA4::Step (double *t*, double *dt*)

7.58.2.9 bool PanelA4::ParseScenarioLine (char * *line*)

7.58.2.10 void PanelA4::SaveState (FILEHANDLE *scn*)

7.58.3 Friends And Related Function Documentation

7.58.3.1 friend class Atlantis [friend]

7.58.4 Member Data Documentation

7.58.4.1 Atlantis* PanelA4::sts [private]

7.58.4.2 UINT PanelA4::switch_state [private]

7.58.4.3 UINT PanelA4::old_switch_state [private]

7.58.4.4 double PanelA4::switch_timer [private]

7.58.4.5 short PanelA4::sTimerSeconds [private]

7.58.4.6 short PanelA4::sTimerMinutes [private]

7.58.4.7 short PanelA4::sMETSeconds [private]

7.58.4.8 short PanelA4::sMETMinutes [private]

7.58.4.9 short PanelA4::sMETHours [private]

7.58.4.10 short PanelA4::sMETDays [private]

7.58.4.11 UINT PanelA4::anim_VC_A4 [private]

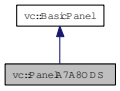
The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelA4.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelA4.cpp

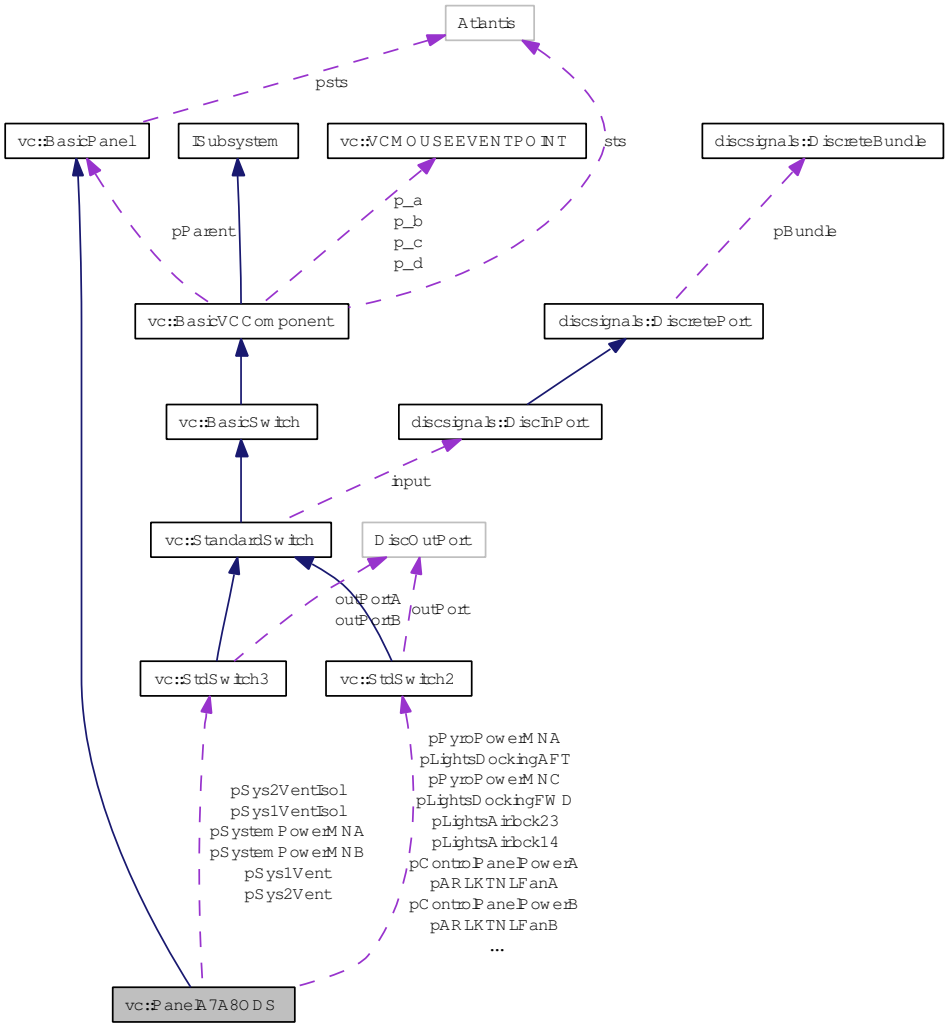
7.59 vc::PanelA7A8ODS Class Reference

`#include <PanelA7A8ODS.h>`

Inheritance diagram for vc::PanelA7A8ODS:



Collaboration diagram for vc::PanelA7A8ODS:



Public Member Functions

- PanelA7A8ODS (Atlantis *_sts)
- virtual ~PanelA7A8ODS ()
- virtual void AddMeshes (const VECTOR3 &ofs)
- virtual void DefineVCAnimations (UINT vcidx)
- virtual void DefineVC ()
- virtual UINT GetVCMeshIndex () const
- virtual void Realize ()
- virtual void RegisterVC ()

Private Attributes

- StdSwitch3 * pSystemPowerMNA
- StdSwitch3 * pSystemPowerMNB
- StdSwitch2 * pPyroPowerMNA
- StdSwitch2 * pPyroPowerMNC
- StdSwitch3 * pSys1VentIsol
- StdSwitch3 * pSys1Vent
- StdSwitch3 * pSys2VentIsol
- StdSwitch3 * pSys2Vent
- StdSwitch2 * pPSUPowerMNA
- StdSwitch2 * pPSUPowerMNB
- StdSwitch2 * pLightsAirlock14
- StdSwitch2 * pLightsAirlock23
- StdSwitch2 * pLightsDockingFWD
- StdSwitch2 * pLightsDockingAFT
- StdSwitch2 * pARLKTNLFanA
- StdSwitch2 * pARLKTNLFanB
- StdSwitch2 * pLightsCLVestPORT
- StdSwitch2 * pLightsCLVestSTBD
- StdSwitch2 * pControlPanelPowerA
- StdSwitch2 * pControlPanelPowerB
- StdSwitch2 * pControlPanelPowerC
- StdSwitch2 * pPyrosAp
- StdSwitch2 * pPyrosBp
- StdSwitch2 * pPyrosCp
- UINT midx_odspanel

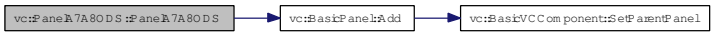
7.59.1 Detailed Description

Example implementation of a BasicPanel derived panel

7.59.2 Constructor & Destructor Documentation

7.59.2.1 vc::PanelA7A8ODS::PanelA7A8ODS (Atlantis * *_sts*)

Here is the call graph for this function:



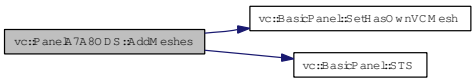
7.59.2.2 vc::PanelA7A8ODS::~PanelA7A8ODS () [virtual]

7.59.3 Member Function Documentation

7.59.3.1 void vc::PanelA7A8ODS::AddMeshes (const VECTOR3 & *ofs*) [virtual]

Reimplemented from vc::BasicPanel.

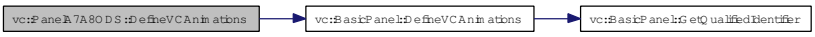
Here is the call graph for this function:



7.59.3.2 void vc::PanelA7A8ODS::DefineVCAnimations (UINT *vcidx*) [virtual]

Reimplemented from vc::BasicPanel.

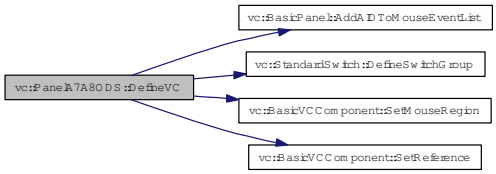
Here is the call graph for this function:



7.59.3.3 void vc::PanelA7A8ODS::DefineVC () [virtual]

Implements vc::BasicPanel.

Here is the call graph for this function:



7.59.3.4 UINT vc::PanelA7A8ODS::GetVCMeshIndex () const [virtual]

Reimplemented from vc::BasicPanel.

7.59.3.5 void vc::PanelA7A8ODS::Realize () [virtual]

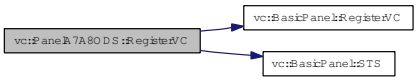
Reimplemented from vc::BasicPanel.

Here is the call graph for this function:

7.59.3.6 void vc::PanelA7A8ODS::RegisterVC () [virtual]

Reimplemented from vc::BasicPanel.

Here is the call graph for this function:



7.59.4 Member Data Documentation

7.59.4.1 StdSwitch3* vc::PanelA7A8ODS::pSystemPowerMNA [private]

7.59.4.2 StdSwitch3* vc::PanelA7A8ODS::pSystemPowerMNB [private]

7.59.4.3 StdSwitch2* vc::PanelA7A8ODS::pPyroPowerMNA [private]

7.59.4.4 StdSwitch2* vc::PanelA7A8ODS::pPyroPowerMNC [private]

7.59.4.5 StdSwitch3* vc::PanelA7A8ODS::pSys1VentIsol [private]

7.59.4.6 StdSwitch3* vc::PanelA7A8ODS::pSys1Vent [private]

7.59.4.7 StdSwitch3* vc::PanelA7A8ODS::pSys2VentIsol [private]

7.59.4.8 StdSwitch3* vc::PanelA7A8ODS::pSys2Vent [private]

7.59.4.9 StdSwitch2* vc::PanelA7A8ODS::pPSUPowerMNA [private]

7.59.4.10 StdSwitch2* vc::PanelA7A8ODS::pPSUPowerMNB [private]

7.59.4.11 StdSwitch2* vc::PanelA7A8ODS::pLightsAirlock14 [private]

7.59.4.12 StdSwitch2* vc::PanelA7A8ODS::pLightsAirlock23 [private]

7.59.4.13 StdSwitch2* vc::PanelA7A8ODS::pLightsDockingFWD [private]

7.59.4.14 StdSwitch2* vc::PanelA7A8ODS::pLightsDockingAFT [private]

7.59.4.15 StdSwitch2* vc::PanelA7A8ODS::pARLKTNLFanA [private]

7.59.4.16 StdSwitch2* vc::PanelA7A8ODS::pARLKTNLFanB [private]

7.59.4.17 StdSwitch2* vc::PanelA7A8ODS::pLightsCLVestPORT [private]

7.59.4.18 StdSwitch2* vc::PanelA7A8ODS::pLightsCLVestSTBD [private]

7.59.4.19 StdSwitch2* vc::PanelA7A8ODS::pControlPanelPowerA [private]

7.59.4.20 StdSwitch2* vc::PanelA7A8ODS::pControlPanelPowerB [private]

7.59.4.21 StdSwitch2* vc::PanelA7A8ODS::pControlPanelPowerC [private]

7.59.4.22 StdSwitch2* vc::PanelA7A8ODS::pPyrosAp [private]

7.59.4.23 StdSwitch2* vc::PanelA7A8ODS::pPyrosBp [private]

7.59.4.24 StdSwitch2* vc::PanelA7A8ODS::pPyrosCp [private]

7.59.4.25 UINT vc::PanelA7A8ODS::midx_odspanel [private]

The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelA7A8ODS.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelA7A8ODS.cpp

7.60 PanelA8 Class Reference

`#include <PanelA8.h>`

Collaboration diagram for PanelA8:

Public Member Functions

- PanelA8 (Atlantis *_sts)
- void DefineVCAnimations (UINT vcidx)
- void RegisterVC ()
- void UpdateVC ()
- void UpdateMesh ()
- bool VCMouseEvent (int id, int event, VECTOR3 &p)
- bool VCRedrawEvent (int id, int event, SURFHANDLE surf)
- void Step (double t, double dt)
- bool ParseScenarioLine (char *line)
- void SaveState (FILEHANDLE scn)

Private Member Functions

- bool VCDrawTalkback (SURFHANDLE surf, int idx, int label)

Private Attributes

- Atlantis * sts
- UINT anim_VC_A8 [20]
- int switch_state [20]
- int tkbk_state [20]

Friends

- class Atlantis
- class CRT

7.60.1 Constructor & Destructor Documentation

7.60.1.1 PanelA8::PanelA8 (Atlantis * *_sts*)

7.60.2 Member Function Documentation

7.60.2.1 void PanelA8::DefineVCAnimations (UINT *vcidx*)

7.60.2.2 void PanelA8::RegisterVC ()

7.60.2.3 void PanelA8::UpdateVC ()

7.60.2.4 void PanelA8::UpdateMesh ()

7.60.2.5 bool PanelA8::VCMouseEvent (int *id*, int *event*, VECTOR3 & *p*)

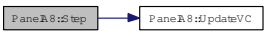
Here is the call graph for this function:

7.60.2.6 bool PanelA8::VCRedrawEvent (int *id*, int *event*, SURFHANDLE *surf*)

Here is the call graph for this function:

7.60.2.7 void PanelA8::Step (double *t*, double *dt*)

Here is the call graph for this function:



7.60.2.8 bool PanelA8::ParseScenarioLine (char * *line*)

7.60.2.9 void PanelA8::SaveState (FILEHANDLE *scn*)

7.60.2.10 bool PanelA8::VCDrawTalkback (SURFHANDLE *surf*, int *idx*, int *label*) [private]

7.60.3 Friends And Related Function Documentation

7.60.3.1 friend class Atlantis [friend]

7.60.3.2 friend class CRT [friend]

7.60.4 Member Data Documentation

7.60.4.1 Atlantis* PanelA8::sts [private]

7.60.4.2 UINT PanelA8::anim_VC_A8[20] [private]

7.60.4.3 int PanelA8::switch_state[20] [private]

7.60.4.4 int PanelA8::tkbk_state[20] [private]

The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelA8.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelA8.cpp

7.61 PanelC2 Class Reference

`#include <PanelC2.h>`

Collaboration diagram for PanelC2:

Public Member Functions

- PanelC2 (Atlantis *_sts)
- ~PanelC2 ()
- void DefineAnimations ()
- void DefineVCAnimations (UINT vcidx)
- void RegisterVC ()
- void UpdateVC ()
- void CreateVisual (VISHANDLE hVisual)
- void UpdateMesh ()
- bool VCMouseEvent (int id, int event, VECTOR3 &p)
- bool VCRedrawEvent (int id, int event, SURFHANDLE surf)
- void Step (double t, double dt)
- bool ParseScenarioLine (char *line)
- void SaveState (FILEHANDLE scn)

Private Attributes

- Atlantis * sts
- UINT anim_VC_C2 [12]
- UINT anim_VC_C2Wheel [4]
- SURFHANDLE wheel_tex [4]
- SURFHANDLE digit_texture
- double tgtwheel_state [4]
- short wheelnumber [4]
- short oldwheelnumber [4]
- UINT switch_state [12]
- UINT old_switch_state [12]
- double switch_timer [12]
- double wheelState [4]
- UINT EventTimerWheelRevs [4]
- UINT EventTimerMode
- UINT EventTimerControl

Friends

- class Atlantis
- class CRT

7.61.1 Constructor & Destructor Documentation

7.61.1.1 PanelC2::PanelC2 (Atlantis * *_sts*)

7.61.1.2 PanelC2::~PanelC2 ()

7.61.2 Member Function Documentation

7.61.2.1 void PanelC2::DefineAnimations ()

7.61.2.2 void PanelC2::DefineVCAnimations (UINT *vcidx*)

7.61.2.3 void PanelC2::RegisterVC ()

7.61.2.4 void PanelC2::UpdateVC ()

7.61.2.5 void PanelC2::CreateVisual (VISHANDLE *hVisual*)

7.61.2.6 void PanelC2::UpdateMesh ()

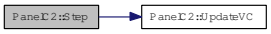
7.61.2.7 bool PanelC2::VCMouseEvent (int *id*, int *event*, VECTOR3 & *p*)

Here is the call graph for this function:

7.61.2.8 bool PanelC2::VCRedrawEvent (int *id*, int *event*, SURFHANDLE *surf*)

7.61.2.9 void PanelC2::Step (double *t*, double *dt*)

Here is the call graph for this function:



7.61.2.10 bool PanelC2::ParseScenarioLine (char * *line*)

7.61.2.11 void PanelC2::SaveState (FILEHANDLE *scn*)

7.61.3 Friends And Related Function Documentation

7.61.3.1 friend class Atlantis [friend]

7.61.3.2 friend class CRT [friend]

7.61.4 Member Data Documentation

7.61.4.1 Atlantis* PanelC2::sts [private]

7.61.4.2 UINT PanelC2::anim_VC_C2[12] [private]

7.61.4.3 UINT PanelC2::anim_VC_C2Wheel[4] [private]

7.61.4.4 SURFHANDLE PanelC2::wheel_tex[4] [private]

7.61.4.5 SURFHANDLE PanelC2::digit_texture [private]

7.61.4.6 double PanelC2::tgtwheel_state[4] [private]

7.61.4.7 short PanelC2::wheelnumber[4] [private]

7.61.4.8 short PanelC2::oldwheelnumber[4] [private]

7.61.4.9 UINT PanelC2::switch_state[12] [private]

7.61.4.10 UINT PanelC2::old_switch_state[12] [private]

7.61.4.11 double PanelC2::switch_timer[12] [private]

7.61.4.12 double PanelC2::wheelState[4] [private]

7.61.4.13 UINT PanelC2::EventTimerWheelRevs[4] [private]

7.61.4.14 UINT PanelC2::EventTimerMode [private]

7.61.4.15 UINT PanelC2::EventTimerControl [private]

The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelC2.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelC2.cpp

7.62 PanelC3 Class Reference

`#include <PanelC3.h>`

Collaboration diagram for PanelC3:

Public Member Functions

- PanelC3 (Atlantis *_sts)
- void DefineAnimations ()
- void DefineVCAnimations (UINT vcidx)
- void RegisterVC ()
- void UpdateVC ()
- void UpdateMesh ()
- bool VCMouseEvent (int id, int event, VECTOR3 &p)
- bool VCRedrawEvent (int id, int event, SURFHANDLE surf)
- bool CheckProbesDeployed ()
- bool CheckOMSArm (int nEng)
- void Step (double t, double dt)
- bool ParseScenarioLine (char *line)
- void SaveState (FILEHANDLE scn)

Private Member Functions

- void EngControl (int eng)
- void AirDataProbeControl ()

Private Attributes

- Atlantis * sts
- int Air_Data_Probe [2]
- int Air_Data_Stow [2]
- int OMS_Eng [2]
- UINT anim_VC_C3 [37]
- AnimState AirDataProbe [2]

Friends

- class Atlantis

7.62.1 Constructor & Destructor Documentation

7.62.1.1 PanelC3::PanelC3 (Atlantis * *_sts*)

7.62.2 Member Function Documentation

7.62.2.1 void PanelC3::DefineAnimations ()

7.62.2.2 void PanelC3::DefineVCAnimations (UINT *vcidx*)

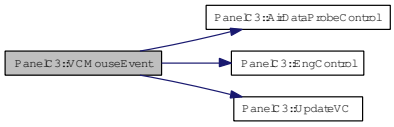
7.62.2.3 void PanelC3::RegisterVC ()

7.62.2.4 void PanelC3::UpdateVC ()

7.62.2.5 void PanelC3::UpdateMesh ()

7.62.2.6 bool PanelC3::VCMouseEvent (int *id*, int *event*, VECTOR3 & *p*)

Here is the call graph for this function:



7.62.2.7 bool PanelC3::VCRedrawEvent (int *id*, int *event*, SURFHANDLE *surf*)

7.62.2.8 bool PanelC3::CheckProbesDeployed ()

7.62.2.9 bool PanelC3::CheckOMSArm (int *nEng*)

7.62.2.10 void PanelC3::Step (double *t*, double *dt*)

7.62.2.11 bool PanelC3::ParseScenarioLine (char * *line*)

Here is the call graph for this function:

7.62.2.12 void PanelC3::SaveState (FILEHANDLE *scn*)

7.62.2.13 void PanelC3::EngControl (int *eng*) [private]

7.62.2.14 void PanelC3::AirDataProbeControl () [private]

7.62.3 Friends And Related Function Documentation

7.62.3.1 friend class Atlantis [friend]

7.62.4 Member Data Documentation

7.62.4.1 Atlantis* PanelC3::sts [private]

7.62.4.2 int PanelC3::Air_Data_Probe[2] [private]

7.62.4.3 int PanelC3::Air_Data_Stow[2] [private]

7.62.4.4 int PanelC3::OMS_Eng[2] [private]

7.62.4.5 UINT PanelC3::anim_VC_C3[37] [private]

7.62.4.6 AnimState PanelC3::AirDataProbe[2] [private]

The documentation for this class was generated from the following files:

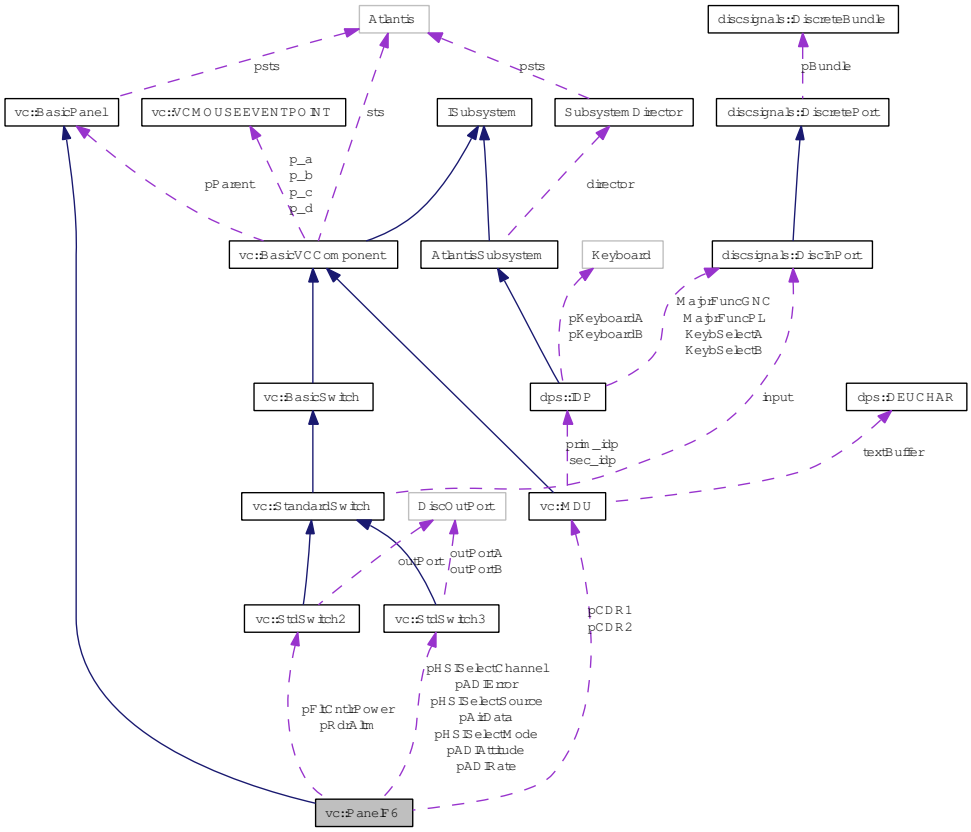
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelC3.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelC3.cpp

7.63 vc::PanelF6 Class Reference

`#include <PanelF6.h>`

Inheritance diagram for vc::PanelF6:

Collaboration diagram for vc::PanelF6:



Public Member Functions

- PanelF6 (Atlantis *_sts)
- virtual ~PanelF6 ()
- virtual void DefineVC ()
- virtual void RegisterVC ()

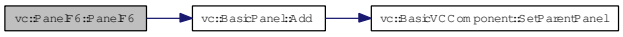
Private Attributes

- MDU * pCDR1
- MDU * pCDR2
- StdSwitch3 * pHSISelectMode
- StdSwitch3 * pHSISelectSource
- StdSwitch3 * pHSISelectChannel
- StdSwitch2 * pRdrAltm
- StdSwitch2 * pFltCntlrPower
- StdSwitch3 * pADIAttitude
- StdSwitch3 * pADIError
- StdSwitch3 * pADIRate
- StdSwitch3 * pAirData

7.63.1 Constructor & Destructor Documentation

7.63.1.1 vc::PanelF6::PanelF6 (Atlantis * *_sts*)

Here is the call graph for this function:



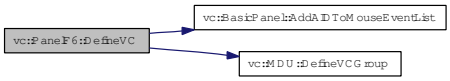
7.63.1.2 vc::PanelF6::~PanelF6 () [virtual]

7.63.2 Member Function Documentation

7.63.2.1 void vc::PanelF6::DefineVC () [virtual]

Implements vc::BasicPanel.

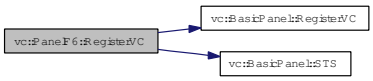
Here is the call graph for this function:



7.63.2.2 void vc::PanelF6::RegisterVC () [virtual]

Reimplemented from vc::BasicPanel.

Here is the call graph for this function:



7.63.3 Member Data Documentation

7.63.3.1 MDU* vc::PanelF6::pCDR1 [private]

7.63.3.2 MDU* vc::PanelF6::pCDR2 [private]

7.63.3.3 StdSwitch3* vc::PanelF6::pHSISelectMode [private]

7.63.3.4 StdSwitch3* vc::PanelF6::pHSISelectSource [private]

7.63.3.5 StdSwitch3* vc::PanelF6::pHSISelectChannel [private]

7.63.3.6 StdSwitch2* vc::PanelF6::pRdrAltm [private]

7.63.3.7 StdSwitch2* vc::PanelF6::pFltCntlrPower [private]

7.63.3.8 StdSwitch3* vc::PanelF6::pADIAttitude [private]

7.63.3.9 StdSwitch3* vc::PanelF6::pADIError [private]

7.63.3.10 StdSwitch3* vc::PanelF6::pADIRate [private]

7.63.3.11 StdSwitch3* vc::PanelF6::pAirData [private]

The documentation for this class was generated from the following files:

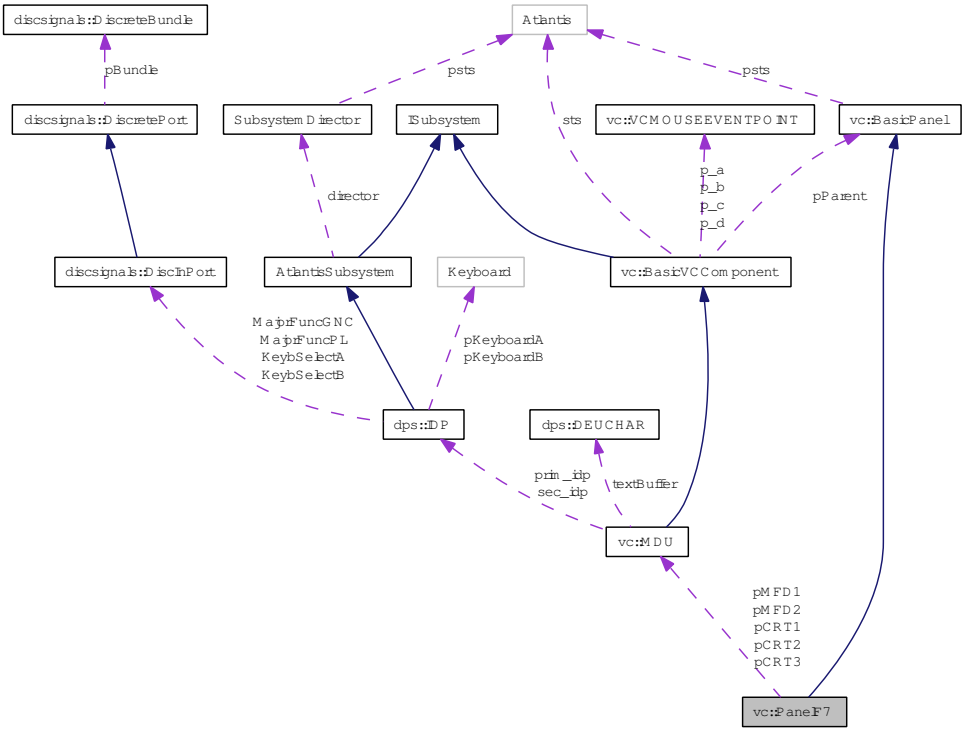
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelF6.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelF6.cpp

7.64 vc::PanelF7 Class Reference

`#include <PanelF7.h>`

Inheritance diagram for vc::PanelF7:

Collaboration diagram for vc::PanelF7:



Public Member Functions

- PanelF7 (Atlantis *_sts)
- virtual ~PanelF7 ()
- virtual void DefineVC ()
- virtual void RegisterVC ()

Private Attributes

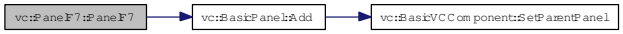
- MDU * pCRT1

- MDU * pCRT2
- MDU * pCRT3
- MDU * pMFD1
- MDU * pMFD2

7.64.1 Constructor & Destructor Documentation

7.64.1.1 PanelF7::PanelF7 (Atlantis * *_sts*)

Here is the call graph for this function:



7.64.1.2 PanelF7::~PanelF7 () [virtual]

7.64.2 Member Function Documentation

7.64.2.1 void PanelF7::DefineVC () [virtual]

Implements vc::BasicPanel.

7.64.2.2 void PanelF7::RegisterVC () [virtual]

Reimplemented from vc::BasicPanel.

7.64.3 Member Data Documentation

7.64.3.1 MDU* vc::PanelF7::pCRT1 [private]

7.64.3.2 MDU* vc::PanelF7::pCRT2 [private]

7.64.3.3 MDU* vc::PanelF7::pCRT3 [private]

7.64.3.4 MDU* vc::PanelF7::pMFD1 [private]

7.64.3.5 MDU* vc::PanelF7::pMFD2 [private]

The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelF7.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelF7.cpp

7.65 PanelF7 Class Reference

`#include <PanelF7.h>`

Collaboration diagram for PanelF7:

Public Member Functions

- PanelF7 (Atlantis *_sts)
- void DefineAnimations ()
- void DefineVCAnimations (UINT vcidx)
- void RegisterVC ()
- void UpdateVC ()
- void UpdateMesh ()
- bool VCMouseEvent (int id, int event, VECTOR3 &p)
- bool VCRedrawEvent (int id, int event, SURFHANDLE surf)
- void Step (double t, double dt)
- bool ParseScenarioLine (char *line)
- void SaveState (FILEHANDLE scn)

Private Attributes

- Atlantis * sts
- short sTimerSeconds
- short sTimerMinutes

Friends

- class Atlantis

7.65.1 Constructor & Destructor Documentation

7.65.1.1 PanelF7::PanelF7 (Atlantis * *_sts*)

7.65.2 Member Function Documentation

7.65.2.1 void PanelF7::DefineAnimations ()

7.65.2.2 void PanelF7::DefineVCAnimations (UINT *vcidx*)

7.65.2.3 void PanelF7::RegisterVC ()

7.65.2.4 void PanelF7::UpdateVC ()

7.65.2.5 void PanelF7::UpdateMesh ()

7.65.2.6 bool PanelF7::VCMouseEvent (int *id*, int *event*, VECTOR3 & *p*)

7.65.2.7 bool PanelF7::VCRedrawEvent (int *id*, int *event*, SURFHANDLE *surf*)

7.65.2.8 void PanelF7::Step (double *t*, double *dt*)

7.65.2.9 bool PanelF7::ParseScenarioLine (char * *line*)

7.65.2.10 void PanelF7::SaveState (FILEHANDLE *scn*)

7.65.3 Friends And Related Function Documentation

7.65.3.1 friend class Atlantis `[friend]`

7.65.4 Member Data Documentation

7.65.4.1 Atlantis* PanelF7::sts `[private]`

7.65.4.2 short PanelF7::sTimerSeconds `[private]`

7.65.4.3 short PanelF7::sTimerMinutes `[private]`

The documentation for this class was generated from the following files:

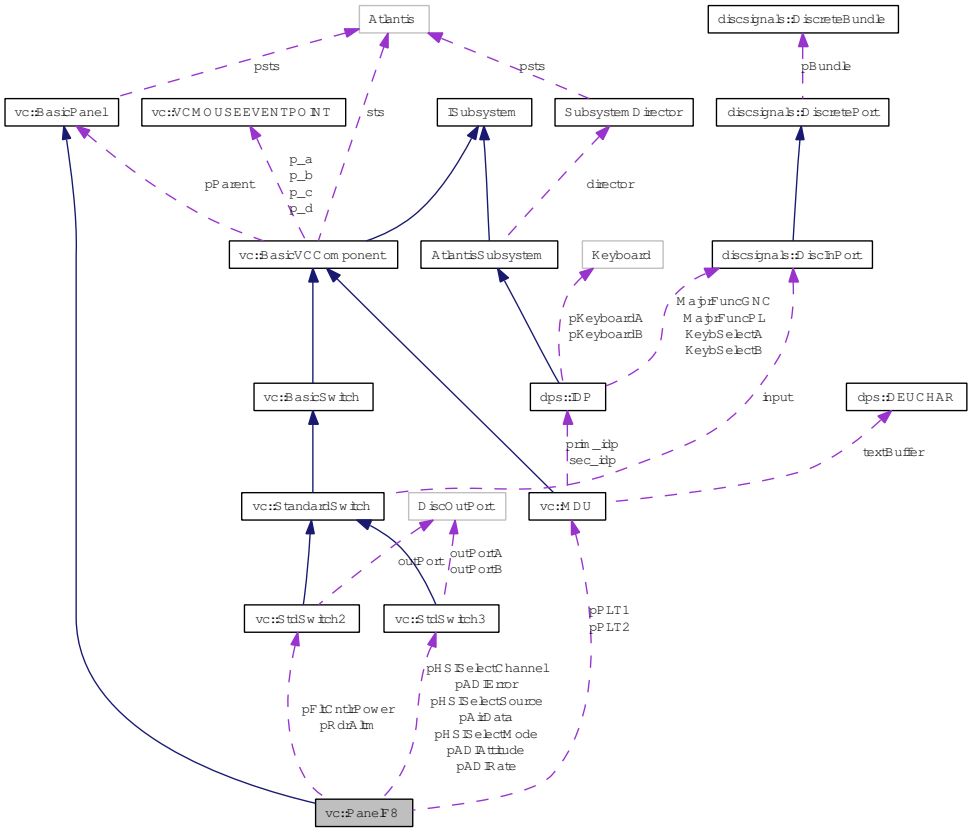
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelF7.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelF7.cpp

7.66 vc::PanelF8 Class Reference

`#include <PanelF8.h>`

Inheritance diagram for vc::PanelF8:

Collaboration diagram for vc::PanelF8:



Public Member Functions

- PanelF8 (Atlantis *_sts)
- virtual ~PanelF8 ()
- virtual void DefineVC ()
- virtual void RegisterVC ()

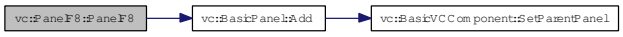
Private Attributes

- MDU * pPLT1
- MDU * pPLT2
- StdSwitch3 * pHSISelectMode
- StdSwitch3 * pHSISelectSource
- StdSwitch3 * pHSISelectChannel
- StdSwitch2 * pRdrAltm
- StdSwitch2 * pFltCntlrPower
- StdSwitch3 * pADIAttitude
- StdSwitch3 * pADIError
- StdSwitch3 * pADIRate
- StdSwitch3 * pAirData

7.66.1 Constructor & Destructor Documentation

7.66.1.1 vc::PanelF8::PanelF8 (Atlantis * *_sts*)

Here is the call graph for this function:



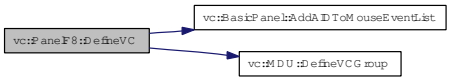
7.66.1.2 vc::PanelF8::~PanelF8 () [virtual]

7.66.2 Member Function Documentation

7.66.2.1 void vc::PanelF8::DefineVC () [virtual]

Implements vc::BasicPanel.

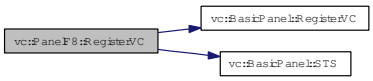
Here is the call graph for this function:



7.66.2.2 void vc::PanelF8::RegisterVC () [virtual]

Reimplemented from vc::BasicPanel.

Here is the call graph for this function:



7.66.3 Member Data Documentation

7.66.3.1 MDU* vc::PanelF8::pPLT1 [private]

7.66.3.2 MDU* vc::PanelF8::pPLT2 [private]

7.66.3.3 StdSwitch3* vc::PanelF8::pHSISelectMode [private]

7.66.3.4 StdSwitch3* vc::PanelF8::pHSISelectSource [private]

7.66.3.5 StdSwitch3* vc::PanelF8::pHSISelectChannel [private]

7.66.3.6 StdSwitch2* vc::PanelF8::pRdrAltm [private]

7.66.3.7 StdSwitch2* vc::PanelF8::pFltCntlrPower [private]

7.66.3.8 StdSwitch3* vc::PanelF8::pADIAttitude [private]

7.66.3.9 StdSwitch3* vc::PanelF8::pADIError [private]

7.66.3.10 StdSwitch3* vc::PanelF8::pADIRate [private]

7.66.3.11 StdSwitch3* vc::PanelF8::pAirData [private]

The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelF8.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelF8.cpp

7.67 vc::PanelGroup Class Reference

```
#include <PanelGroup.h>
```

Public Member Functions

- PanelGroup ()
- virtual ~PanelGroup ()
- virtual bool AddPanel (BasicPanel *pPanel)
- void RegisterVC ()
- void DefineVCAnimations (UINT vcidx)
- void DefineVC ()
- void Realize ()
- bool OnVCMouseEvent (int id, int _event, VECTOR3 &p)
- bool OnVCRedrawEvent (int id, int _event, SURFHANDLE surf)
- void OnPreStep (double fSimT, double fDeltaT, double fMJD)
- void OnPostStep (double fSimT, double fDeltaT, double fMJD)
- void OnPropagate (double fSimT, double fDeltaT, double fMJD)
- bool OnParseLine (char *line)
- void OnSaveState (FILEHANDLE scn)
- bool EnableCoordinateDisplayMode ()
- bool DisableCoordinateDisplayMode ()
- bool ToggleCoordinateDisplayMode ()

Private Attributes

- vector< BasicPanel * > panels

7.67.1 Constructor & Destructor Documentation

7.67.1.1 vc::PanelGroup::PanelGroup ()

7.67.1.2 vc::PanelGroup::~PanelGroup () [virtual]

7.67.2 Member Function Documentation

7.67.2.1 bool vc::PanelGroup::AddPanel (BasicPanel * *pPanel*) [virtual]

7.67.2.2 void vc::PanelGroup::RegisterVC ()

7.67.2.3 void vc::PanelGroup::DefineVCAnimations (UINT *vcidx*)

7.67.2.4 void vc::PanelGroup::DefineVC ()

7.67.2.5 void vc::PanelGroup::Realize ()

7.67.2.6 bool vc::PanelGroup::OnVCMouseEvent (int *id*, int *_event*, VECTOR3 & *p*)

7.67.2.7 bool vc::PanelGroup::OnVCRedrawEvent (int *id*, int *_event*, SURFHANDLE *surf*)

7.67.2.8 void vc::PanelGroup::OnPreStep (double *fSimT*, double *fDeltaT*, double *fMJD*)

7.67.2.9 void vc::PanelGroup::OnPostStep (double *fSimT*, double *fDeltaT*, double *fMJD*)

7.67.2.10 void vc::PanelGroup::OnPropagate (double *fSimT*, double *fDeltaT*, double *fMJD*)

7.67.2.11 bool vc::PanelGroup::OnParseLine (char * *line*)

7.67.2.12 void vc::PanelGroup::OnSaveState (FILEHANDLE *scn*)

7.67.2.13 bool vc::PanelGroup::EnableCoordinateDisplayMode ()

7.67.2.14 bool vc::PanelGroup::DisableCoordinateDisplayMode ()

7.67.2.15 bool vc::PanelGroup::ToggleCoordinateDisplayMode ()

7.67.3 Member Data Documentation

7.67.3.1 vector<BasicPanel*> vc::PanelGroup::panels [private]

The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelGroup.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelGroup.cpp

7.68 PanelO3 Class Reference

`#include <PanelO3.h>`

Collaboration diagram for PanelO3:

Public Member Functions

- PanelO3 (Atlantis *_sts)
- void DefineAnimations ()
- void DefineVCAnimations (UINT vcidx)
- void RegisterVC ()
- void UpdateVC ()
- void UpdateMesh ()
- bool VCMouseEvent (int id, int event, VECTOR3 &p)
- bool VCRedrawEvent (int id, int event, SURFHANDLE surf)
- void Step (double t, double dt)
- bool ParseScenarioLine (char *line)
- void SaveState (FILEHANDLE scn)

Private Attributes

- Atlantis * sts
- UINT anim_VC_O3 [NUM_O3_SWITCHES]
- UINT switch_state [NUM_O3_SWITCHES]
- UINT old_switch_state [NUM_O3_SWITCHES]
- double switch_timer [NUM_O3_SWITCHES]
- short sTimerSeconds
- short sTimerMinutes
- short sMETSeconds
- short sMETMinutes
- short sMETHours
- short sMETDays
- short sOMSFuel [3]

Friends

- class Atlantis

7.68.1 Constructor & Destructor Documentation

7.68.1.1 PanelO3::PanelO3 (Atlantis * *_sts*)

7.68.2 Member Function Documentation

7.68.2.1 void PanelO3::DefineAnimations ()

7.68.2.2 void PanelO3::DefineVCAnimations (UINT *vcidx*)

7.68.2.3 void PanelO3::RegisterVC ()

7.68.2.4 void PanelO3::UpdateVC ()

7.68.2.5 void PanelO3::UpdateMesh ()

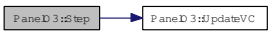
7.68.2.6 bool PanelO3::VCMouseEvent (int *id*, int *event*, VECTOR3 & *p*)

Here is the call graph for this function:

7.68.2.7 bool PanelO3::VCRedrawEvent (int *id*, int *event*, SURFHANDLE *surf*)

7.68.2.8 void PanelO3::Step (double *t*, double *dt*)

Here is the call graph for this function:



7.68.2.9 bool PanelO3::ParseScenarioLine (char * *line*)

7.68.2.10 void PanelO3::SaveState (FILEHANDLE *scn*)

7.68.3 Friends And Related Function Documentation

7.68.3.1 friend class Atlantis [friend]

7.68.4 Member Data Documentation

7.68.4.1 Atlantis* PanelO3::sts [private]

7.68.4.2 UINT PanelO3::anim_VC_O3[NUM_O3_SWITCHES] [private]

7.68.4.3 UINT PanelO3::switch_state[NUM_O3_SWITCHES] [private]

7.68.4.4 UINT PanelO3::old_switch_state[NUM_O3_SWITCHES] [private]

7.68.4.5 double PanelO3::switch_timer[NUM_O3_SWITCHES] [private]

7.68.4.6 short PanelO3::sTimerSeconds [private]

7.68.4.7 short PanelO3::sTimerMinutes [private]

7.68.4.8 short PanelO3::sMETSeconds [private]

7.68.4.9 short PanelO3::sMETMinutes [private]

7.68.4.10 short PanelO3::sMETHours [private]

7.68.4.11 short PanelO3::sMETDays [private]

7.68.4.12 short PanelO3::sOMSFuel[3] [private]

The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelO3.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelO3.cpp

7.69 PanelR2 Class Reference

```
#include <PanelR2.h>
```

Collaboration diagram for PanelR2:

Public Member Functions

- PanelR2 (Atlantis *_sts)
- void DefineAnimations ()
- void DefineVCAnimations (UINT vcidx)
- void RegisterVC ()
- void UpdateVC ()
- void UpdateMesh ()
- bool VCMouseEvent (int id, int event, VECTOR3 &p)
- bool VCRedrawEvent (int id, int event, SURFHANDLE surf)
- void Step (double t, double dt)
- bool ParseScenarioLine (char *line)
- void SaveState (FILEHANDLE scn)

Private Member Functions

- bool VCDrawTalkback (SURFHANDLE surf, int idx, int label)
- void CheckMPSArmed (int eng)
- void CheckAPUReadytoStart ()
- void CheckAPUShutdown ()
- void SetETUmbDoorAction (AnimState::Action action, int door)

Private Attributes

- Atlantis * sts
- int BOILER_N2_SUPPLY [3]
- int BOILER_CNTLR [3]
- int BOILER_CNTLRHTR [3]
- int APU_FUEL_TK_VLV [3]
- int APU_CNTLR_PWR [3]
- int HYD_CIRC_PUMP [3]
- int APU_AUTO_SHUT_DOWN
- int APU_SPEED_SELECT [3]
- int APU_CONTROL [3]
- int HYD_MAIN_PUMP_PRESS [3]
- int MPS_ENGINE_POWER [2][3]
- int HE_ISOLATION_A [3]

- int HE_ISOLATION_B [3]
- int CENTERLINE_LATCH
- int LEFT_DOOR
- int LEFT_LATCH
- int RIGHT_DOOR
- int RIGHT_LATCH
- int APU_STATE [3]
- int MPS_STATE [3]
- bool APU_READY [3]
- double APU_FUEL [3]
- int Hydraulic_Press [3]
- double APU_Speed [3]
- int Fuel_Press [3]
- int tkbk_state [8]
- bool bHydraulicPressure
- UINT anim_VC_R2 [56]
- AnimState LETUmbDoorStatus
- AnimState RETUmbDoorStatus
- AnimState CenterlineLatches
- AnimState LDoorLatches
- AnimState RDoorLatches

Friends

- class Atlantis
- class CRT

7.69.1 Constructor & Destructor Documentation

7.69.1.1 PanelR2::PanelR2 (Atlantis * *_sts*)

7.69.2 Member Function Documentation

7.69.2.1 void PanelR2::DefineAnimations ()

7.69.2.2 void PanelR2::DefineVCAnimations (UINT *vcidx*)

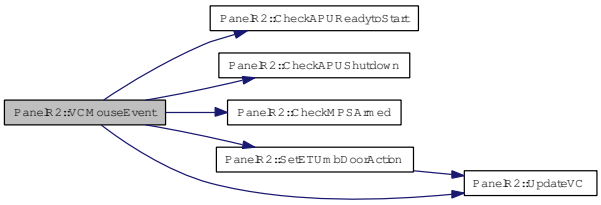
7.69.2.3 void PanelR2::RegisterVC ()

7.69.2.4 void PanelR2::UpdateVC ()

7.69.2.5 void PanelR2::UpdateMesh ()

7.69.2.6 bool PanelR2::VCMouseEvent (int *id*, int *event*, VECTOR3 & *p*)

Here is the call graph for this function:

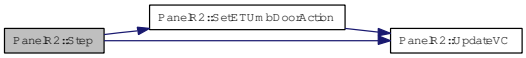


7.69.2.7 bool PanelR2::VCRedrawEvent (int *id*, int *event*, SURFHANDLE *surf*)

Here is the call graph for this function:

7.69.2.8 void PanelR2::Step (double *t*, double *dt*)

Here is the call graph for this function:



7.69.2.9 bool PanelR2::ParseScenarioLine (char * *line*)

7.69.2.10 void PanelR2::SaveState (FILEHANDLE *scn*)

7.69.2.11 bool PanelR2::VCDrawTalkback (SURFHANDLE *surf*, int *idx*, int *label*) [private]

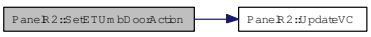
7.69.2.12 void PanelR2::CheckMPSArmed (int *eng*) [private]

7.69.2.13 void PanelR2::CheckAPUReadytoStart () [private]

7.69.2.14 void PanelR2::CheckAPUShutdown () [private]

7.69.2.15 void PanelR2::SetETUmbDoorAction (AnimState::Action *action*, int *door*) [private]

Here is the call graph for this function:



7.69.3 Friends And Related Function Documentation

7.69.3.1 friend class Atlantis [friend]

7.69.3.2 friend class CRT [friend]

7.69.4 Member Data Documentation

7.69.4.1 Atlantis* PanelR2::sts [private]

7.69.4.2 int PanelR2::BOILER_N2_SUPPLY[3] [private]

7.69.4.3 int PanelR2::BOILER_CNTLR[3] [private]

7.69.4.4 int PanelR2::BOILER_CNTLRHTR[3] [private]

7.69.4.5 int PanelR2::APU_FUEL_TK_VLV[3] [private]

7.69.4.6 int PanelR2::APU_CNTLR_PWR[3] [private]

7.69.4.7 int PanelR2::HYD_CIRC_PUMP[3] [private]

7.69.4.8 int PanelR2::APU_AUTO_SHUT_DOWN [private]

7.69.4.9 int PanelR2::APU_SPEED_SELECT[3] [private]

7.69.4.10 int PanelR2::APU_CONTROL[3] [private]

7.69.4.11 int PanelR2::HYD_MAIN_PUMP_PRESS[3] [private]

7.69.4.12 int PanelR2::MPS_ENGINE_POWER[2][3] [private]

7.69.4.13 int PanelR2::HE_ISOLATION_A[3] [private]

7.69.4.14 int PanelR2::HE_ISOLATION_B[3] [private]

7.69.4.15 int PanelR2::CENTERLINE_LATCH [private]

7.69.4.16 int PanelR2::LEFT_DOOR [private]

7.69.4.17 int PanelR2::LEFT_LATCH [private]

7.69.4.18 int PanelR2::RIGHT_DOOR [private]

7.69.4.19 int PanelR2::RIGHT_LATCH [private]

7.69.4.20 int PanelR2::APU_STATE[3] [private]

7.69.4.21 int PanelR2::MPS_STATE[3] [private]

7.69.4.22 bool PanelR2::APU_READY[3] [private]

7.69.4.23 double PanelR2::APU_FUEL[3] [private]

7.69.4.24 int PanelR2::Hydraulic_Press[3] [private]

7.69.4.25 double PanelR2::APU_Speed[3] [private]

7.69.4.26 int PanelR2::Fuel_Press[3] [private]

7.69.4.27 int PanelR2::tkbk_state[8] [private]

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelR2.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelR2.cpp

7.70 PayloadBayOp Class Reference

```
#include <PlBayOp.h>
```

Collaboration diagram for PayloadBayOp:

Public Member Functions

- PayloadBayOp (Atlantis *_sts)
- void OpenDialog ()
- void DefineAnimations (UINT vcidx)
- void RegisterVC ()
- void UpdateVC ()
- bool VCMouseEvent (int id, int event, VECTOR3 &p)
- bool VCRedrawEvent (int id, int event, SURFHANDLE surf)
- void Step (double t, double dt)
- AnimState::Action GetDoorAction () const
- void SetDoorAction (AnimState::Action action)
- void AutoDoorSequenceControl ()
- AnimState::Action GetRadiatorAction () const
- void SetRadiatorAction (AnimState::Action action)
- void SetRadLatchAction (AnimState::Action action)
- void SetKuAntennaAction (AnimState::Action action)
- void RevertKuAntennaAction ()
- bool ParseScenarioLine (char *line)
- void SaveState (FILEHANDLE scn)

Private Types

- enum { BD_ENABLE, BD_DISABLE }
- enum { BDO_OPEN, BDO_STOP, BDO_CLOSE }
- enum { MP_ON, MP_OFF }
- enum { LC_RELEASE, LC_OFF, LC_LATCH }
- enum { RC_DEPLOY, RC_OFF, RC_STOW }
- enum { KU_DEPLOY, KU_GND, KU_STOW }
- enum { KU_DIRECT_ON, KU_DIRECT_OFF }

Private Member Functions

- BOOL DlgProc (HWND hWnd, UINT uMsg, WPARAM wParam, LPARAM lParam)
- void UpdateDialog (HWND hWnd)
- bool VCDrawTalkback (SURFHANDLE surf, int idx, int label)

Private Attributes

- Atlantis * sts
- HWND hDlg
- enum PayloadBayOp:: { ... } BayDoor [2]
- enum PayloadBayOp:: { ... } BayDoorOp
- enum PayloadBayOp:: { ... } MechPwr [2]
- enum PayloadBayOp:: { ... } RadLatchCtrl [2]
- enum PayloadBayOp:: { ... } RadiatorCtrl [2]
- enum PayloadBayOp:: { ... } KuCtrl
- enum PayloadBayOp:: { ... } KuDirectCtrl
- UINT anim_VC_R13L [11]
- int tkbk_state [6]
- AnimState BayDoorStatus
- AnimState CLBayDoorLatch [4]
- AnimState RadiatorStatus
- AnimState RadLatchStatus
- AnimState KuAntennaStatus

Friends

- class Atlantis
- class PanelA8
- BOOL CALLBACK PlOp_DlgProc (HWND hWnd, UINT uMsg, WPARAM wParam, LPARAM lParam)

7.70.1 Member Enumeration Documentation

7.70.1.1 anonymous enum [private]

Enumerator:

BD_ENABLE

BD_DISABLE

7.70.1.2 anonymous enum [private]

Enumerator:

BDO_OPEN

BDO_STOP

BDO_CLOSE

7.70.1.3 anonymous enum [private]

Enumerator:

MP_ON

MP_OFF

7.70.1.4 anonymous enum [private]

Enumerator:

LC_RELEASE

LC_OFF

LC_LATCH

7.70.1.5 anonymous enum [private]

Enumerator:

RC_DEPLOY

RC_OFF

RC_STOW

7.70.1.6 anonymous enum [private]

Enumerator:

KU_DEPLOY

KU_GND

KU_STOW

7.70.1.7 anonymous enum [private]

Enumerator:

KU_DIRECT_ON

KU_DIRECT_OFF

7.70.2 Constructor & Destructor Documentation

7.70.2.1 PayloadBayOp::PayloadBayOp (Atlantis * *_sts*)

7.70.3 Member Function Documentation

7.70.3.1 void PayloadBayOp::OpenDialog ()

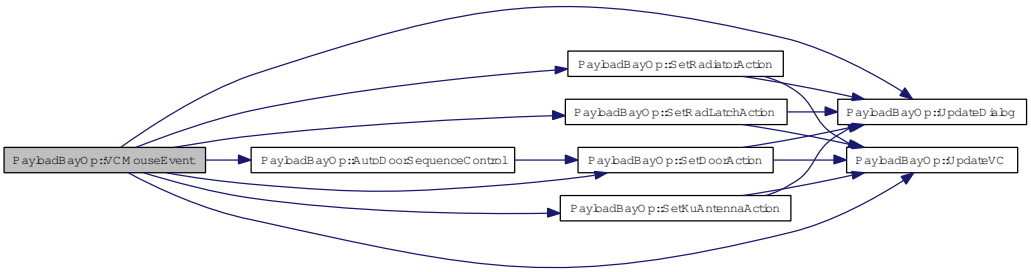
7.70.3.2 void PayloadBayOp::DefineAnimations (UINT *vcidx*)

7.70.3.3 void PayloadBayOp::RegisterVC ()

7.70.3.4 void PayloadBayOp::UpdateVC ()

7.70.3.5 bool PayloadBayOp::VCMouseEvent (int *id*, int *event*, VECTOR3 & *p*)

Here is the call graph for this function:

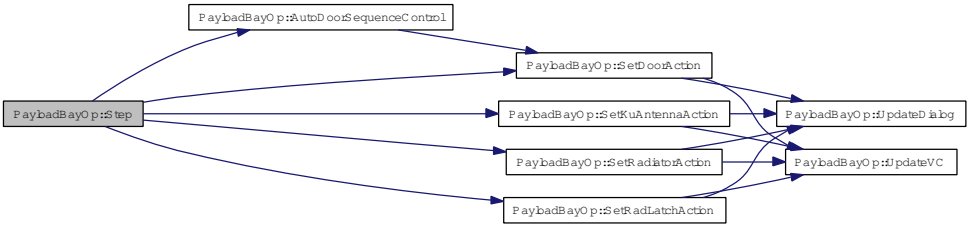


7.70.3.6 bool PayloadBayOp::VCRedrawEvent (int *id*, int *event*, SURFHANDLE *surf*)

Here is the call graph for this function:

7.70.3.7 void PayloadBayOp::Step (double *t*, double *dt*)

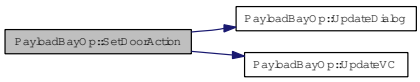
Here is the call graph for this function:



7.70.3.8 AnimState::Action PayloadBayOp::GetDoorAction () const [inline]

7.70.3.9 void PayloadBayOp::SetDoorAction (AnimState::Action *action*)

Here is the call graph for this function:



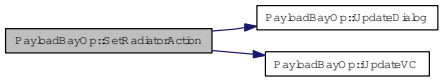
7.70.3.10 void PayloadBayOp::AutoDoorSequenceControl ()

Here is the call graph for this function:

7.70.3.11 AnimState::Action PayloadBayOp::GetRadiatorAction () const [inline]

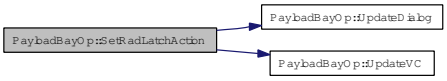
7.70.3.12 void PayloadBayOp::SetRadiatorAction (AnimState::Action *action*)

Here is the call graph for this function:



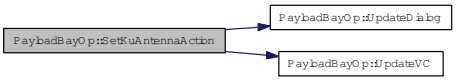
7.70.3.13 void PayloadBayOp::SetRadLatchAction (AnimState::Action *action*)

Here is the call graph for this function:



7.70.3.14 void PayloadBayOp::SetKuAntennaAction (AnimState::Action *action*)

Here is the call graph for this function:



7.70.3.15 void PayloadBayOp::RevertKuAntennaAction ()

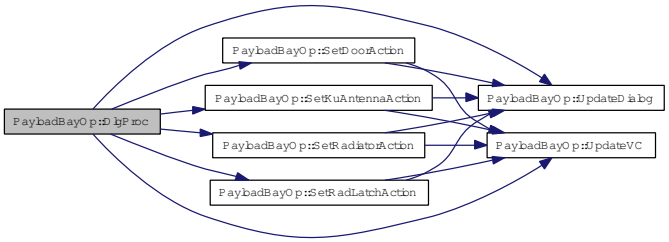
Here is the call graph for this function:

7.70.3.16 bool PayloadBayOp::ParseScenarioLine (char * *line*)

7.70.3.17 void PayloadBayOp::SaveState (FILEHANDLE *scn*)

7.70.3.18 BOOL PayloadBayOp::DlgProc (HWND *hWnd*, UINT *uMsg*, WPARAM *wParam*, LPARAM *lParam*) [private]

Here is the call graph for this function:



7.70.3.19 void PayloadBayOp::UpdateDialog (HWND *hWnd*) [private]

7.70.3.20 bool PayloadBayOp::VCDrawTalkback (SURFHANDLE *surf*, int *idx*, int *label*) [private]

7.70.4 Friends And Related Function Documentation

7.70.4.1 friend class Atlantis [friend]

7.70.4.2 friend class PanelA8 [friend]

7.70.4.3 BOOL CALLBACK PlOp_DlgProc (HWND *hWnd*, UINT *uMsg*, WPARAM *wParam*, LPARAM *lParam*) [friend]

7.70.5 Member Data Documentation

7.70.5.1 Atlantis* PayloadBayOp::sts [private]

7.70.5.2 HWND PayloadBayOp::hDlg [private]

7.70.5.3 enum { ... } PayloadBayOp::BayDoor[2] [private]

7.70.5.4 enum { ... } PayloadBayOp::BayDoorOp [private]

7.70.5.5 enum { ... } PayloadBayOp::MechPwr[2] [private]

7.70.5.6 enum { ... } PayloadBayOp::RadLatchCtrl[2] [private]

7.70.5.7 enum { ... } PayloadBayOp::RadiatorCtrl[2] [private]

7.70.5.8 enum { ... } PayloadBayOp::KuCtrl [private]

7.70.5.9 enum { ... } PayloadBayOp::KuDirectCtrl [private]

7.70.5.10 UINT PayloadBayOp::anim_VC_R13L[11] [private]

7.70.5.11 int PayloadBayOp::tkbk_state[6] [private]

7.70.5.12 AnimState PayloadBayOp::BayDoorStatus [private]

7.70.5.13 AnimState PayloadBayOp::CLBayDoorLatch[4] [private]

7.70.5.14 AnimState PayloadBayOp::RadiatorStatus [private]

7.70.5.15 AnimState PayloadBayOp::RadLatchStatus [private]

7.70.5.16 AnimState PayloadBayOp::KuAntennaStatus [private]

The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PlBayOp.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PlBayOp.cpp

7.71 dps::RAM16 Class Reference

```
#include <RAM16.h>
```

Public Member Functions

- RAM16 (unsigned long size)
- ~RAM16 ()
- short GetWord (unsigned long ulAddress)
- long GetDWord (unsigned long ulAddress)
- void SetWord (unsigned long ulAddress, short sWord)
- void OnParseLine (char *line)
- void OnSaveState (FILEHANDLE scn)
- void ReadImage (const string &strFileName)
- void WriteImage (const string &strFileName) const

Private Attributes

- short * pMemory

7.71.1 Constructor & Destructor Documentation

7.71.1.1 dps::RAM16::RAM16 (unsigned long *size*)

7.71.1.2 dps::RAM16::~RAM16 ()

7.71.2 Member Function Documentation

7.71.2.1 short dps::RAM16::GetWord (unsigned long *ulAddress*)

7.71.2.2 long dps::RAM16::GetDWord (unsigned long *ulAddress*)

7.71.2.3 void dps::RAM16::SetWord (unsigned long *ulAddress*, short *sWord*)

7.71.2.4 void dps::RAM16::OnParseLine (char * *line*)

7.71.2.5 void dps::RAM16::OnSaveState (FILEHANDLE *scn*)

7.71.2.6 void dps::RAM16::ReadImage (const string & *strFileName*)

7.71.2.7 void dps::RAM16::WriteImage (const string & *strFileName*) const

7.71.3 Member Data Documentation

7.71.3.1 short* dps::RAM16::pMemory [private]

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

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/RAM16.h

7.72 RefPoints Struct Reference

```
#include <Atlantis.h>
```

Public Attributes

- VECTOR3 Pitch
- VECTOR3 Yaw
- VECTOR3 Roll

7.72.1 Member Data Documentation

7.72.1.1 VECTOR3 RefPoints::Pitch

7.72.1.2 VECTOR3 RefPoints::Yaw

7.72.1.3 VECTOR3 RefPoints::Roll

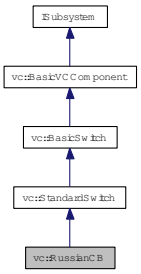
The documentation for this struct was generated from the following file:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Atlantis.h

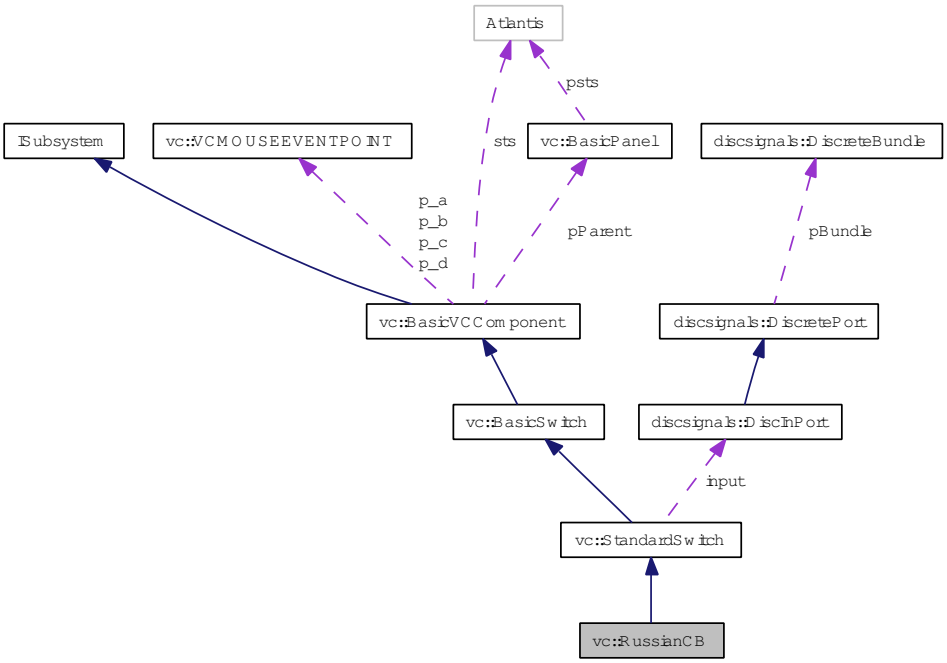
7.73 vc::RussianCB Class Reference

`#include <StandardSwitch.h>`

Inheritance diagram for vc::RussianCB:



Collaboration diagram for vc::RussianCB:



Public Member Functions

- RussianCB (Atlantis *_sts, const string &_ident)
- virtual ~RussianCB ()
- virtual void DefineVCAnimations (UINT vc_idx)

7.73.1 Constructor & Destructor Documentation

7.73.1.1 vc::RussianCB::RussianCB (Atlantis * *_sts*, const string & *_ident*)

7.73.1.2 virtual vc::RussianCB::~RussianCB () [virtual]

7.73.2 Member Function Documentation

7.73.2.1 virtual void vc::RussianCB::DefineVCAnimations (UINT *vc_idx*) [virtual]

Reimplemented from vc::StandardSwitch.

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

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/StandardSwitch.h

7.74 dps::ShuttleBus Class Reference

```
#include <ShuttleBus.h>
```

Public Member Functions

- ShuttleBus (const string &_ident)
- virtual ~ShuttleBus ()

7.74.1 Constructor & Destructor Documentation

7.74.1.1 dps::ShuttleBus::ShuttleBus (const string & *_ident*)

7.74.1.2 dps::ShuttleBus::~ShuttleBus () [virtual]

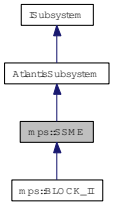
The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/ShuttleBus.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/ShuttleBus.cpp

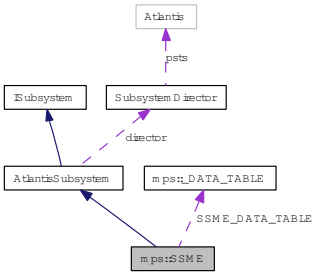
7.75 mps::SSME Class Reference

`#include <SSME.h>`

Inheritance diagram for mps::SSME:



Collaboration diagram for mps::SSME:



Public Member Functions

- bool cmdPurgeSequence1 (void)
- bool cmdPurgeSequence2 (void)
- bool cmdPurgeSequence3 (void)
- bool cmdPurgeSequence4 (void)
- bool cmdStartEnable (void)
- bool cmdStart (void)
- bool cmdControllerReset (void)
- bool cmdCheckoutStandby (void)
- bool cmdTerminateSequence (void)
- bool cmdShutdownEnable (void)
- bool cmdShutdown (void)
- bool cmdFRT1 (void)
- bool cmdOxidizerDump (void)
- bool cmdExitPROM (void)
- bool cmdPowerOn (void)
- bool cmdChannelReset (void)
- bool cmdThrottle (double)
- bool cmdPowerOff (void)

- int dataDataTable (DATA_TABLE *)
- SSME (SubsystemDirector *_director, const string &_ident, int nID)
- ~SSME (void)
- virtual void Realize ()
- virtual THRUSTER_HANDLE GetHandle () const
- void OnPostStep (double, double, double)
- virtual void OnSaveState (FILEHANDLE) const =0
- virtual bool OnParseLine (const char *)=0

Protected Member Functions

- void Ignition (double)
- void Shutdown (double)
- void Throttling (double)
- void SetCOTime (void)
- double PCfromOSFStoSTS (double)
- double PCfromSTStoOSFS (double)
- double PCfromPCTtoPSI (double)
- virtual double dcPC_ESC (double)=0
- virtual double dcMOV_ESC (double)=0
- virtual double dcMFV_ESC (double)=0
- virtual double dcCCV_ESC (double)=0
- virtual double dcFPOV_ESC (double)=0
- virtual double dcOPOV_ESC (double)=0
- virtual double dcPC_CO (double)=0
- virtual double dcMOV_CO (double)=0
- virtual double dcMFV_CO (double)=0
- virtual double dcCCV_CO (double)=0
- virtual double dcFPOV_CO (double, double)=0
- virtual double dcOPOV_CO (double, double)=0
- virtual double dcPC_MS (double)=0
- virtual double dcCCV_MS (double)=0
- virtual double dcFPOV_MS (double)=0
- virtual double dcOPOV_MS (double)=0

Protected Attributes

- int ID
- THRUSTER_HANDLE thSSME
- PROPELLANT_HANDLE phET
- VECTOR3 pos
- VECTOR3 dir
- double ISP0
- double ISP1
- double FPL_THRUST
- double COtimecoef
- double OPOV_pos_CO
- double FPOV_pos_CO
- double PSN4time

- int MPL
- int FPL
- DATA_TABLE * SSME_DATA_TABLE
- DCU activeDCU

7.75.1 Constructor & Destructor Documentation

7.75.1.1 mps::SSME::SSME (SubsystemDirector * *_director*, const string & *_ident*, int *nID*)

Create a new SSME object.

Parameters:

_director reference to the subsystem director

_ident identification of the subsystem inside the model

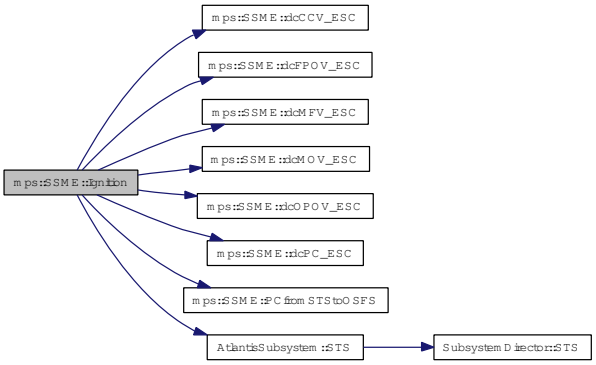
nID identification number of the SSME

7.75.1.2 mps::SSME::~SSME (void)

7.75.2 Member Function Documentation

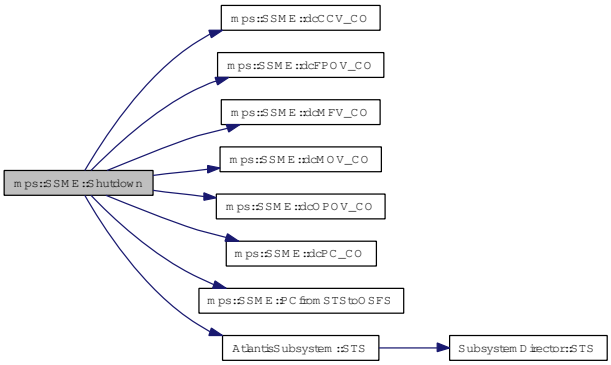
7.75.2.1 void mps::SSME::Ignition (double *time*) [protected]

Here is the call graph for this function:



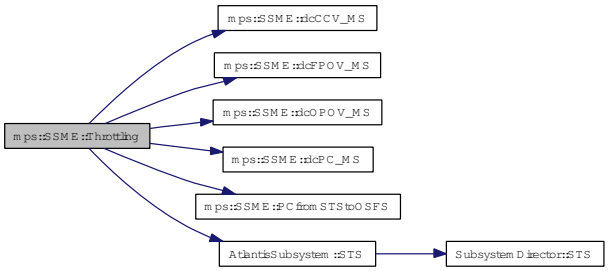
7.75.2.2 void mps::SSME::Shutdown (double *time*) [protected]

Here is the call graph for this function:



7.75.2.3 void mps::SSME::Throttling (double *dtime*) [protected]

Here is the call graph for this function:



7.75.2.4 void mps::SSME::SetCOTime (void) [protected]

7.75.2.5 double mps::SSME::PCfromOSFStoSTS (double *pcOSFS*) [protected]

7.75.2.6 double mps::SSME::PCfromSTStoOSFS (double *pcSTS*) [protected]

7.75.2.7 double mps::SSME::PCfromPCTtoPSI (double *pcPCT*) [protected]

7.75.2.8 virtual double mps::SSME::dcPC_ESC (double) [protected, pure virtual]

Implemented in mps::BLOCK_II.

7.75.2.9 virtual double mps::SSME::dcMOV_ESC (double) [protected, pure virtual]

Implemented in mps::BLOCK_II.

7.75.2.10 virtual double mps::SSME::dcMFV_ESC (double) [protected, pure virtual]

Implemented in mps::BLOCK_II.

7.75.2.11 virtual double mps::SSME::dcCCV_ESC (double) [protected, pure virtual]

Implemented in mps::BLOCK_II.

7.75.2.12 virtual double mps::SSME::dcFPOV_ESC (double) [protected, pure virtual]

Implemented in mps::BLOCK_II.

7.75.2.13 virtual double mps::SSME::dcOPOV_ESC (double) [protected, pure virtual]

Implemented in mps::BLOCK_II.

7.75.2.14 virtual double mps::SSME::dcPC_CO (double) [protected, pure virtual]

Implemented in mps::BLOCK_II.

7.75.2.15 virtual double mps::SSME::dcMOV_CO (double) [protected, pure virtual]

Implemented in mps::BLOCK_II.

7.75.2.16 virtual double mps::SSME::dcMFV_CO (double) [protected, pure virtual]

Implemented in mps::BLOCK_II.

7.75.2.17 virtual double mps::SSME::dcCCV_CO (double) [protected, pure virtual]

Implemented in mps::BLOCK_II.

7.75.2.18 virtual double mps::SSME::dcFPOV_CO (double, double) [protected, pure virtual]

Implemented in mps::BLOCK_II.

7.75.2.19 virtual double mps::SSME::dcOPOV_CO (double, double) [protected, pure virtual]

Implemented in mps::BLOCK_II.

7.75.2.20 virtual double mps::SSME::dcPC_MS (double) [protected, pure virtual]

Implemented in mps::BLOCK_II.

7.75.2.21 virtual double mps::SSME::dcCCV_MS (double) [protected, pure virtual]

Implemented in mps::BLOCK_II.

7.75.2.22 virtual double mps::SSME::dcFPOV_MS (double) [protected, pure virtual]

Implemented in mps::BLOCK_II.

7.75.2.23 virtual double mps::SSME::dcOPOV_MS (double) [protected, pure virtual]

Implemented in mps::BLOCK_II.

7.75.2.24 bool mps::SSME::cmdPurgeSequence1 (void)

7.75.2.25 bool mps::SSME::cmdPurgeSequence2 (void)

7.75.2.26 bool mps::SSME::cmdPurgeSequence3 (void)

7.75.2.27 bool mps::SSME::cmdPurgeSequence4 (void)

7.75.2.28 bool mps::SSME::cmdStartEnable (void)

7.75.2.29 bool mps::SSME::cmdStart (void)

7.75.2.30 bool mps::SSME::cmdControllerReset (void)

7.75.2.31 bool mps::SSME::cmdCheckoutStandby (void)

7.75.2.32 bool mps::SSME::cmdTerminateSequence (void)

7.75.2.33 bool mps::SSME::cmdShutdownEnable (void)

7.75.2.34 bool mps::SSME::cmdShutdown (void)

7.75.2.35 bool mps::SSME::cmdFRT1 (void)

7.75.2.36 bool mps::SSME::cmdOxidizerDump (void)

7.75.2.37 bool mps::SSME::cmdExitPROM (void)

7.75.2.38 bool mps::SSME::cmdPowerOn (void)

7.75.2.39 bool mps::SSME::cmdChannelReset (void)

7.75.2.40 bool mps::SSME::cmdThrottle (double *itgtPC*)

7.75.2.41 bool mps::SSME::cmdPowerOff (void)

7.75.2.42 int mps::SSME::dataDataTable (DATA_TABLE * *ptrDataTable*)

7.75.2.43 void mps::SSME::Realize () `[virtual]`

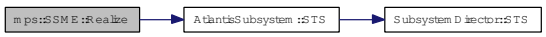
Create the thruster reference of the SSME and link the cable connections of the SSME.

See also:

AtlantisSubsystem::Realize

Reimplemented from AtlantisSubsystem.

Here is the call graph for this function:



7.75.2.44 THRUSTER_HANDLE mps::SSME::GetHandle () const [virtual]

7.75.2.45 void mps::SSME::OnPostStep (double *fSimT*, double *fDeltaT*, double *fMJD*) [virtual]

Finish calculations after the propagation of the state vector

Parameters:

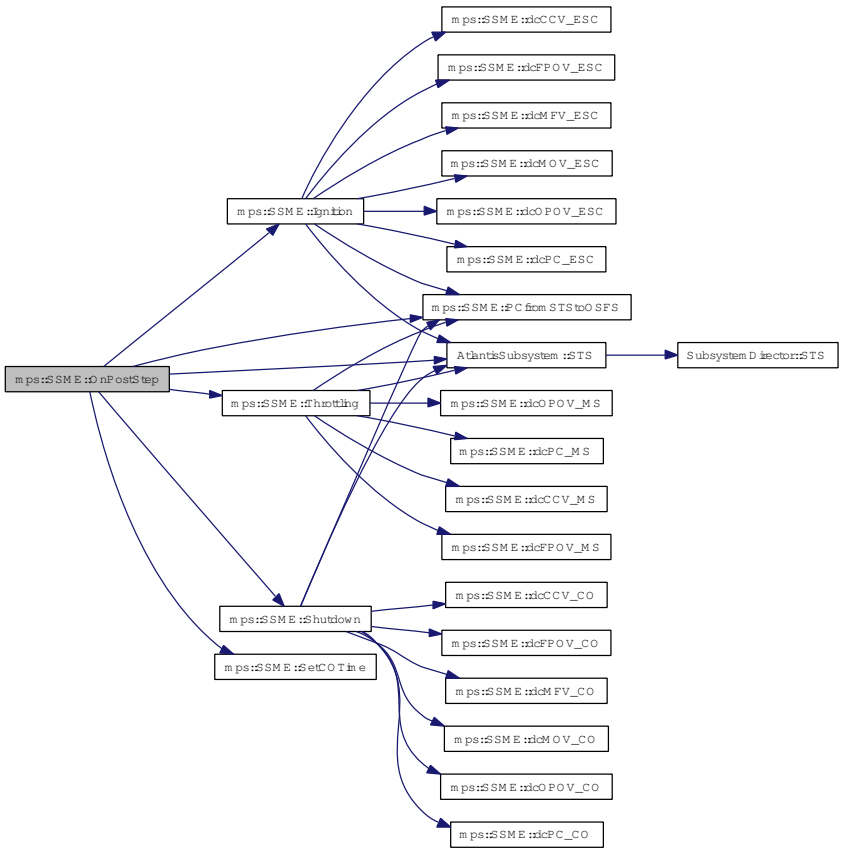
fSimT current simulation time

fDeltaT Simulation time step length

fMJD current modified julian date

Reimplemented from AtlantisSubsystem.

Here is the call graph for this function:



7.75.2.46 virtual void mps::SSME::OnSaveState (FILEHANDLE *scn*) const [pure virtual]

Save the state of the subsystem

Reimplemented from AtlantisSubsystem.

Implemented in mps::BLOCK_II.

7.75.2.47 virtual bool mps::SSME::OnParseLine (const char * *line*) [pure virtual]

Parse a line of the scenario file.

Warning:

Should get replaced by a block oriented scenario file format, like NASSP, making this function deprecated.

Reimplemented from AtlantisSubsystem.

Implemented in mps::BLOCK_II.

7.75.3 Member Data Documentation

7.75.3.1 int mps::SSME::ID [protected]

7.75.3.2 THRUSTER_HANDLE mps::SSME::thSSME [protected]

7.75.3.3 PROPELLANT_HANDLE mps::SSME::phET [protected]

7.75.3.4 VECTOR3 mps::SSME::pos [protected]

7.75.3.5 VECTOR3 mps::SSME::dir [protected]

7.75.3.6 double mps::SSME::ISP0 [protected]

7.75.3.7 double mps::SSME::ISP1 [protected]

7.75.3.8 double mps::SSME::FPL_THRUST [protected]

7.75.3.9 double mps::SSME::COtimecoef [protected]

7.75.3.10 double mps::SSME::OPOV_pos_CO [protected]

7.75.3.11 double mps::SSME::FPOV_pos_CO [protected]

7.75.3.12 double mps::SSME::PSN4time [protected]

7.75.3.13 int mps::SSME::MPL [protected]

7.75.3.14 int mps::SSME::FPL [protected]

7.75.3.15 DATA_TABLE* mps::SSME::SSME_DATA_TABLE [protected]

7.75.3.16 DCU mps::SSME::activeDCU [protected]

The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/SSME.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/SSME.cpp

7.76 SSU_FSS_RSS Class Reference

```
#include <SSU_FSS_RSS.h>
```

Public Types

- enum DoorStatus { DOOR_CLOSED, DOOR_OPEN, DOOR_CLOSING, DOOR_OPENING }

Public Member Functions

- SSU_FSS_RSS (OBJHANDLE hObj, int fmodel)
- void SetClassCaps (FILEHANDLE cfg)
- void DefineAnimations (void)
- void LoadState (FILEHANDLE scn, void *vs)
- void SaveState (FILEHANDLE scn)
- void Timestep (double simt)
- void RevertOAA (void)
- void RevertGVA (void)
- void RevertRSS (void)

Public Attributes

- enum SSU_FSS_RSS::DoorStatus OAA_status
- enum SSU_FSS_RSS::DoorStatus GVA_status
- enum SSU_FSS_RSS::DoorStatus RSS_status

Private Attributes

- UINT anim_OAA
- UINT anim_GVA
- UINT anim_RSS
- double OAA_proc
- double GVA_proc
- double RSS_proc

7.76.1 Member Enumeration Documentation

7.76.1.1 enum SSU_FSS_RSS::DoorStatus

Enumerator:

DOOR_CLOSED

DOOR_OPEN

DOOR_CLOSING

DOOR_OPENING

7.76.2 Constructor & Destructor Documentation

7.76.2.1 SSU_FSS_RSS::SSU_FSS_RSS (OBJHANDLE *hObj*, int *fmodel*)

Here is the call graph for this function:

7.76.3 Member Function Documentation

7.76.3.1 void SSU_FSS_RSS::SetClassCaps (FILEHANDLE *cfg*)

7.76.3.2 void SSU_FSS_RSS::DefineAnimations (void)

7.76.3.3 void SSU_FSS_RSS::LoadState (FILEHANDLE *scn*, void * *vs*)

7.76.3.4 void SSU_FSS_RSS::SaveState (FILEHANDLE *scn*)

7.76.3.5 void SSU_FSS_RSS::Timestep (double *simt*)

7.76.3.6 void SSU_FSS_RSS::RevertOAA (void)

7.76.3.7 void SSU_FSS_RSS::RevertGVA (void)

7.76.3.8 void SSU_FSS_RSS::RevertRSS (void)

7.76.4 Member Data Documentation

7.76.4.1 enum SSU_FSS_RSS::DoorStatus SSU_FSS_RSS::OAA_status

7.76.4.2 enum SSU_FSS_RSS::DoorStatus SSU_FSS_RSS::GVA_status

7.76.4.3 enum SSU_FSS_RSS::DoorStatus SSU_FSS_RSS::RSS_status

7.76.4.4 UINT SSU_FSS_RSS::anim_OAA [private]

7.76.4.5 UINT SSU_FSS_RSS::anim_GVA [private]

7.76.4.6 UINT SSU_FSS_RSS::anim_RSS [private]

7.76.4.7 double SSU_FSS_RSS::OAA_proc [private]

7.76.4.8 double SSU_FSS_RSS::GVA_proc [private]

7.76.4.9 double SSU_FSS_RSS::RSS_proc [private]

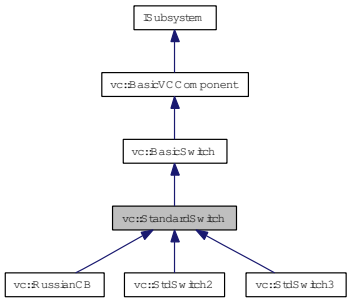
The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/FSSRSS/SSU_FSS_RSS.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/FSSRSS/SSU_FSS_RSS.cpp

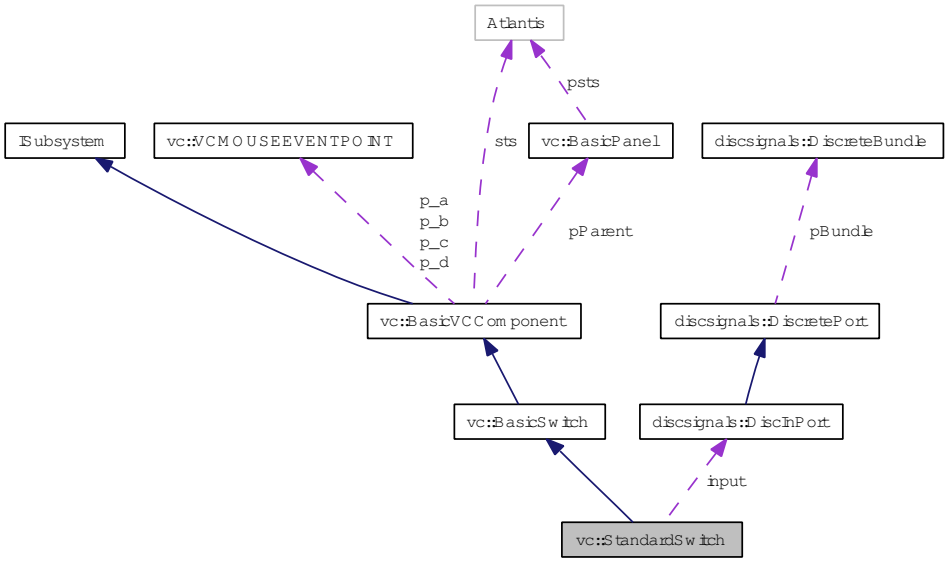
7.77 vc::StandardSwitch Class Reference

`#include <StandardSwitch.h>`

Inheritance diagram for vc::StandardSwitch:



Collaboration diagram for vc::StandardSwitch:



Public Member Functions

- StandardSwitch (Atlantis *psts, unsigned short usNumPositions, const string &_ident)
- virtual ~StandardSwitch ()
- void DefineSwitchGroup (UINT _grpIndex)
- virtual void DefineVCAnimations (UINT vc_idx)
- virtual void OnPositionChange (unsigned short usNewPosition)

- virtual void Realize ()
- virtual bool ConnectPort (unsigned short usPort, DiscreteBundle *pBundle, unsigned short usLine)

Private Attributes

- UINT grpIndex
- MGROUP_ROTATE * pswitchrot
- UINT anim_switch
- DiscInPort input

7.77.1 Constructor & Destructor Documentation

7.77.1.1 vc::StandardSwitch::StandardSwitch (Atlantis * *psts*, unsigned short *usNumPositions*, const string & *_ident*)

7.77.1.2 vc::StandardSwitch::~StandardSwitch () [virtual]

7.77.2 Member Function Documentation

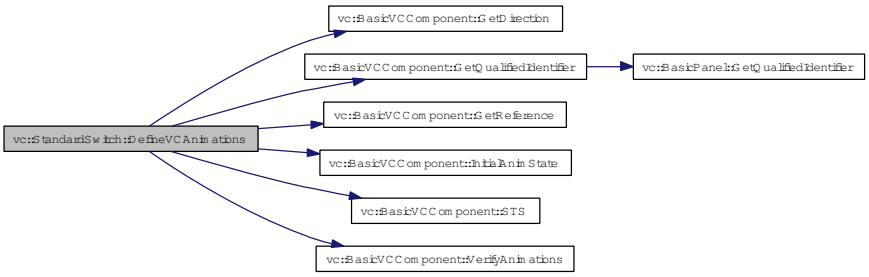
7.77.2.1 void vc::StandardSwitch::DefineSwitchGroup (UINT *_grpIndex*)

7.77.2.2 void vc::StandardSwitch::DefineVCAnimations (UINT *vc_idx*) [virtual]

Reimplemented from vc::BasicSwitch.

Reimplemented in vc::RussianCB.

Here is the call graph for this function:



7.77.2.3 void vc::StandardSwitch::OnPositionChange (unsigned short *usNewPosition*) [virtual]

Reimplemented from vc::BasicSwitch.

Here is the call graph for this function:

7.77.2.4 void vc::StandardSwitch::Realize () [virtual]

Override this function to create all Orbiter objects required: Thrusters, animations, meshes.

Reimplemented from vc::BasicVCComponent.

7.77.2.5 virtual bool vc::StandardSwitch::ConnectPort (unsigned short *usPort*, DiscreteBundle * *pBundle*, unsigned short *usLine*) [virtual]

Reimplemented in vc::StdSwitch2, and vc::StdSwitch3.

7.77.3 Member Data Documentation

7.77.3.1 UINT vc::StandardSwitch::grpIndex [private]

7.77.3.2 MGROUP_ROTATE* vc::StandardSwitch::pswitchrot [private]

7.77.3.3 UINT vc::StandardSwitch::anim_switch [private]

7.77.3.4 DiscInPort vc::StandardSwitch::input [private]

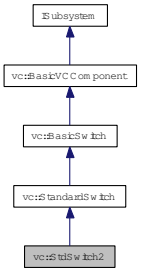
The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/StandardSwitch.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/StandardSwitch.cpp

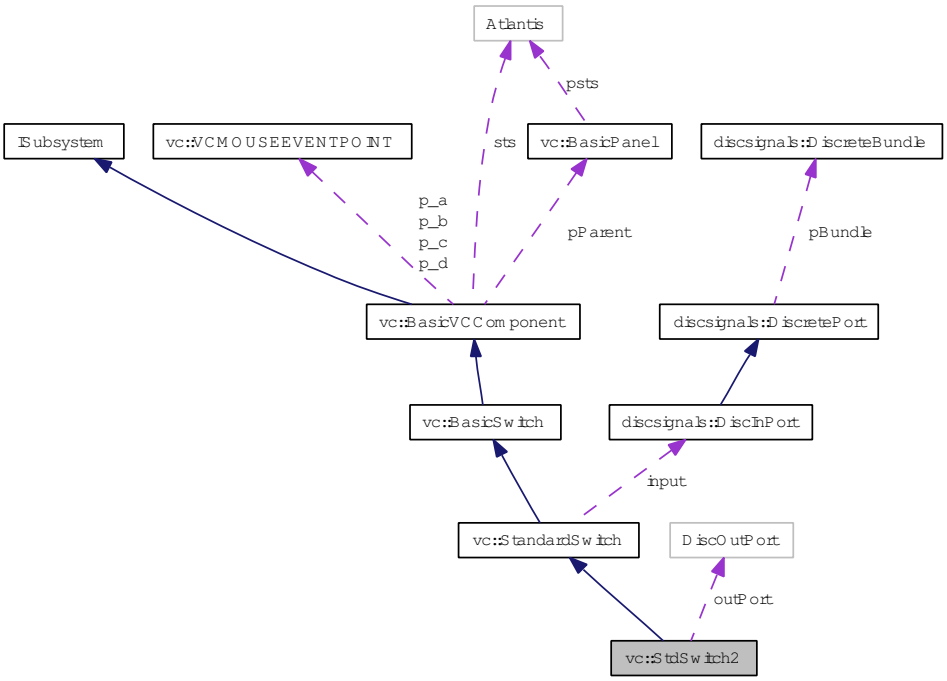
7.78 vc::StdSwitch2 Class Reference

`#include <StandardSwitch.h>`

Inheritance diagram for vc::StdSwitch2:



Collaboration diagram for vc::StdSwitch2:



Public Member Functions

- StdSwitch2 (Atlantis *_sts, const string &_ident)
- virtual ~StdSwitch2 ()
- virtual bool ConnectPort (unsigned short usPort, DiscreteBundle *pBundle, unsigned short usLine)

Protected Attributes

- DiscOutPort outPort

7.78.1 Constructor & Destructor Documentation

7.78.1.1 vc::StdSwitch2::StdSwitch2 (Atlantis * *_sts*, const string & *_ident*)

7.78.1.2 vc::StdSwitch2::~StdSwitch2 () [virtual]

7.78.2 Member Function Documentation

7.78.2.1 virtual bool vc::StdSwitch2::ConnectPort (unsigned short *usPort*, DiscreteBundle * *pBundle*, unsigned short *usLine*) [virtual]

Reimplemented from vc::StandardSwitch.

7.78.3 Member Data Documentation

7.78.3.1 DiscOutPort vc::StdSwitch2::outPort [protected]

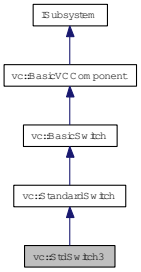
The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/StandardSwitch.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/StandardSwitch.cpp

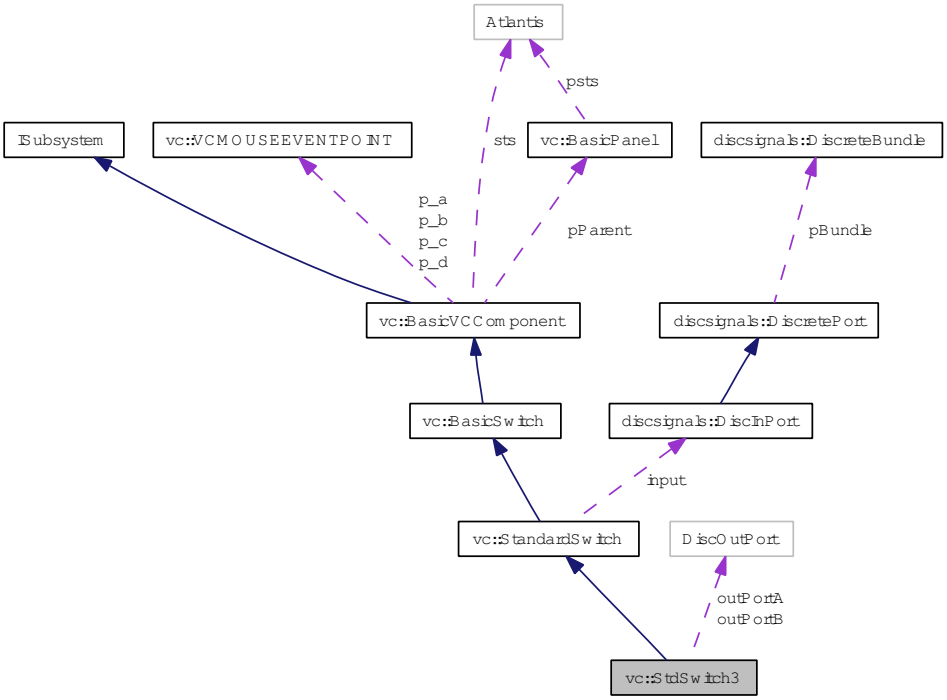
7.79 vc::StdSwitch3 Class Reference

`#include <StandardSwitch.h>`

Inheritance diagram for vc::StdSwitch3:



Collaboration diagram for vc::StdSwitch3:



Public Member Functions

- StdSwitch3 (Atlantis *_sts, const string &_ident)
- virtual ~StdSwitch3 ()

- virtual bool ConnectPort (unsigned short usPort, DiscreteBundle *pBundle, unsigned short usLine)

Protected Attributes

- DiscOutPort outPortA
- DiscOutPort outPortB

7.79.1 Constructor & Destructor Documentation

7.79.1.1 vc::StdSwitch3::StdSwitch3 (Atlantis * *_sts*, const string & *_ident*)

7.79.1.2 vc::StdSwitch3::~StdSwitch3 () [virtual]

7.79.2 Member Function Documentation

7.79.2.1 virtual bool vc::StdSwitch3::ConnectPort (unsigned short *usPort*, DiscreteBundle * *pBundle*, unsigned short *usLine*) [virtual]

Reimplemented from vc::StandardSwitch.

7.79.3 Member Data Documentation

7.79.3.1 DiscOutPort vc::StdSwitch3::outPortA [protected]

7.79.3.2 DiscOutPort vc::StdSwitch3::outPortB [protected]

The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/StandardSwitch.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/StandardSwitch.cpp

7.80 Stopwatch Class Reference

```
#include <Stopwatch.h>
```

Public Member Functions

- Stopwatch ()
- ~Stopwatch ()
- void Start ()
- double Stop ()

Private Attributes

- LARGE_INTEGER liFrequency
- LARGE_INTEGER liCurrent
- LARGE_INTEGER liLast

7.80.1 Constructor & Destructor Documentation

7.80.1.1 Stopwatch::Stopwatch ()

7.80.1.2 Stopwatch::~Stopwatch ()

7.80.2 Member Function Documentation

7.80.2.1 void Stopwatch::Start ()

7.80.2.2 double Stopwatch::Stop ()

7.80.3 Member Data Documentation

7.80.3.1 LARGE_INTEGER Stopwatch::liFrequency [private]

7.80.3.2 LARGE_INTEGER Stopwatch::liCurrent [private]

7.80.3.3 LARGE_INTEGER Stopwatch::liLast [private]

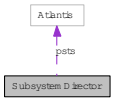
The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/util/Stopwatch.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/util/Stopwatch.cpp

7.81 SubsystemDirector Class Reference

`#include <SubsystemDirector.h>`

Collaboration diagram for SubsystemDirector:



Public Member Functions

- virtual Atlantis ∗ STS ()
- SubsystemDirector (Atlantis ∗_sts)
- ∼SubsystemDirector ()
- bool AddSubsystem (AtlantisSubsystem ∗pSubsys)
- void SetClassCaps (FILEHANDLE cfg)
- bool ParseScenarioLine (char ∗line)
- bool PlaybackEvent (double fSimT, double fEventT, const char ∗event_t, const char ∗event)
- bool SaveState (FILEHANDLE scn)
- bool PostStep (double fSimT, double fDeltaT, double fMJD)
- bool PreStep (double fSimT, double fDeltaT, double fMJD)
- bool WriteLog (const AtlantisSubsystem ∗src, char ∗message)
- virtual DiscreteBundleManager ∗ BundleManager () const

Protected Attributes

- Atlantis ∗ psts
- vector< AtlantisSubsystem ∗ > subsystems

7.81.1 Constructor & Destructor Documentation

7.81.1.1 SubsystemDirector::SubsystemDirector (Atlantis ∗ *_sts*)

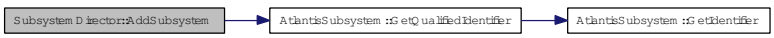
7.81.1.2 SubsystemDirector::∼SubsystemDirector ()

7.81.2 Member Function Documentation

7.81.2.1 Atlantis ∗ SubsystemDirector::STS () `[virtual]`

7.81.2.2 bool SubsystemDirector::AddSubsystem (AtlantisSubsystem ∗ *pSubsys*)

Here is the call graph for this function:



7.81.2.3 void SubsystemDirector::SetClassCaps (FILEHANDLE *cfg*)

7.81.2.4 bool SubsystemDirector::ParseScenarioLine (char * *line*)

7.81.2.5 bool SubsystemDirector::PlaybackEvent (double *fSimT*, double *fEventT*, const char * *event_t*, const char * *event*)

7.81.2.6 bool SubsystemDirector::SaveState (FILEHANDLE *scn*)

7.81.2.7 bool SubsystemDirector::PostStep (double *fSimT*, double *fDeltaT*, double *fMJD*)

7.81.2.8 bool SubsystemDirector::PreStep (double *fSimT*, double *fDeltaT*, double *fMJD*)

7.81.2.9 bool SubsystemDirector::WriteLog (const AtlantisSubsystem * *src*, char * *message*)

7.81.2.10 DiscreteBundleManager * SubsystemDirector::BundleManager () const [virtual]

7.81.3 Member Data Documentation

7.81.3.1 Atlantis* SubsystemDirector::psts [protected]

7.81.3.2 vector<AtlantisSubsystem*> SubsystemDirector::subsystems [protected]

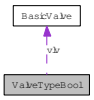
The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/SubsystemDirector.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/SubsystemDirector.cpp

7.82 ValveTypeBool Class Reference

`#include <ValveTypeBool.h>`

Collaboration diagram for ValveTypeBool:



Public Member Functions

- bool Open (void)
- bool Close (void)
- bool GetPos (void)
- void tmestp (double)
- void _backdoor (bool)
- ValveTypeBool (bool, double)
- ~ValveTypeBool (void)

Private Attributes

- BasicValve * vlv
- bool pos

7.82.1 Constructor & Destructor Documentation

7.82.1.1 ValveTypeBool::ValveTypeBool (bool *initpos*, double *imaxrate*)

7.82.1.2 ValveTypeBool::~ValveTypeBool (void)

7.82.2 Member Function Documentation

7.82.2.1 bool ValveTypeBool::Open (void)

Here is the call graph for this function:

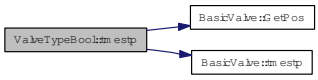
7.82.2.2 bool ValveTypeBool::Close (void)

Here is the call graph for this function:

7.82.2.3 bool ValveTypeBool::GetPos (void)

7.82.2.4 void ValveTypeBool::tmestp (double *dtme*)

Here is the call graph for this function:



7.82.2.5 void ValveTypeBool::_backdoor (bool *ipos*)

Here is the call graph for this function:

7.82.3 Member Data Documentation

7.82.3.1 BasicValve* ValveTypeBool::vlv [private]

7.82.3.2 bool ValveTypeBool::pos [private]

The documentation for this class was generated from the following files:

- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/ValveTypeBool.h
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/ValveTypeBool.cpp

7.83 vc::VCMOUSEEVENTPOINT Struct Reference

```
#include <vc_defs.h>
```

Public Attributes

- float x
- float y

7.83.1 Member Data Documentation

7.83.1.1 float vc::VCMOUSEEVENTPOINT::x

7.83.1.2 float vc::VCMOUSEEVENTPOINT::y

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

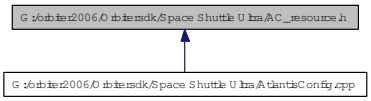
- G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/vc_defs.h

Chapter 8

File Documentation

8.1 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AC_resource.h File Reference

This graph shows which files directly or indirectly include this file:



Defines

- #define IDD_ACONFIG 101
- #define IDC_RADIO1 1000
- #define IDC_RADIO2 1001
- #define IDC_RADIO3 1002
- #define IDC_RADIO4 1003

8.1.1 Define Documentation

8.1.1.1 #define IDC_RADIO1 1000

8.1.1.2 #define IDC_RADIO2 1001

8.1.1.3 #define IDC_RADIO3 1002

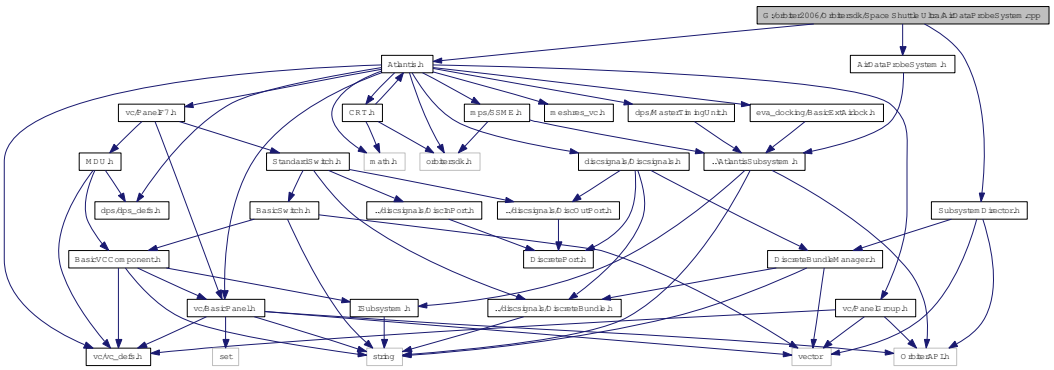
8.1.1.4 #define IDC_RADIO4 1003

8.1.1.5 #define IDD_ACONFIG 101

8.2 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AirDataProbeSystem.cpp File Reference

```
#include "Atlantis.h"
#include "SubsystemDirector.h"
#include "AirDataProbeSystem.h"
```

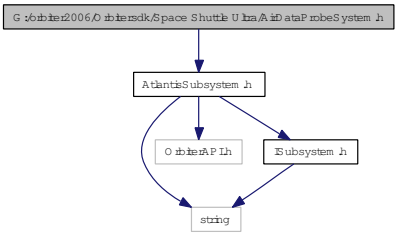
Include dependency graph for AirDataProbeSystem.cpp:



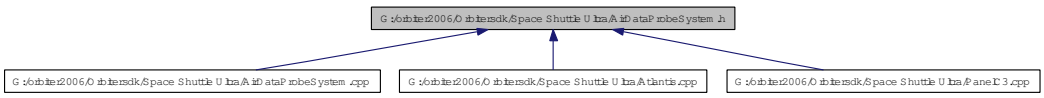
8.3 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AirDataProbeSystem.h File Reference

```
#include "AtlantisSubsystem.h"
```

Include dependency graph for AirDataProbeSystem.h:



This graph shows which files directly or indirectly include this file:



Classes

- class AirDataProbeSystem

Defines

- #define AFX_AIRDATAPROBESYSTEM_H__74A5598A_2DB5_47F6_AC02_B106E9C86F81_-_INCLUDED_

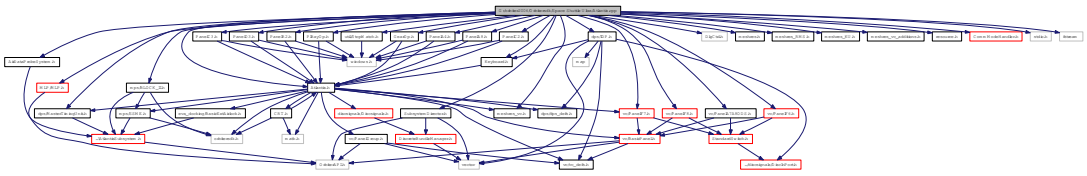
8.3.1 Define Documentation

8.3.1.1 #define AFX_AIRDATAPROBESYSTEM_H__74A5598A_2DB5_47F6_AC02_-B106E9C86F81__INCLUDED_

8.4 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Atlantis.cpp File Reference

```
#include "Atlantis.h"
#include "PlBayOp.h"
#include "GearOp.h"
#include "PanelA4.h"
#include "PanelA8.h"
#include "PanelC2.h"
#include "PanelC3.h"
#include "PanelO3.h"
#include "PanelR2.h"
#include "Keyboard.h"
#include "DlgCtrl.h"
#include "meshres.h"
#include "meshres_vc.h"
#include "meshres_RMS.h"
#include "meshres_KU.h"
#include "meshres_vc_additions.h"
#include "resource.h"
#include "SubsystemDirector.h"
#include "dps/MasterTimingUnit.h"
#include "dps/IDP.h"
#include "AirDataProbeSystem.h"
#include "mps/BLOCK_II.h"
#include "vc/PanelA7A8ODS.h"
#include "vc/PanelF6.h"
#include "vc/PanelF7.h"
#include "vc/PanelF8.h"
#include "CommModeHandler.h"
#include "MLP/MLP.h"
#include <stdio.h>
#include <fstream>
#include "util/StopWatch.h"
```

Include dependency graph for Atlantis.cpp:



Defines

- #define STRICT 1
- #define ORBITER_MODULE
- #define GRP_Endeffector 10
- #define LOADBMP(id) (LoadBitmap (g_Param.hDLL, MAKEINTRESOURCE (id)))

Functions

- BOOL CALLBACK Atlantis_DlgProc (HWND, UINT, WPARAM, LPARAM)
- BOOL CALLBACK RMS_DlgProc (HWND, UINT, WPARAM, LPARAM)
- void GetSRB_State (double met, double &thrust_level, double &prop_level)
- int tpir (const double ∗list, int n_items, double target)
- double linterp (double x0, double y0, double x1, double y1, double x)
- double listerp (const double ∗listx, const double ∗listy, int n_items, double x)
- double tableterp (const double ∗table, const double ∗listrow, int n_row, const double ∗listcol, int n_col, double rowlookup, double collookup)
- void FlatPlateCoeff (double aoa, double ∗cl, double ∗cm, double ∗cd)
- void VLiftCoeff (VESSEL ∗vv, double aoa, double M, double Re, void ∗stuff, double ∗cl, double ∗cm, double ∗cd)
- void HLiftCoeff (double beta, double M, double Re, double ∗cl, double ∗cm, double ∗cd)
- DLLCLBK void InitModule (HINSTANCE hModule)
- DLLCLBK void ExitModule (HINSTANCE hModule)
- DLLCLBK VESSEL ∗ ovcInit (OBJHANDLE hvessel, int flightmodel)
- DLLCLBK void ovcExit (VESSEL ∗vessel)
- BOOL CALLBACK PAYCAM_DlgProc (HWND hWnd, UINT uMsg, WPARAM wParam, LPARAM lParam)
- DLLCLBK bool gpcReadValue (VESSEL ∗pVessel, UINT gpc, UINT val_index, DWORD ∗value)
- DLLCLBK bool gpcSetValue (VESSEL ∗pVessel, UINT gpc, UINT val_index, const DWORD value)
- double angle (const VECTOR3 dir, const VECTOR3 dir0)

Variables

- GDIParams g_Param
- char ∗ ActionString [5] = {"STOPPED", "ISCLOSED", "ISOPEN", "CLOSE", "OPEN"}
- HELPCONTEXT g_hc
- const double RollOff = 10∗47.880259
- const double PitchOff = 40∗47.880259

- const double YawOff = 1.0
- const int n_mach = 21
- const double mach [n_mach] = {0.25,0.4,0.6,0.8,0.85,0.9,0.92,0.95,0.98,1.05,1.1,1.2,1.3,1.5,2,2.5,3,4,5,8,10}
- const int n_aoa1 = 19
- const double aoa1 [n_aoa1] = {-10,-5,-2.5,0,2.5,5,7.5,10,12.5,15,17.5,20,22.5,25,30,35,40,45,50}
- const double clBase [n_aoa1][n_mach]
- const double cdBase [n_aoa1][n_mach]
- const double cmBase [n_aoa1][n_mach]
- const int n_trimExt = 3
- const double trimExt [n_trimExt] = {-11.7,0,22.5}
- const double cmTrim [n_trimExt][n_mach]
- const double OrbiterS = 2690∗0.3048∗0.3048
- const double Orbiterb = 78.056∗0.3048
- const double Orbiterc = 39.56∗0.3048
- const double OrbiterA = Orbiterb∗Orbiterb/OrbiterS

8.4.1 Define Documentation

8.4.1.1 #define GRP_Endeffector 10

8.4.1.2 #define LOADBMP(id) (LoadBitmap (g_Param.hDLL, MAKEINTRESOURCE (id)))

8.4.1.3 #define ORBITER_MODULE

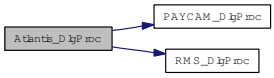
8.4.1.4 #define STRICT 1

8.4.2 Function Documentation

8.4.2.1 double angle (const VECTOR3 *dir*, const VECTOR3 *dir0*) [inline]

8.4.2.2 BOOL CALLBACK Atlantis_DlgProc (HWND *hWnd*, UINT *uMsg*, WPARAM *wParam*, LPARAM *lParam*)

Here is the call graph for this function:



8.4.2.3 DLLCLBK void ExitModule (HINSTANCE *hModule*)

8.4.2.4 void FlatPlateCoeff (double *aoa*, double * *cl*, double * *cm*, double * *cd*)

8.4.2.5 void GetSRB_State (double *met*, double & *thrust_level*, double & *prop_level*)

8.4.2.6 DLLCLBK bool gpcReadValue (VESSEL * *pVessel*, UINT *gpc*, UINT *val_index*, DWORD * *value*)

8.4.2.7 DLLCLBK bool gpcSetValue (VESSEL * *pVessel*, UINT *gpc*, UINT *val_index*, const DWORD *value*)

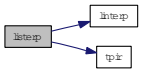
8.4.2.8 void HLiftCoeff (double *beta*, double *M*, double *Re*, double * *cl*, double * *cm*, double * *cd*)

8.4.2.9 DLLCLBK void InitModule (HINSTANCE *hModule*)

8.4.2.10 double linterp (double *x0*, double *y0*, double *x1*, double *y1*, double *x*)

8.4.2.11 double listerp (const double * *listx*, const double * *listy*, int *n_items*, double *x*)

Here is the call graph for this function:



8.4.2.12 DLLCLBK void ovcExit (VESSEL * *vessel*)

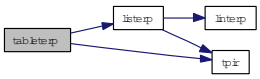
8.4.2.13 DLLCLBK VESSEL* ovcInit (OBJHANDLE *hvessel*, int *flightmodel*)

8.4.2.14 BOOL CALLBACK PAYCAM_DlgProc (HWND *hWnd*, UINT *uMsg*, WPARAM *wParam*, LPARAM *lParam*)

8.4.2.15 BOOL CALLBACK RMS_DlgProc (HWND *hWnd*, UINT *uMsg*, WPARAM *wParam*, LPARAM *lParam*)

8.4.2.16 double tableterp (const double * *table*, const double * *listrow*, int *n_row*, const double * *listcol*, int *n_col*, double *rowlookup*, double *collookup*)

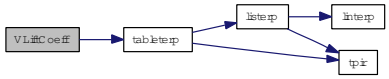
Here is the call graph for this function:



8.4.2.17 int tpir (const double ∗ *list*, int *n_items*, double *target*)

8.4.2.18 void VLiftCoeff (VESSEL ∗ *vv*, double *aoa*, double *M*, double *Re*, void ∗ *stuff*, double ∗ *cl*, double ∗ *cm*, double ∗ *cd*)

Here is the call graph for this function:



8.4.3 Variable Documentation

8.4.3.1 char∗ ActionString[5] = {"STOPPED", "ISCLOSED", "ISOPEN", "CLOSE", "OPEN"}

8.4.3.2 const double aoa1[n_aoa1] = {-10,-5,-2.5,0,2.5,5,7.5,10,12.5,15,17.5,20,22.5,25,30,35,40,45,50}

8.4.3.3 const double cdBase[n_aoa1][n_mach]

Initial value:

```
{{0.11485,0.11975,0.13521,0.16897,0.18654,0.19703,0.20867,0.22925,0.25227,0.26933,0.27358,0.27477,0.27037,
{0.07985,0.08119,0.08489,0.09953,0.10785,0.11831,0.12552,0.14223,0.16817,0.18470,0.18942,0.19259,0.19067,0
{0.07170,0.07263,0.07517,0.08323,0.08675,0.09887,0.10710,0.12269,0.14461,0.16251,0.16647,0.17013,0.16958,0
{0.06740,0.06610,0.07020,0.07630,0.08000,0.09070,0.09900,0.11450,0.13460,0.15410,0.15840,0.16140,0.16110,0
{0.06687,0.05747,0.07011,0.07557,0.08048,0.09309,0.10180,0.11702,0.13644,0.16935,0.16376,0.16502,0.16395,0
{0.07038,0.07096,0.07350,0.08175,0.08916,0.10574,0.11515,0.13127,0.15286,0.17724,0.18173,0.18139,0.17763,0
{0.08056,0.06145,0.08464,0.10038,0.10990,0.12750,0.13806,0.15696,0.18221,0.20398,0.20794,0.20709,0.20226,0
{0.09599,0.09775,0.10146,0.13088,0.14389,0.16155,0.17477,0.19588,0.22487,0.24256,0.24485,0.24298,0.23613,0
{0.10625,0.12444,0.13381,0.17827,0.19090,0.21031,0.22196,0.24823,0.27902,0.29479,0.29645,0.29055,0.28325,0
{0.15784,0.16278,0.18832,0.23803,0.25555,0.27066,0.28350,0.31191,0.34859,0.35909,0.35895,0.35301,0.34037,0
{0.21346,0.21758,0.26637,0.30939,0.31973,0.34168,0.35577,0.38252,0.41746,0.43401,0.43355,0.42081,0.40809,0
{0.26859,0.29389,0.35519,0.38857,0.39759,0.42137,0.43584,0.46452,0.50331,0.51932,0.52016,0.50090,0.48367,0
{0.36651,0.37144,0.43592,0.45493,0.46003,0.48172,0.50467,0.53313,0.57864,0.59902,0.59994,0.58377,0.56200,0
{0.45537,0.45790,0.50401,0.50538,0.51675,0.54316,0.56348,0.59974,0.65071,0.67457,0.67979,0.67010,0.64751,0
{0.62879,0.63228,0.69595,0.69784,0.71354,0.75001,0.77807,0.82814,0.89852,0.93146,0.93867,0.92529,0.89410,0
{0.85388,0.85863,0.94509,0.94766,0.96898,1.01850,1.05660,1.12460,1.22017,1.26491,1.27470,1.25653,1.21417,1
{1.11596,1.12216,1.23516,1.23852,1.26638,1.33111,1.38090,1.46976,1.59467,1.65315,1.66594,1.64219,1.58683,1
{1.40622,1.41403,1.55642,1.56065,1.59576,1.67732,1.74007,1.85204,2.00944,2.08312,2.09924,2.06932,1.99956,1
{1.74092,1.75059,1.92688,1.93211,1.97558,2.07655,2.15424,2.29286,2.48772,2.57894,2.59890,2.56185,2.47549,2
```

8.4.3.4 const double clBase[n_aoa1][n_mach]

Initial value:

```
{{-0.50280,-0.51097,-0.52957,-0.58454,-0.59773,-0.61005,-0.62831,-0.65514,-0.65616,-0.64808,-0.64225,-0.61
{-0.25906,-0.27397,-0.28670,-0.32761,-0.33790,-0.34601,-0.35039,-0.35897,-0.36172,-0.36027,-0.35685,-0.334
{-0.15202,-0.16199,-0.17186,-0.19155,-0.19632,-0.20588,-0.21553,-0.21485,-0.19888,-0.20310,-0.19292,-0.182
{-0.05000,-0.05000,-0.05500,-0.06500,-0.06200,-0.06000,-0.06000,-0.06000,-0.05500,-0.04700,-0.03600,-0.025
{0.06214,0.06212,0.07201,0.06176,0.06652,0.07101,0.07563,0.08498,0.09414,0.10315,0.10796,0.10460,0.10295,0
{0.17453,0.17448,0.17928,0.18859,0.19295,0.20356,0.21579,0.23947,0.24762,0.25552,0.25513,0.25014,0.24043,0
{0.29705,0.29691,0.30657,0.31963,0.31838,0.32413,0.34997,0.38783,0.38955,0.39173,0.39121,0.38627,0.37178,0
```

```
{0.40448,0.40924,0.41674,0.44384,0.44172,0.44572,0.47690,0.51887,0.52899,0.53602,0.53054,0.51767,0.49146,0
{0.51599,0.52757,0.54905,0.56480,0.56566,0.56794,0.58585,0.64660,0.66026,0.66393,0.66357,0.63927,0.61835,0
{0.63581,0.65002,0.67426,0.68162,0.67309,0.68323,0.70567,0.76535,0.77783,0.78377,0.77863,0.75952,0.72567,0
{0.77676,0.78385,0.79709,0.79087,0.77995,0.79401,0.81577,0.86501,0.89069,0.89596,0.89086,0.85608,0.83073,0
{0.91657,0.92529,0.92851,0.89578,0.87690,0.89485,0.91619,0.96428,0.99805,1.00286,1.00255,0.95636,0.92538,0
{1.00634,1.01512,1.01007,0.93184,0.91266,0.92074,0.95994,1.00227,1.04837,1.06699,1.06660,1.03541,1.00114,0
{1.08364,1.08846,1.04489,0.92283,0.90795,0.93837,0.96751,1.02232,1.07579,1.10328,1.11188,1.09985,1.07176,0
{1.30045,1.30623,1.25395,1.10747,1.08961,1.12612,1.16109,1.22686,1.29103,1.32402,1.33434,1.31990,1.28619,1
{1.51313,1.51986,1.45902,1.28858,1.26781,1.31028,1.35097,1.42751,1.50217,1.54055,1.55256,1.53576,1.49654,1
{1.69584,1.70339,1.63520,1.44418,1.42090,1.46850,1.51411,1.59988,1.68356,1.72658,1.74004,1.72121,1.67725,1
{1.82367,1.83178,1.75845,1.55304,1.52800,1.57919,1.62823,1.72047,1.81045,1.85672,1.87119,1.85095,1.80367,1
{1.91170,1.92021,1.84334,1.62801,1.60176,1.65542,1.70683,1.80353,1.89785,1.94635,1.96152,1.94030,1.89074,1
```

8.4.3.5 const double cmBase[n_aoa1][n_mach]

Initial value:

```
{{0.03200,0.03400,0.04000,0.05900,0.06800,0.09000,0.10100,0.12600,0.14000,0.15800,0.16250,0.16200,0.14800,
{0.03000,0.03200,0.03650,0.04900,0.05400,0.06100,0.06700,0.08200,0.09800,0.11400,0.11800,0.11200,0.09200,0
{0.02900,0.03050,0.03470,0.04450,0.04700,0.04500,0.05300,0.06200,0.07800,0.08400,0.08200,0.07200,0.05800,0
{0.02800,0.29000,0.03300,0.04000,0.04000,0.03500,0.03800,0.04200,0.05500,0.05500,0.04700,0.03400,0.02500,0
{0.02700,0.02750,0.03050,0.03600,0.03800,0.03500,0.02200,0.02000,0.03100,0.02400,0.01500,0.00600,-0.00100,
{0.02600,0.02600,0.02800,0.03150,0.03200,0.02250,0.01600,0.00000,0.00500,-0.00300,-0.01050,-0.02000,-0.023
{0.02400,0.02400,0.02400,0.02250,0.02200,0.02000,0.01200,-0.01600,-0.01500,-0.02200,-0.02900,-0.04000,-0.0
{0.02300,0.02200,0.02050,0.01800,0.01700,0.01500,0.00400,-0.02700,-0.03000,-0.03400,-0.04000,-0.05400,-0.0
{0.02300,0.02250,0.02100,0.01700,0.01200,0.00750,-0.00600,-0.04000,-0.04200,-0.04500,-0.04900,-0.06100,-0.
{0.02400,0.02200,0.01600,0.00600,0.00050,-0.00250,-0.01800,-0.05000,-0.05200,-0.05800,-0.06200,-0.07200,-0
{0.01900,0.01600,0.00900,-0.00600,-0.01100,-0.01000,-0.02800,-0.05650,-0.06300,-0.07000,-0.07400,-0.08200,
{0.01400,0.00800,-0.00400,-0.00500,-0.01000,-0.00300,-0.03000,-0.06000,-0.07200,-0.07700,-0.08100,-0.08800
{0.01050,0.01000,0.00600,0.00260,0.02000,0.02000,-0.01200,-0.03500,-0.04700,-0.06100,-0.06800,-0.08000,-0.
{0.10000,0.01200,0.03300,0.08000,0.09000,0.09000,0.08700,0.07000,0.02000,-0.02300,-0.04200,-0.06400,-0.077
{0.12007,0.01441,0.03962,0.09606,0.10807,0.10807,0.10446,0.08405,0.02401,-0.02762,-0.05043,-0.07685,-0.092
{0.15655,0.01879,0.05166,0.12524,0.14089,0.14089,0.13619,0.10958,0.03131,-0.03601,-0.06575,-0.10019,-0.120
{0.19818,0.02378,0.06540,0.15855,0.17837,0.17837,0.17242,0.13873,0.03964,-0.04558,-0.08324,-0.12684,-0.152
{0.24466,0.02936,0.08074,0.19573,0.22020,0.22020,0.21286,0.17126,0.04893,-0.05627,-0.10276,-0.15658,-0.188
{0.32582,0.03910,0.10752,0.26066,0.29324,0.29324,0.28346,0.22807,0.06516,-0.07494,-0.13684,-0.20852,-0.250
```

8.4.3.6 const double cmTrim[n_trimExt][n_mach]

Initial value:

```
{{0.0158,0.016,0.0179,0.0204,0.020165,0.01993,0.020866,0.02149,0.01803,0.01574,0.0136,0.0148,0.01141,0.009
{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
{-0.0424,-0.0442,-0.0488,-0.05562,-0.0559,-0.05618,-0.058004,-0.05922,-0.05222,-0.05241,-0.04894,-0.04195,
```

8.4.3.7 HELPCONTEXT g_hc

Initial value:

```
 {
  "html/vessels/Atlantis.chm",
  0,
  "html/vessels/Atlantis.chm::/Atlantis.hhc",
  "html/vessels/Atlantis.chm::/Atlantis.hhk"
}
```

8.4.3.8 GDIParams g_Param

8.4.3.9 const double mach[n_mach] = {0.25,0.4,0.6,0.8,0.85,0.9,0.92,0.95,0.98,1.05,1.1,1.2,1.3,1.5,2,2.5,3,4,5,8,10}

8.4.3.10 const int n_aoa1 = 19

8.4.3.11 const int n_mach = 21

8.4.3.12 const int n_trimExt = 3

8.4.3.13 const double OrbiterA = Orbiterb∗Orbiterb/OrbiterS

8.4.3.14 const double Orbiterb = 78.056∗0.3048

8.4.3.15 const double Orbiterc = 39.56∗0.3048

8.4.3.16 const double OrbiterS = 2690∗0.3048∗0.3048

8.4.3.17 const double PitchOff = 40∗47.880259

8.4.3.18 const double RollOff = 10∗47.880259

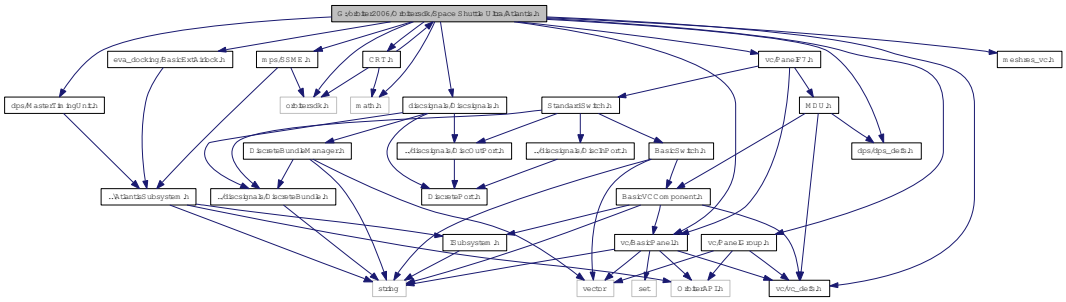
8.4.3.19 const double trimExt[n_trimExt] = {-11.7,0,22.5}

8.4.3.20 const double YawOff = 1.0

8.5 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Atlantis.h File Reference

```
#include "CRT.h"
#include "orbitersdk.h"
#include <math.h>
#include "meshres_vc.h"
#include "dps/dps_defs.h"
#include "dps/MasterTimingUnit.h"
#include "mps/SSME.h"
#include "vc/vc_defs.h"
#include "vc/PanelGroup.h"
#include "vc/BasicPanel.h"
#include "vc/PanelF7.h"
#include "discsignals/Discsignals.h"
#include "eva_docking/BasicExtAirlock.h"
```

Include dependency graph for Atlantis.h:



This graph shows which files directly or indirectly include this file:

Classes

- struct FLOWSTATE
- struct GDIParams
- struct RefPoints
- struct DAPConfig
- struct KeyboardInput
- class Atlantis
- class Atlantis_SRB
- class Atlantis_Tank

Defines

- #define AID_CDR1_BUTTONS 1
- #define AID_CDR2_BUTTONS 2
- #define AID_PLT1_BUTTONS 3
- #define AID_PLT2_BUTTONS 4
- #define AID_CRT1_BUTTONS 5
- #define AID_CRT2_BUTTONS 6
- #define AID_CRT3_BUTTONS 7
- #define AID_CRT4_BUTTONS 8
- #define AID_MFD1_BUTTONS 9
- #define AID_MFD2_BUTTONS 10
- #define AID_AFD_BUTTONS 11
- #define AID_CDR1_PWR 12
- #define AID_CDR2_PWR 13
- #define AID_PLT1_PWR 14
- #define AID_PLT2_PWR 15
- #define AID_CRT1_PWR 16
- #define AID_CRT2_PWR 17
- #define AID_CRT3_PWR 18
- #define AID_CRT4_PWR 19
- #define AID_MFD1_PWR 20
- #define AID_MFD2_PWR 21
- #define AID_AFD_PWR 22
- #define AID_CDR1_BRT 23
- #define AID_CDR2_BRT 24
- #define AID_PLT1_BRT 25
- #define AID_PLT2_BRT 26
- #define AID_CRT1_BRT 27
- #define AID_CRT2_BRT 28
- #define AID_CRT3_BRT 29
- #define AID_CRT4_BRT 30
- #define AID_MFD1_BRT 31
- #define AID_MFD2_BRT 32
- #define AID_AFD_BRT 33
- #define AID_R13L_MIN 100
- #define AID_R13L 100
- #define AID_R13L_TKBK1 101
- #define AID_R13L_TKBK2 102
- #define AID_R13L_TKBK3 103
- #define AID_R13L_TKBK4 104
- #define AID_R13L_TKBK5 105
- #define AID_R13L_TKBK6 106
- #define AID_R13L_MAX 120
- #define AID_F6_MIN 200
- #define AID_F6 200
- #define AID_F6_TKBK1 201
- #define AID_F6_TKBK2 202
- #define AID_F6_TKBK3 203
- #define AID_F6_MAX 220

- #define AID_F7_MIN 270
- #define AID_F7 270
- #define AID_F7_EVTTMR1 271
- #define AID_F7_EVTTMR2 272
- #define AID_F7_EVTTMR3 273
- #define AID_F7_EVTTMR4 274
- #define AID_F7_MAX 279
- #define AID_C2_MIN 280
- #define AID_C2 280
- #define AID_C2_WND0 281
- #define AID_C2_WND1 282
- #define AID_C2_WND2 283
- #define AID_C2_WND3 284
- #define AID_C2_MAX 289
- #define AID_KYBD_CDR 290
- #define AID_KYBD_PLT 291
- #define AID_C3_MIN 300
- #define AID_C3 300
- #define AID_C3_MAX 319
- #define AID_O3_MIN 320
- #define AID_O3 320
- #define AID_O3_METTMR1 321
- #define AID_O3_METTMR2 322
- #define AID_O3_RCS 323
- #define AID_O3_MAX 339
- #define AID_R2_MIN 400
- #define AID_R2 400
- #define AID_R2_TKBK1 401
- #define AID_R2_TKBK2 402
- #define AID_R2_TKBK3 403
- #define AID_R2_TKBK4 404
- #define AID_R2_TKBK5 405
- #define AID_R2_TKBK6 406
- #define AID_R2_TKBK7 407
- #define AID_R2_TKBK8 408
- #define AID_R2_MAX 419
- #define AID_A4_MIN 420
- #define AID_A4 420
- #define AID_A4_EVTTMR 421
- #define AID_A4_METTMR1 422
- #define AID_A4_METTMR2 423
- #define AID_A4_MAX 429
- #define AID_A8_MIN 430
- #define AID_A8 430
- #define AID_A8_TKBK1 431
- #define AID_A8_TKBK2 432
- #define AID_A8_TKBK3 433
- #define AID_A8_TKBK4 434
- #define AID_A8_TKBK5 435
- #define AID_A8_TKBK6 436

- #define AID_A8_TKBK7 437
- #define AID_A8_TKBK8 438
- #define AID_A8_TKBK9 439
- #define AID_A8_TKBK10 440
- #define AID_A8_TKBK11 441
- #define AID_A8_TKBK12 442
- #define AID_A8_TKBK13 443
- #define AID_A8_TKBK14 444
- #define AID_A8_TKBK15 445
- #define AID_A8_TKBK16 446
- #define AID_A8_TKBK17 447
- #define AID_A8_MAX 459
- #define AID_F8_MIN 460
- #define AID_F8 460
- #define AID_F8_MAX 470
- #define AID_MDU_CDR1 1000
- #define AID_MDU_CDR2 1001
- #define AID_MDU_CRT1 1002
- #define AID_MDU_CRT2 1003
- #define AID_MDU_CRT3 1004
- #define AID_MDU_CRT4 1005
- #define AID_MDU_MFD1 1006
- #define AID_MDU_MFD2 1007
- #define AID_MDU_PLT1 1008
- #define AID_MDU_PLT2 1009
- #define AID_MDU_AFD1 1010
- #define SWITCH1 0
- #define SWITCH2 1
- #define SWITCH3 2
- #define SWITCH4 3
- #define SWITCH5 4
- #define SWITCH6 5
- #define SWITCH7 6
- #define SWITCH8 7
- #define SWITCH9 8
- #define SWITCH10 9
- #define SWITCH11 10
- #define SWITCH12 11
- #define SWITCH13 12
- #define SWITCH14 13
- #define SWITCH15 14
- #define SWITCH16 15
- #define SWITCH17 16
- #define SWITCH18 17
- #define SWITCH19 18
- #define SWITCH20 19

Enumerations

- enum AXIS { PITCH, YAW, ROLL }
- enum LATCH_STATE { LT_LATCHED = 0, LT_OPENING, LT_CLOSING, LT_RELEASED }
- enum OMS_REF { OMS_LEFT = 0, OMS_KIT, OMS_RIGHT }

Variables

- const double LBM = 0.45359237
- const double MPS2FPS = 3.280839895
- const short VARSTATE_OK = 0
- const short VARSTATE_MISSING = 1
- const short VARSTATE_OFFSCALE_LOW = 2
- const short VARSTATE_LOW = 3
- const short VARSTATE_HIGH = 4
- const short VARSTATE_OFFSCALE_HIGH = 5
- static const char * DEFAULT_MESHNAME_VC = "SSU/VC"
- static const char * DEFAULT_MESHNAME_KU = "SSU/KU"
- static const char * DEFAULT_MESHNAME_COCKPIT = "SSU/Cockpit"
- static const char * DEFAULT_MESHNAME_MIDDECK = "SSU/MidDeck"
- static const char * DEFAULT_MESHNAME_RMS = "SSU/RMS"
- static const char * DEFAULT_MESHNAME_ORBITER = "SSU/Orbiter"
- static const char * DEFAULT_MESHNAME_ET = "SSU/ET125"
- static const char * DEFAULT_MESHNAME_LSRB = "SSU/LSRB"
- static const char * DEFAULT_MESHNAME_RSRB = "SSU/RSRB"
- static const char * ODSPANEL_MESHNAME = "SSU/ODSVC"
- static const char * DEFAULT_MESHNAME_ODS = "SSU/ODS"
- const double ORBITER_EMPTY_MASS = 81100.0
- const double ORBITER_MAX_PROPELLANT_MASS = 14538.0
- const double ORBITER_FRCS_PROPELLANT_MASS = 1464 * LBM + 923*LBM
- const double SSME_RATED_THRUST = 2090664.159
- const double ORBITER_MAIN_THRUST = 2170732.15
- const double ORBITER_MAX_THRUST = 2275094.273
- const double ORBITER_OMS_THRUST = 26700.0
- const double ORBITER_RCS_THRUST = 7740.0
- const double ORBITER_MAIN_ISP0 = 453*9.80665
- const double ORBITER_MAIN_ISP1 = 363*9.80665
- const double ORBITER_OMS_ISP0 = 316*9.80665
- const double ORBITER_OMS_ISP1 = ORBITER_OMS_ISP0*0.75
- const double ORBITER_RCS_ISP0 = ORBITER_OMS_ISP0
- const double ORBITER_RCS_ISP1 = ORBITER_RCS_ISP0*0.75
- const double ORBITER_OMS_PITCH = 15.0089
- const double ORBITER_OMS_YAW = 5.7000
- const double ORBITER_PITCH_TORQUE = 239900.00
- const double ORBITER_YAW_TORQUE = 239900.00
- const double ORBITER_ROLL_TORQUE = 41800.00
- const double OMS_HELIUM_TANK_MASS = 21.0
- const double APU_FUEL_TANK_MASS = 325.0*0.45359237
- const double APU_FUEL_TANK_FLOWRATE [2] = {(3.25*0.45359237)/60.0, (1.5*0.45359237)/60.0}

- const double GEAR_OPERATING_SPEED = 0.3
- const double DOOR_OPERATING_SPEED = 0.007353
- const double DOORLATCH_OPERATING_SPEED = 0.05
- const double RAD_OPERATING_SPEED = 0.023255814
- const double RADLATCH_OPERATING_SPEED = 0.0384615385
- const double KU_OPERATING_SPEED = 0.0435
- const double SPEEDBRAKE_OPERATING_SPEED = 0.20284
- const double ET_UMB_DOOR_OPERATING_SPEED = 0.0416667
- const double LATCH_OPERATING_SPEED = 0.166666667
- const double AIR_DATA_PROBE_SPEED = 0.06666667
- const double ARM_OPERATING_SPEED = 0.005
- const double ARM_TRANSLATE_SPEED = 0.1
- const double ARM_DEPLOY_SPEED = 0.0294117647
- const double ARM_GRAPPLE_SPEED = 0.4
- const double ARM_RIGID_SPEED = 0.25
- const double ARM_EXTEND_SPEED = 0.142857
- const double SHOULDER_BRACE_SPEED = 0.11765
- const VECTOR3 ARM_WRIST_CAM_OFFSET = {-0.091886, 0.276656, 0.666001}
- const double DYNP_MAX = 100e3
- const double GEAR_MAX_DEPLOY_SPEED = 450/1.943844
- const VECTOR3 ORBITER_CS = {234.8,389.1,68.2}
- const VECTOR3 ORBITER_CS_GEAR = {10.0,0.0,3.0}
- const double ORBITER_CW [4] = {0.13, 0.5, 1.4, 1.4}
- const double ORBITER_CW_GEAR [4] = {0.04, 0.04, 0.05, 0.0}
- const double MAX_GRAPPLING_DIST = 0.5
- const double MAX_GRAPPLING_ANGLE = 0.087266
- const double CAM_HIGHRATE_SPEED = 12
- const double CAM_LOWRATE_SPEED = 1.2
- const int NoseRotThrusters [6] = {0, 2, 4, 6, 8, 10}
- const int NosePitchThrusters [2] = {0, 2}
- const int NoseYawThrusters [2] = {4, 6}
- const int NoseRollThrusters [2] = {8, 10}
- const int AftRotThrusters [6] = {1, 3, 5, 7, 9, 11}
- const int AftPitchThrusters [2] = {1, 3}
- const int AftYawThrusters [2] = {5, 7}
- const int AftRollThrusters [2] = {9, 11}
- const double TMajorCycle = 1.0000
- const double TPEGStop = 7.000
- const double AXIS_TILT = 23.4458878∗RAD
- const double TANK_MAX_PROPELLANT_MASS = 719115.0
- const double TANK_EMPTY_MASS = 26535
- const double ET_SEP_TIME = 18.00000
- const double ET_SEP_RATE = 1.2192
- const double SRB_MAX_PROPELLANT_MASS = 502146.0
- const double SRB_EMPTY_MASS = 87603.65
- const double SRB_ISP0 = 2638.89
- const double SRB_ISP1 = 2368.79
- const double SRB_THRUST = 15.12904086E6
- const double MAX_ATT_LAUNCH = 1e-5
- const double MAX_ROLL_SRB = 2.5e-5

- const double SRB_STABILISATION_TIME = 6.6
- const double SRB_SEPARATION_TIME = 124.0
- const int SRB_nt = 30
- const double SRB_CUTOUT_TIME = 130.0
- const double BSM_MAX_PROPELLANT_MASS = 75.0 ∗ 0.45349
- const double BSM_THRUST0 = 82.76475E3
- const double BSM_ISP0 = 2400.0
- const UINT MIDX_ORBITER = 1
- const UINT MIDX_COCKPIT = 2
- const UINT MIDX_VC = 3
- const UINT MIDX_RMS = 4
- const UINT MIDX_MIDDECK = 5
- const UINT MIDX_KU_ANTENNA = 6
- const UINT MIDX_ET = 7
- const UINT MIDX_LSRB = 8
- const UINT MIDX_RSRB = 9
- const UINT MIDX_ODS = 10
- const UINT MIDX_ODS_VC = 11
- const UINT MIDX_PLPANEL_A7 = 12
- const UINT MIDX_PLPANEL_A8 = 13
- const UINT MIDX_PLPANEL_A9 = 14
- const double NO_SLAG_TIME = 240.0
- const double SLAG_SPREAD = 0.2
- const double SLAG1_RATE = 5
- const double SLAG2_RATE = 11.0
- const double SLAG3_RATE = 49.0
- const VECTOR3 SLAG_DIR = _V(0.0, 0.0, -1.0)
- const double fps_to_ms = 0.3048
- const double kg_to_pounds = 2.205
- const VECTOR3 OFS_ZERO = { 0.0, 0.0, 0.0 }
- const VECTOR3 OFS_LAUNCH_ORBITER = { 0.0, 6.04,-7.989}
- const VECTOR3 OFS_LAUNCH_TANK = { 0.0,-1.91, 5.72 }
- const VECTOR3 OFS_LAUNCH_RIGHTSRB = { 6.88,-1.91,-5.68 }
- const VECTOR3 OFS_LAUNCH_LEFTSRB = {-6.88,-1.91,-5.68 }
- const VECTOR3 OFS_WITHTANK_ORBITER = { 0.0, 4.658,-9.414}
- const VECTOR3 OFS_WITHTANK_TANK = { 0.0,-3.34, 4.33 }
- const VECTOR3 ORBITER_DOCKPOS = { 0.0, 2.3729,10.1529}
- const VECTOR3 OFS_MMU = {0,2.44,10.44}
- const VECTOR3 SSMER_REF = _V(1.458, -0.194, -11.7875)
- const VECTOR3 SSMEL_REF = _V(-1.458, -0.194, -11.7875)
- const VECTOR3 SSMET_REF = _V(0.0, 1.945, -10.76250)
- const VECTOR3 SSMET_GOX_REF = _V(1.149, 3.166, -13.91)
- const VECTOR3 SSMEL_GOX_REF = _V(-1.754, 1.274, -14.471)
- const VECTOR3 SSMER_GOX_REF = _V(2.615, 0.239, -14.665)
- const VECTOR3 POS_HDP = _V(0.0, -1.91, -25.8)
- const VECTOR3 UMBDOORL_REF = _V(-1.31087, -2.71022, -6.75496)
- const VECTOR3 UMBDOORR_REF = _V(1.31087, -2.71022, -6.75496)
- const VECTOR3 UMBDOOR_AXIS = _V(0, -0.05, 0.99875)
- const VECTOR3 PROBEL_REF = _V(-1.122688, -1.0894815, 19.4175)
- const VECTOR3 PROBER_REF = _V(1.122688, -1.0894815, 19.4175)

- const VECTOR3 PROBE_AXIS = _V(0.0, cos(15 ∗ RAD), sin(15∗RAD))
- const int STATE_PRELAUNCH = 0
- const int STATE_STAGE1 = 1
- const int STATE_STAGE2 = 2
- const int STATE_ORBITER = 3
- const int STATE_DESTROYED = 99
- const int RCS_F1L = 0
- const int RCS_F1U = 1
- const int RCS_F1D = 2
- const int RCS_F1F = 3
- const int RCS_F2R = 0
- const int RCS_F2U = 1
- const int RCS_F2D = 2
- const int RCS_F2F = 3
- const int RCS_F3L = 0
- const int RCS_F3U = 1
- const int RCS_F3D = 2
- const int RCS_F3F = 3
- const int RCS_F4R = 0
- const int RCS_F4D = 1
- const int RCS_F5L = 0
- const int RCS_F5R = 1
- const int RCS_L1L = 0
- const int RCS_L1U = 1
- const int RCS_L1A = 2
- const int RCS_L2L = 0
- const int RCS_L2U = 1
- const int RCS_L2D = 2
- const int RCS_L3L = 0
- const int RCS_L3D = 1
- const int RCS_L3A = 2
- const int RCS_L4L = 0
- const int RCS_L4U = 1
- const int RCS_L4D = 2
- const int RCS_L5R = 0
- const int RCS_L5D = 1
- const int RCS_R1R = 0
- const int RCS_R1U = 1
- const int RCS_R1A = 2
- const int RCS_R2R = 0
- const int RCS_R2U = 1
- const int RCS_R2D = 2
- const int RCS_R3R = 0
- const int RCS_R3D = 1
- const int RCS_R3A = 2
- const int RCS_R4R = 0
- const int RCS_R4U = 1
- const int RCS_R4D = 2
- const int RCS_R5R = 0
- const int RCS_R5D = 1

- const VECTOR3 RCS_F1L_OFS = _V(-1.7 ,-0.15, 17.40)
- const VECTOR3 RCS_F1L_DIR = _V(1,0,0)
- const VECTOR3 RCS_F1U_OFS = _V(-0.4 , 1.10, 17.9)
- const VECTOR3 RCS_F1U_DIR = _V(0, -1,0)
- const VECTOR3 RCS_F1D_OFS = _V(-1.55,-0.20, 18.25)
- const VECTOR3 RCS_F1D_DIR = _V(0.4339,0.8830,0.1793)
- const VECTOR3 RCS_F1F_OFS = _V(-0.4, 0.7, 19.0)
- const VECTOR3 RCS_F1F_DIR = _V(0, -0.0499, -0.9988)
- const VECTOR3 RCS_F2R_OFS = _V(1.75, 0.25, 17.45)
- const VECTOR3 RCS_F2R_DIR = _V(-1,0,0)
- const VECTOR3 RCS_F2U_OFS = _V(0.4, 1.10, 17.9)
- const VECTOR3 RCS_F2U_DIR = _V(0, -1,0)
- const VECTOR3 RCS_F2D_OFS = _V(1.60,-0.20, 18.25)
- const VECTOR3 RCS_F2D_DIR = _V(-0.4339,0.8830,0.1793)
- const VECTOR3 RCS_F2F_OFS = _V(0.4, 0.7, 19.0)
- const VECTOR3 RCS_F2F_DIR = _V(0, -0.0499, -0.9988)
- const VECTOR3 RCS_F3L_OFS = _V(-1.65, 0.25 , 17.45)
- const VECTOR3 RCS_F3L_DIR = _V(1,0,0)
- const VECTOR3 RCS_F3U_OFS = _V(0.0 , 1.15, 17.9)
- const VECTOR3 RCS_F3U_DIR = _V(0, -1,0)
- const VECTOR3 RCS_F3D_OFS = _V(-1.63,-0.18, 17.9)
- const VECTOR3 RCS_F3D_DIR = _V(0.4339,0.8830,0.1793)
- const VECTOR3 RCS_F3F_OFS = _V(0.0 , 0.75, 19.0)
- const VECTOR3 RCS_F3F_DIR = _V(0, -0.0499, -0.9988)
- const VECTOR3 RCS_F4R_OFS = _V(1.8 ,-0.15, 17.40)
- const VECTOR3 RCS_F4R_DIR = _V(-1,0,0)
- const VECTOR3 RCS_F4D_OFS = _V(1.68,-0.18, 17.9)
- const VECTOR3 RCS_F4D_DIR = _V(-0.4339,0.8830,0.1793)
- const VECTOR3 ODS_POS = _V(0.0, 0.0, 10.1529)
- const int VC_CDR = 0
- const int VC_PLT = 1
- const int VC_MS1 = 2
- const int VC_MS2 = 3
- const int VC_MS3 = 4
- const int VC_MS4 = 5
- const int VC_MS5 = 6
- const int VC_STBDSTATION = 7
- const int VC_AFTPILOT = 8
- const int VC_RMSSTATION = 9
- const int VC_PORTSTATION = 10
- const int VC_AFTWORKSTATION = 11
- const int VC_DOCKCAM = 12
- const int VC_PLBCAMFL = 13
- const int VC_PLBCAMFR = 14
- const int VC_PLBCAMBL = 15
- const int VC_PLBCAMBR = 16
- const int VC_LEECAM = 17
- const int VC_MIDDECK = 100
- const int VC_EXT_AL = 200
- static const char * VC_LBL_CDR = "Commander seat"

- const VECTOR3 VC_POS_CDR = _V(-0.671257, 2.629396, 14.1)
- static const char * VC_LBL_PLT = "Pilot seat"
- const VECTOR3 VC_POS_PLT = _V(0.671257, 2.629396, 14.1)
- static const char * VC_LBL_MS1 = "MS1 seat"
- const VECTOR3 VC_POS_MS1 = _V(0.2, 2.629396, 13.5)
- const VECTOR3 VC_DIR_MS1 = _V(0.0, 0, 1.0)
- static const char * VC_LBL_MS2 = "MS2 seat"
- const VECTOR3 VC_POS_MS2 = _V(0.7, 2.75, 13.5)
- const VECTOR3 VC_DIR_MS2 = _V(0.0, 0.0, 1.0)
- static const char * VC_LBL_COAS = "COAS"
- const VECTOR3 VC_POS_COAS = _V(0.4, 3.15, 12.6)
- const VECTOR3 VC_DIR_COAS = _V(0.0, 0.45399, -0.891007)
- static const char * VC_LBL_PORTSTATION = "Port workstation"
- const VECTOR3 VC_POS_PORTSTATION = _V(-0.6, 2.95, 13.0)
- const VECTOR3 VC_DIR_PORTSTATION = _V(-cos(25.0 * RAD), -sin(25.0 * RAD), 0)
- static const char * VC_LBL_AFTPILOT = "Aft pilot station"
- const VECTOR3 VC_POS_AFTPILOT = _V(0.4, 3.15, 12.7)
- const VECTOR3 VC_DIR_AFTPILOT = _V(0, 0, -1)
- const VECTOR3 VC_OFSFWD_AFTPILOT = VC_POS_COAS - VC_POS_AFTPILOT
- static const char * VC_LBL_RMSSTATION = "RMS work station"
- const VECTOR3 VC_POS_RMSSTATION = _V(-0.4, 3.15, 12.7)
- const VECTOR3 VC_DIR_RMSSTATION = _V(0.0, 0.0, -1.0)
- static const char * VC_LBL_STBDSTATION = "Starboard workstation"
- const VECTOR3 VC_POS_STBDSTATION = _V(0.6, 2.95, 13.0)
- const VECTOR3 VC_DIR_STBDSTATION = _V(cos(25.0 * RAD), -sin(25.0 * RAD), 0)
- static const char * VC_LBL_AFTWORKSTATION = "Aft Workstation"
- const VECTOR3 VC_POS_AFTWORKSTATION = _V(0.0, 2.95, 13.2)
- const VECTOR3 VC_DIR_AFTWORKSTATION = _V(0.0, 0.0, -1.0)
- static const char * VC_LBL_DOCKCAM = "ODS centerline camera"
- static const char * VC_LBL_LEECAM = "RMS EE camera"
- static const char * VC_LBL_PLBCAMFL = "Payload bay FL camera"
- static const char * VC_LBL_PLBCAMFR = "Payload bay FR camera"
- static const char * VC_LBL_PLBCAMBL = "Payload bay BL camera"
- static const char * VC_LBL_PLBCAMBR = "Payload bay BR camera"
- static const char * VC_LBL_MIDDECK = "Mid Deck"
- const VECTOR3 VC_POS_MIDDECK = _V(-1.44, 0.0, 13.353)
- const VECTOR3 VC_DIR_MIDDECK = _V(0.0, -sin(24.5*RAD), cos(24.5 * RAD))
- static const char * VC_LBL_EXT_AL = "External Airlock"
- const VECTOR3 VC_POS_EXT_AL = ODS_POS + _V(0.0, 0.3, 0.25)
- const VECTOR3 VC_DIR_EXT_AL = _V(0.0, -sin(24.5*RAD), cos(24.5 * RAD))
- const UINT MESH_UNDEFINED = (UINT)-1
- const int MFDGROUPS [11]
- const double defaultStage1Guidance [2][8]
- const double LAUNCH_SITE [2] = {28.608, 34.581}
- const UINT AID_CUSTOM_PANELS_MIN = 10000
- const UINT AID_CUSTOM_PANELS_MAX = 30000
- const UINT AID_A7A3 = 11000
- const UINT AID_A8A3 = 11010
- const double SWITCH_HOLD_TIME = 0.5

8.5.1 Define Documentation

8.5.1.1 #define AID_A4 420

8.5.1.2 #define AID_A4_EVTTMR 421

8.5.1.3 #define AID_A4_MAX 429

8.5.1.4 #define AID_A4_METTMR1 422

8.5.1.5 #define AID_A4_METTMR2 423

8.5.1.6 #define AID_A4_MIN 420

8.5.1.7 #define AID_A8 430

8.5.1.8 #define AID_A8_MAX 459

8.5.1.9 #define AID_A8_MIN 430

8.5.1.10 #define AID_A8_TKBK1 431

8.5.1.11 #define AID_A8_TKBK10 440

8.5.1.12 #define AID_A8_TKBK11 441

8.5.1.13 #define AID_A8_TKBK12 442

8.5.1.14 #define AID_A8_TKBK13 443

8.5.1.15 #define AID_A8_TKBK14 444

8.5.1.16 #define AID_A8_TKBK15 445

8.5.1.17 #define AID_A8_TKBK16 446

8.5.1.18 #define AID_A8_TKBK17 447

8.5.1.19 #define AID_A8_TKBK2 432

8.5.1.20 #define AID_A8_TKBK3 433

8.5.1.21 #define AID_A8_TKBK4 434

8.5.1.22 #define AID_A8_TKBK5 435

8.5.1.23 #define AID_A8_TKBK6 436

8.5.1.24 #define AID_A8_TKBK7 437

8.5.1.25 #define AID_A8_TKBK8 438

8.5.1.26 #define AID_A8_TKBK9 439

8.5.1.27 #define AID_AFD_BRT 33

8.5.1.28 #define AID_AFD_BUTTONS 11

8.5.1.29 #define AID_AFD_PWR 22

8.5.1.30 #define AID_C2 280

YAW

ROLL

8.5.2.2 enum LATCH_STATE

Enumerator:

LT_LATCHED

LT_OPENING

LT_CLOSING

LT_RELEASED

8.5.2.3 enum OMS_REF

Enumerator:

OMS_LEFT

OMS_KIT

OMS_RIGHT

8.5.3 Variable Documentation

8.5.3.1 const int AftPitchThrusters[2] = {1, 3}

8.5.3.2 const int AftRollThrusters[2] = {9, 11}

8.5.3.3 const int AftRotThrusters[6] = {1, 3, 5, 7, 9, 11}

8.5.3.4 const int AftYawThrusters[2] = {5, 7}

8.5.3.5 const UINT AID_A7A3 = 11000

8.5.3.6 const UINT AID_A8A3 = 11010

8.5.3.7 const UINT AID_CUSTOM_PANELS_MAX = 30000

8.5.3.8 const UINT AID_CUSTOM_PANELS_MIN = 10000

8.5.3.9 const double AIR_DATA_PROBE_SPEED = 0.06666667

8.5.3.10 const double APU_FUEL_TANK_FLOWRATE[2] = {(3.25∗0.45359237)/60.0, (1.5∗0.45359237)/60.0}

8.5.3.11 const double APU_FUEL_TANK_MASS = 325.0∗0.45359237

8.5.3.12 const double ARM_DEPLOY_SPEED = 0.0294117647

8.5.3.13 const double ARM_EXTEND_SPEED = 0.142857

8.5.3.14 const double ARM_GRAPPLE_SPEED = 0.4

8.5.3.15 const double ARM_OPERATING_SPEED = 0.005

8.5.3.16 const double ARM_RIGID_SPEED = 0.25

8.5.3.17 const double ARM_TRANSLATE_SPEED = 0.1

8.5.3.18 const VECTOR3 ARM_WRIST_CAM_OFFSET = {-0.091886, 0.276656, 0.666001}

8.5.3.19 const double AXIS_TILT = 23.4458878∗RAD

8.5.3.20 const double BSM_ISP0 = 2400.0

8.5.3.21 const double BSM_MAX_PROPELLANT_MASS = 75.0 ∗ 0.45349

8.5.3.22 const double BSM_THRUST0 = 82.76475E3

8.5.3.23 const double CAM_HIGHRATE_SPEED = 12

8.5.3.24 const double CAM_LOWRATE_SPEED = 1.2

8.5.3.25 const char∗ DEFAULT_MESHNAME_COCKPIT = "SSU/Cockpit" `[static]`

8.5.3.26 const char∗ DEFAULT_MESHNAME_ET = "SSU/ET125" `[static]`

8.5.3.27 const char∗ DEFAULT_MESHNAME_KU = "SSU/KU" `[static]`

8.5.3.28 const char∗ DEFAULT_MESHNAME_LSRB = "SSU/LSRB" `[static]`

8.5.3.29 const char∗ DEFAULT_MESHNAME_MIDDECK = "SSU/MidDeck" `[static]`

```
{{0.0, 136.855, 219.456, 363.3216, 562.9656, 882.0912, 1236.8784, 1516.38},
                                                                                {90.0, 78.0, 68.0,
```

8.5.3.36 const double DOOR_OPERATING_SPEED = 0.007353

8.5.3.37 const double DOORLATCH_OPERATING_SPEED = 0.05

8.5.3.38 const double DYNP_MAX = 100e3

8.5.3.39 const double ET_SEP_RATE = 1.2192

8.5.3.40 const double ET_SEP_TIME = 18.00000

8.5.3.41 const double ET_UMB_DOOR_OPERATING_SPEED = 0.0416667

8.5.3.42 const double fps_to_ms = 0.3048

8.5.3.43 const double GEAR_MAX_DEPLOY_SPEED = 450/1.943844

8.5.3.44 const double GEAR_OPERATING_SPEED = 0.3

8.5.3.45 const double kg_to_pounds = 2.205

8.5.3.46 const double KU_OPERATING_SPEED = 0.0435

8.5.3.47 const double LATCH_OPERATING_SPEED = 0.166666667

8.5.3.48 const double LAUNCH_SITE[2] = {28.608, 34.581}

8.5.3.49 const double LBM = 0.45359237

8.5.3.50 const double MAX_ATT_LAUNCH = 1e-5

8.5.3.51 const double MAX_GRAPPLING_ANGLE = 0.087266

8.5.3.52 const double MAX_GRAPPLING_DIST = 0.5

8.5.3.53 const double MAX_ROLL_SRB = 2.5e-5

8.5.3.54 const UINT MESH_UNDEFINED = (UINT)-1

8.5.3.55 const int MFDGROUPS[11]

Initial value:

```
{
   GRP_CDR1_VC,GRP_CDR2_VC,GRP_PLT1_VC,GRP_PLT2_VC,
   GRP_MFD1_VC, GRP_MFD4_VC, GRP_MFD3_VC, GRP_MFD_aft_VC, GRP_MFD2_VC, GRP_MFD5_VC,
       NULL
   }
```

8.5.3.56 const UINT MIDX_COCKPIT = 2

8.5.3.57 const UINT MIDX_ET = 7

8.5.3.58 const UINT MIDX_KU_ANTENNA = 6

8.5.3.59 const UINT MIDX_LSRB = 8

8.5.3.60 const UINT MIDX_MIDDECK = 5

8.5.3.61 const UINT MIDX_ODS = 10

8.5.3.62 const UINT MIDX_ODS_VC = 11

8.5.3.63 const UINT MIDX_ORBITER = 1

8.5.3.64 const UINT MIDX_PLPANEL_A7 = 12

8.5.3.65 const UINT MIDX_PLPANEL_A8 = 13

8.5.3.66 const UINT MIDX_PLPANEL_A9 = 14

8.5.3.67 const UINT MIDX_RMS = 4

8.5.3.68 const UINT MIDX_RSRB = 9

8.5.3.69 const UINT MIDX_VC = 3

8.5.3.70 const double MPS2FPS = 3.280839895

8.5.3.71 const double NO_SLAG_TIME = 240.0

8.5.3.72 const int NosePitchThrusters[2] = {0, 2}

8.5.3.73 const int NoseRollThrusters[2] = {8, 10}

8.5.3.74 const int NoseRotThrusters[6] = {0, 2, 4, 6, 8, 10}

8.5.3.75 const int NoseYawThrusters[2] = {4, 6}

8.5.3.76 const VECTOR3 ODS_POS = _V(0.0, 0.0, 10.1529)

8.5.3.77 const char* ODSPANEL_MESHNAME = "SSU/ODSVC" `[static]`

8.5.3.78 const VECTOR3 OFS_LAUNCH_LEFTSRB = {-6.88,-1.91,-5.68 }

8.5.3.79 const VECTOR3 OFS_LAUNCH_ORBITER = { 0.0, 6.04,-7.989}

8.5.3.80 const VECTOR3 OFS_LAUNCH_RIGHTSRB = { 6.88,-1.91,-5.68 }

8.5.3.81 const VECTOR3 OFS_LAUNCH_TANK = { 0.0,-1.91, 5.72 }

8.5.3.82 const VECTOR3 OFS_MMU = {0,2.44,10.44}

8.5.3.83 const VECTOR3 OFS_WITHTANK_ORBITER = { 0.0, 4.658, 9.414}

8.5.3.84 const VECTOR3 OFS_WITHTANK_TANK = { 0.0,-3.34, 4.33 }

8.5.3.85 const VECTOR3 OFS_ZERO = { 0.0, 0.0, 0.0 }

8.5.3.86 const double OMS_HELIUM_TANK_MASS = 21.0

8.5.3.233 const VECTOR3 VC_DIR_AFTPILOT = _V(0, 0, -1)

8.5.3.234 const VECTOR3 VC_DIR_AFTWORKSTATION = _V(0.0, 0.0, -1.0)

8.5.3.235 const VECTOR3 VC_DIR_COAS = _V(0.0, 0.45399, -0.891007)

8.5.3.236 const VECTOR3 VC_DIR_EXT_AL = _V(0.0, -sin(24.5∗RAD), cos(24.5 ∗ RAD))

8.5.3.237 const VECTOR3 VC_DIR_MIDDECK = _V(0.0, -sin(24.5∗RAD), cos(24.5 ∗ RAD))

8.5.3.238 const VECTOR3 VC_DIR_MS1 = _V(0.0, 0, 1.0)

8.5.3.239 const VECTOR3 VC_DIR_MS2 = _V(0.0, 0.0, 1.0)

8.5.3.240 const VECTOR3 VC_DIR_PORTSTATION = _V(-cos(25.0 ∗ RAD), -sin(25.0 ∗ RAD), 0)

8.5.3.241 const VECTOR3 VC_DIR_RMSSTATION = _V(0.0, 0.0, -1.0)

8.5.3.242 const VECTOR3 VC_DIR_STBDSTATION = _V(cos(25.0 ∗ RAD), -sin(25.0 ∗ RAD), 0)

8.5.3.243 const int VC_DOCKCAM = 12

8.5.3.244 const int VC_EXT_AL = 200

8.5.3.245 const char∗ VC_LBL_AFTPILOT = "Aft pilot station" [static]

8.5.3.246 const char∗ VC_LBL_AFTWORKSTATION = "Aft Workstation" [static]

8.5.3.247 const char∗ VC_LBL_CDR = "Commander seat" [static]

8.5.3.248 const char∗ VC_LBL_COAS = "COAS" [static]

8.5.3.249 const char∗ VC_LBL_DOCKCAM = "ODS centerline camera" [static]

8.5.3.250 const char∗ VC_LBL_EXT_AL = "External Airlock" [static]

8.5.3.251 const char∗ VC_LBL_LEECAM = "RMS EE camera" [static]

8.5.3.252 const char∗ VC_LBL_MIDDECK = "Mid Deck" [static]

8.5.3.253 const char∗ VC_LBL_MS1 = "MS1 seat" [static]

8.5.3.254 const char∗ VC_LBL_MS2 = "MS2 seat" [static]

8.5.3.255 const char∗ VC_LBL_PLBCAMBL = "Payload bay BL camera" [static]

8.5.3.256 const char∗ VC_LBL_PLBCAMBR = "Payload bay BR camera" [static]

8.5.3.257 const char∗ VC_LBL_PLBCAMFL = "Payload bay FL camera" [static]

8.5.3.258 const char∗ VC_LBL_PLBCAMFR = "Payload bay FR camera" [static]

8.5.3.259 const char∗ VC_LBL_PLT = "Pilot seat" [static]

8.5.3.260 const char∗ VC_LBL_PORTSTATION = "Port workstation" [static]

8.5.3.261 const char∗ VC_LBL_RMSSTATION = "RMS work station" [static]

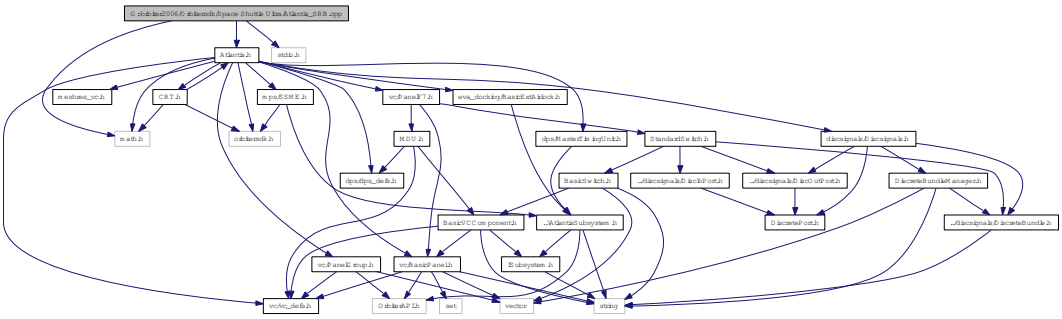
8.5.3.262 const char∗ VC_LBL_STBDSTATION = "Starboard workstation" [static]

8.5.3.263 const int VC_LEECAM = 17

8.6 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Atlantis_SRB.cpp File Reference

```
#include "Atlantis.h"
#include "math.h"
#include "stdio.h"
```

Include dependency graph for Atlantis_SRB.cpp:



Defines

- #define ORBITER_MODULE

Functions

- DLLCLBK VESSEL * ovcInit (OBJHANDLE hvessel, int flightmodel)
- DLLCLBK void ovcExit (VESSEL *vessel)

8.6.1 Define Documentation

8.6.1.1 #define ORBITER_MODULE

8.6.2 Function Documentation

8.6.2.1 DLLCLBK void ovcExit (VESSEL * *vessel*)

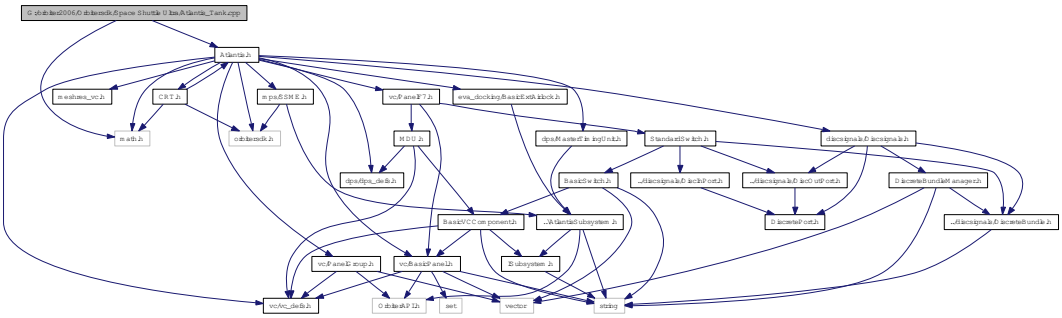
8.6.2.2 DLLCLBK VESSEL* ovcInit (OBJHANDLE *hvessel*, int *flightmodel*)

Here is the call graph for this function:

8.7 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Atlantis_-Tank.cpp File Reference

```
#include "Atlantis.h"
#include "math.h"
```

Include dependency graph for Atlantis_Tank.cpp:



Defines

- #define ORBITER_MODULE

Functions

- DLLCLBK VESSEL ∗ ovcInit (OBJHANDLE hvessel, int flightmodel)
- DLLCLBK void ovcExit (VESSEL ∗vessel)
- DLLCLBK void ovcSetState (VESSEL ∗vessel, const VESSELSTATUS ∗status)

8.7.1 Define Documentation

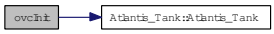
8.7.1.1 #define ORBITER_MODULE

8.7.2 Function Documentation

8.7.2.1 DLLCLBK void ovcExit (VESSEL ∗ *vessel*)

8.7.2.2 DLLCLBK VESSEL∗ ovcInit (OBJHANDLE *hvessel*, int *flightmodel*)

Here is the call graph for this function:

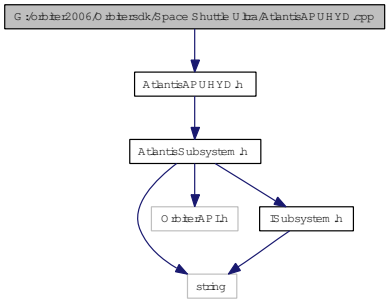


8.7.2.3 DLLCLBK void ovcSetState (VESSEL * *vessel*, const VESSELSTATUS * *status*)

8.8 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AtlantisAPUHYD.cpp File Reference

```
#include "AtlantisAPUHYD.h"
```

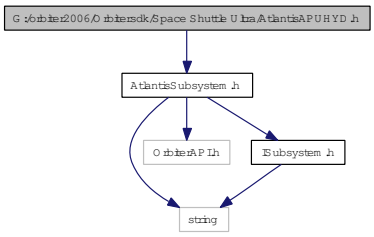
Include dependency graph for AtlantisAPUHYD.cpp:



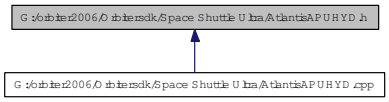
8.9 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AtlantisAPUHYD.h File Reference

`#include "AtlantisSubsystem.h"`

Include dependency graph for AtlantisAPUHYD.h:



This graph shows which files directly or indirectly include this file:



Classes

- class AtlantisAPUHYD

Defines

- #define AFX_ATLANTISAPUHYD_H__76669820_C41F_4F68_8755_0C199CFA6019__INCLUDED_

Typedefs

- typedef unsigned short APU_INDEX

Variables

- const APU_INDEX APU_1 = 0
- const APU_INDEX APU_2 = 1
- const APU_INDEX APU_3 = 2

8.9.1 Define Documentation

8.9.1.1 #define AFX_ATLANTISAPUHYD_H__76669820_C41F_4F68_8755_0C199CFA6019__-INCLUDED_

8.9.2 Typedef Documentation

8.9.2.1 typedef unsigned short APU_INDEX

8.9.3 Variable Documentation

8.9.3.1 const APU_INDEX APU_1 = 0

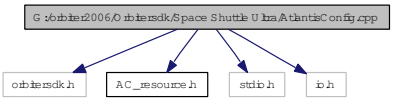
8.9.3.2 const APU_INDEX APU_2 = 1

8.9.3.3 const APU_INDEX APU_3 = 2

8.10 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AtlantisConfig.cpp File Reference

```
#include "orbitersdk.h"
#include "AC_resource.h"
#include <stdio.h>
#include <io.h>
```

Include dependency graph for AtlantisConfig.cpp:



Classes

- class AtlantisConfig

Defines

- #define STRICT 1
- #define ORBITER_MODULE

Functions

- DLLCLBK void InitModule (HINSTANCE hDLL)
- DLLCLBK void ExitModule (HINSTANCE hDLL)

Variables

- static char * tex_hires_enabled = "Textures2\\Atlantis"
- static char * tex_hires_disabled = "Textures2\\~Atlantis"
- static char * vcmsh_fname = "Meshes\\Atlantis\\AtlantisVC.msh"
- static char * msh_hires_bkup = "Meshes\\Atlantis\\~AtlantisVC_hi.msh"
- static char * msh_lores_bkup = "Meshes\\Atlantis\\~AtlantisVC_lo.msh"
- struct {
  HINSTANCE hInst
  AtlantisConfig * item
  } gParams

8.10.1 Define Documentation

8.10.1.1 #define ORBITER_MODULE

8.10.1.2 #define STRICT 1

8.10.2 Function Documentation

8.10.2.1 DLLCLBK void ExitModule (HINSTANCE *hDLL*)

8.10.2.2 DLLCLBK void InitModule (HINSTANCE *hDLL*)

8.10.3 Variable Documentation

8.10.3.1 struct { ... } gParams

8.10.3.2 HINSTANCE hInst

8.10.3.3 AtlantisConfig* item

8.10.3.4 char* msh_hires_bkup = "Meshes\\Atlantis\\~AtlantisVC_hi.msh" [static]

8.10.3.5 char* msh_lores_bkup = "Meshes\\Atlantis\\~AtlantisVC_lo.msh" [static]

8.10.3.6 char* tex_hires_disabled = "Textures2\\~Atlantis" [static]

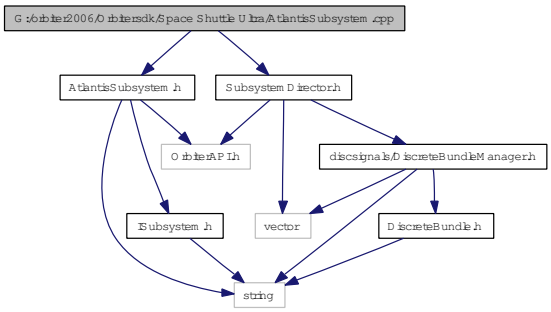
8.10.3.7 char* tex_hires_enabled = "Textures2\\Atlantis" [static]

8.10.3.8 char* vcmsh_fname = "Meshes\\Atlantis\\AtlantisVC.msh" [static]

8.11 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AtlantisSubsystem.cpp File Reference

```
#include "AtlantisSubsystem.h"
#include "SubsystemDirector.h"
```

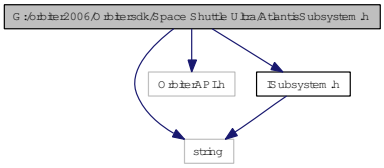
Include dependency graph for AtlantisSubsystem.cpp:



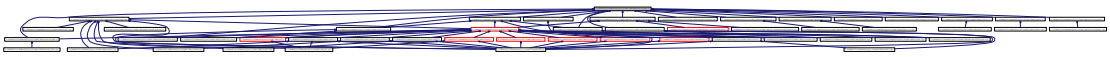
8.12 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AtlantisSubsystem.h File Reference

```
#include <string>
#include "OrbiterAPI.h"
#include "ISubsystem.h"
```

Include dependency graph for AtlantisSubsystem.h:



This graph shows which files directly or indirectly include this file:



Classes

- class AtlantisSubsystem

Defines

- #define AFX_ATLANTISSUBSYSTEM_H__6A9F0F48_D391_4E11_9536_F359182CA558__INCLUDED_

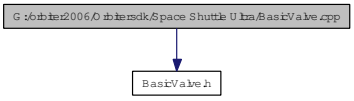
8.12.1 Define Documentation

8.12.1.1 #define AFX_ATLANTISSUBSYSTEM_H__6A9F0F48_D391_4E11_9536_F359182CA558__INCLUDED_

8.13 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/BasicValve.cpp File Reference

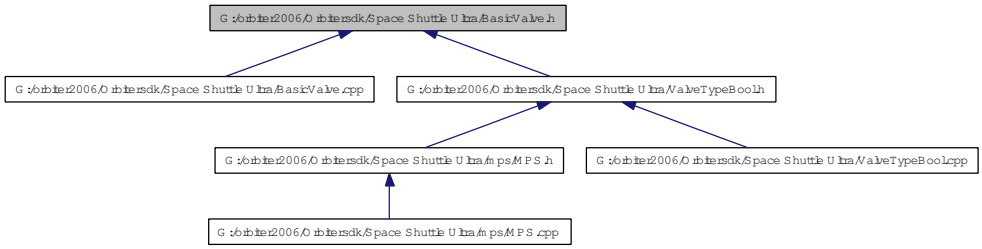
```
#include "BasicValve.h"
```

Include dependency graph for BasicValve.cpp:



8.14 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/BasicValve.h File Reference

This graph shows which files directly or indirectly include this file:



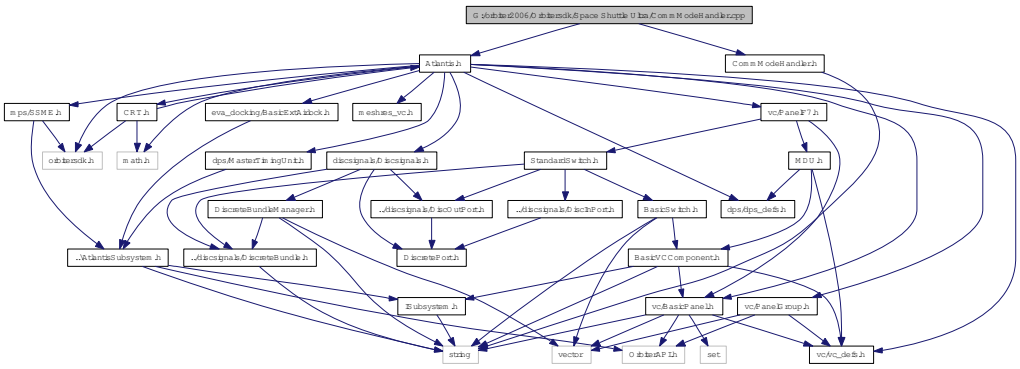
Classes

- class BasicValve

8.15 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/CommModeHandler.cpp File Reference

```
#include "Atlantis.h"
#include "CommModeHandler.h"
```

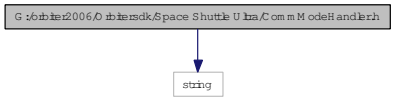
Include dependency graph for CommModeHandler.cpp:



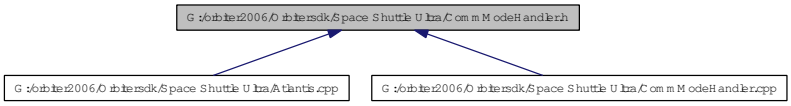
8.16 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/CommModeHandler.h File Reference

```
#include <string>
```

Include dependency graph for CommModeHandler.h:



This graph shows which files directly or indirectly include this file:



Classes

- class CommModeHandler

Enumerations

- enum COMM_ROLE { COMMROLE_CDR_PRELAUNCH, COMMROLE_CDR_ASCENT, COMMROLE_CDR_ORBIT }

8.16.1 Enumeration Type Documentation

8.16.1.1 enum COMM_ROLE

Enumerator:

COMMROLE_CDR_PRELAUNCH

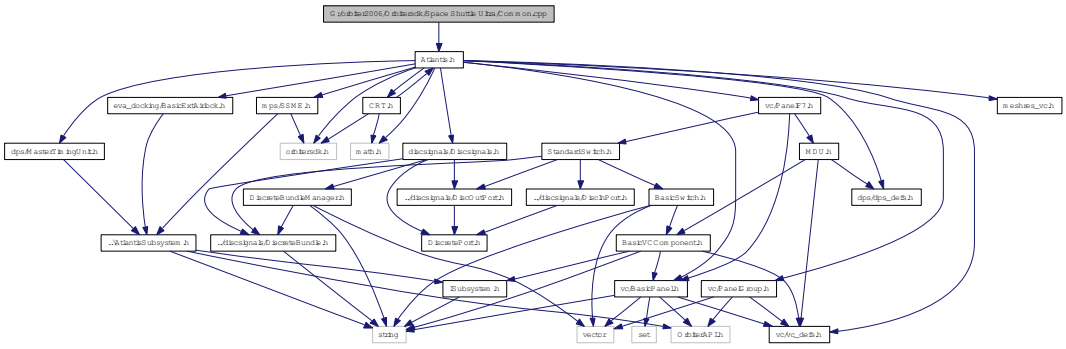
COMMROLE_CDR_ASCENT

COMMROLE_CDR_ORBIT

8.17 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Common.cpp File Reference

```
#include "Atlantis.h"
```

Include dependency graph for Common.cpp:



Functions

- void GetSRB_State (double met, double &thrust_level, double &prop_level)

Variables

- double SRB_Seq [SRB_nt] = {-SRB_STABILISATION_TIME,-0.001, 0, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90, 95, 100, 105, 110, 115, 120, 125, 130, 160}
- double SRB_Thrust [SRB_nt] = { 0.000, 0.000,0.955,0.971,0.992,0.988,1.000,0.915,0.856,0.810,0.779,0.750,0.705,0.721
- double SRB_Prop [SRB_nt]
- PARTICLESTREAMSPEC srb_contrail
- PARTICLESTREAMSPEC srb_exhaust
- PARTICLESTREAMSPEC srb_slag1
- PARTICLESTREAMSPEC srb_slag2
- PARTICLESTREAMSPEC srb_slag3
- PARTICLESTREAMSPEC bsm_exhaust
- PARTICLESTREAMSPEC bsm_contrail
- PARTICLESTREAMSPEC bsm_residual

8.17.1 Function Documentation

8.17.1.1 void GetSRB_State (double *met*, double & *thrust_level*, double & *prop_level*)

8.17.2 Variable Documentation

8.17.2.1 PARTICLESTREAMSPEC bsm_contrail

Initial value:

```
{
        0, 1.6, 40, 70.0, 0.1, 10, 8, 3.0, PARTICLESTREAMSPEC::DIFFUSE,
        PARTICLESTREAMSPEC::LVL_PSQRT, 0, 0.5,
        PARTICLESTREAMSPEC::ATM_PLOG, 1e-6, 0.1
}
```

8.17.2.2 PARTICLESTREAMSPEC bsm_exhaust

Initial value:

```
{
        0, 1.2, 5000, 70.0, 0.1, 0.2, 4, 3.0, PARTICLESTREAMSPEC::EMISSIVE,
        PARTICLESTREAMSPEC::LVL_FLAT, 1, 1,
        PARTICLESTREAMSPEC::ATM_FLAT, 1, 1
}
```

8.17.2.3 PARTICLESTREAMSPEC bsm_residual

Initial value:

```
{
        0, 0.3, 20, 5.0, 0.3, 10, 1, 1.5, PARTICLESTREAMSPEC::DIFFUSE,
        PARTICLESTREAMSPEC::LVL_PSQRT, 0, 0.5,
        PARTICLESTREAMSPEC::ATM_PLOG, 1e-6, 0.1
}
```

8.17.2.4 PARTICLESTREAMSPEC srb_contrail

Initial value:

```
{
        0, 17, 20, 150.0, 0.3, 10, 6, 1.5, PARTICLESTREAMSPEC::DIFFUSE,
        PARTICLESTREAMSPEC::LVL_PSQRT, 0, 0.5,
        PARTICLESTREAMSPEC::ATM_PLOG, 1e-6, 0.1
}
```

8.17.2.5 PARTICLESTREAMSPEC srb_exhaust

Initial value:

```
{
        0, 3.5, 5000, 100.0, 0.0, 1, 30, 0.0, PARTICLESTREAMSPEC::EMISSIVE,
        PARTICLESTREAMSPEC::LVL_FLAT, 1, 1,
        PARTICLESTREAMSPEC::ATM_FLAT, 1, 1
}
```

8.17.2.6 double SRB_Prop[SRB_nt]

Initial value:

```
{ 1,1,1.000,0.951,0.900,0.848,0.796,0.743,0.693,0.646,0.602,0.559,0.519,0.480,0.440,0.400,
                                0.357,0.314,0.270,0.228,0.188,0.150,0.114,0.080,0.049,0.020,0.007,0.002,0.
```

8.17.2.7 double SRB_Seq[SRB_nt] = {-SRB_STABILISATION_TIME,-0.001, 0, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90, 95, 100, 105, 110, 115, 120, 125, 130, 160}

8.17.2.8 PARTICLESTREAMSPEC srb_slag1

Initial value:

```
 {
        0, 0.1, SLAG1_RATE, 50.0, SLAG_SPREAD, 4, 0, 0.1, PARTICLESTREAMSPEC::EMISSIVE,
        PARTICLESTREAMSPEC::LVL_FLAT, 1, 1,
        PARTICLESTREAMSPEC::ATM_FLAT, 1, 1
}
```

8.17.2.9 PARTICLESTREAMSPEC srb_slag2

Initial value:

```
 {
        0, 0.05, SLAG2_RATE, 50.0, SLAG_SPREAD, 3, 0, 0.1, PARTICLESTREAMSPEC::EMISSIVE,
        PARTICLESTREAMSPEC::LVL_FLAT, 1, 1,
        PARTICLESTREAMSPEC::ATM_FLAT, 1, 1
}
```

8.17.2.10 PARTICLESTREAMSPEC srb_slag3

Initial value:

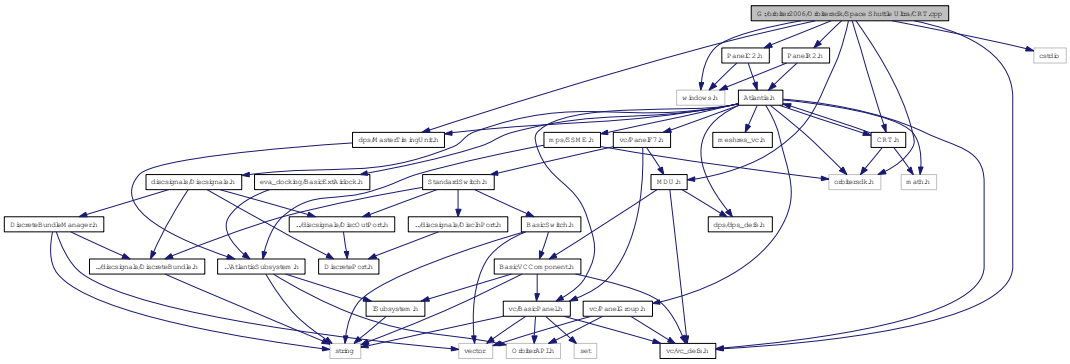
```
 {
        0, 0.01, SLAG3_RATE, 50.0, SLAG_SPREAD, 3, 0, 0.1, PARTICLESTREAMSPEC::EMISSIVE,
        PARTICLESTREAMSPEC::LVL_FLAT, 1, 1,
        PARTICLESTREAMSPEC::ATM_FLAT, 1, 1
}
```

8.17.2.11 double SRB_Thrust[SRB_nt] = { 0.000, 0.000,0.955,0.971,0.992,0.988,1.000,0.915,0.856,0.810,0.779,0.750,0.705,0.721,0.735,0.757,0.764,0.765,0.765,0.0

8.18 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/CRT.cpp File Reference

```
#include "windows.h"
#include "orbitersdk.h"
#include "PanelR2.h"
#include "PanelC2.h"
#include "vc/MDU.h"
#include "vc/vc_defs.h"
#include "CRT.h"
#include <cstdio>
#include "dps/MasterTimingUnit.h"
```

Include dependency graph for CRT.cpp:



Defines

- #define STRICT
- #define ORBITER_MODULE
- #define RED RGB(255, 0, 0)
- #define GREEN RGB(0, 255, 0)
- #define YELLOW RGB(255, 255, 0)
- #define DARK_YELLOW RGB(128, 128, 0)
- #define DARK_GREEN RGB(0, 128, 0)
- #define WHITE RGB(255, 255, 255)
- #define BLUE RGB(0, 0, 255)
- #define GRAY RGB(160, 160, 160)
- #define BRIGHTERGRAY RGB(200, 200, 200)
- #define BLACK RGB(0, 0, 0)

Functions

- DLLCLBK void InitModule (HINSTANCE hDLL)

- DLLCLBK void ExitModule (HINSTANCE hDLL)

- bool cbChangeOps (void *id, char *str, void *data)

- bool cbItem (void *id, char *str, void *data)

- bool cbSpec (void *id, char *str, void *data)

- bool cbItemData (void *id, char *str, void *data)

- bool cbSpecData (void *id, char *str, void *data)

Variables

- int g_MFDmode

- CRT * mfd

- MFD_GDIPARAM mfd_gparam

8.18.1 Define Documentation

8.18.1.1 #define BLACK RGB(0, 0, 0)

8.18.1.2 #define BLUE RGB(0, 0, 255)

8.18.1.3 #define BRIGHTERGRAY RGB(200, 200, 200)

8.18.1.4 #define DARK_GREEN RGB(0, 128, 0)

8.18.1.5 #define DARK_YELLOW RGB(128, 128, 0)

8.18.1.6 #define GRAY RGB(160, 160, 160)

8.18.1.7 #define GREEN RGB(0, 255, 0)

8.18.1.8 #define ORBITER_MODULE

8.18.1.9 #define RED RGB(255, 0, 0)

8.18.1.10 #define STRICT

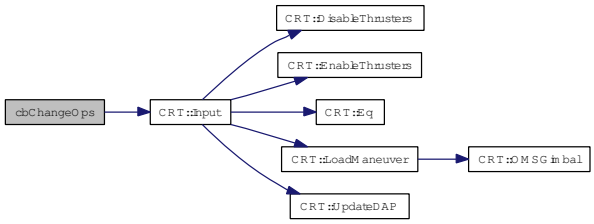
8.18.1.11 #define WHITE RGB(255, 255, 255)

8.18.1.12 #define YELLOW RGB(255, 255, 0)

8.18.2 Function Documentation

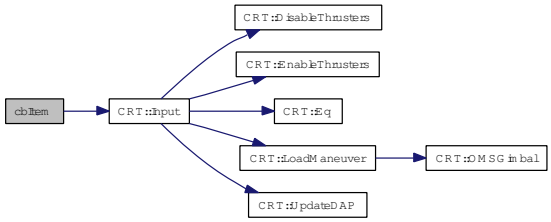
8.18.2.1 bool cbChangeOps (void ∗ *id*, char ∗ *str*, void ∗ *data*)

Here is the call graph for this function:



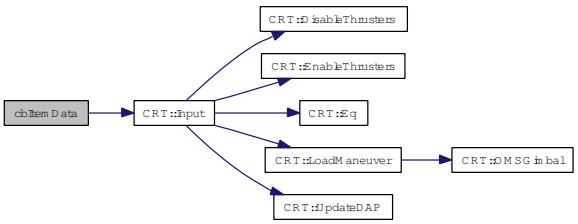
8.18.2.2 bool cbItem (void * *id*, char * *str*, void * *data*)

Here is the call graph for this function:



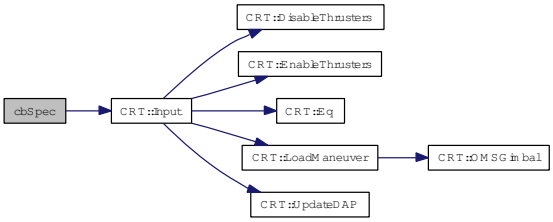
8.18.2.3 bool cbItemData (void * *id*, char * *str*, void * *data*)

Here is the call graph for this function:



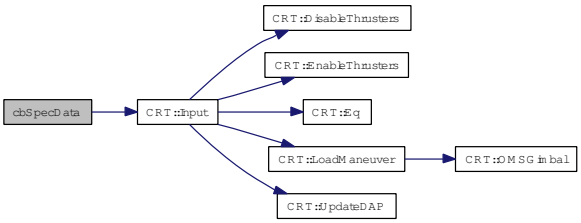
8.18.2.4 bool cbSpec (void * *id*, char * *str*, void * *data*)

Here is the call graph for this function:



8.18.2.5 bool cbSpecData (void ∗ *id*, char ∗ *str*, void ∗ *data*)

Here is the call graph for this function:



8.18.2.6 DLLCLBK void ExitModule (HINSTANCE *hDLL*)

8.18.2.7 DLLCLBK void InitModule (HINSTANCE *hDLL*)

Here is the call graph for this function:

8.18.3 Variable Documentation

8.18.3.1 int g_MFDmode

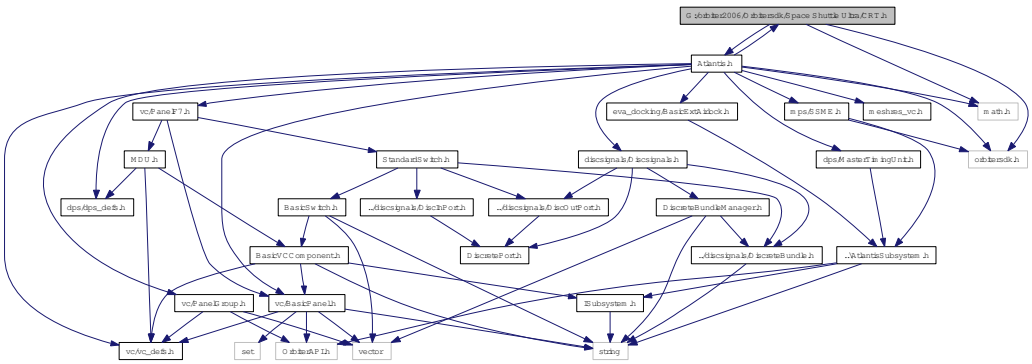
8.18.3.2 CRT∗ mfd

8.18.3.3 MFD_GDIPARAM mfd_gparam

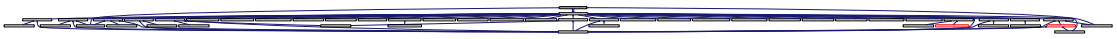
8.19 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/CRT.h File Reference

```
#include "Atlantis.h"
#include "orbitersdk.h"
#include <math.h>
```

Include dependency graph for CRT.h:



This graph shows which files directly or indirectly include this file:



Classes

- struct MFD_GDIPARAM
- class CRT
- struct CRT::SavePrm

Variables

- static const char * STD_CLASS_NAME = "SpaceShuttleUltra"
- const int convert [69]

8.19.1 Variable Documentation

8.19.1.1 const int convert[69]

Initial value:

```
 {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2}
```

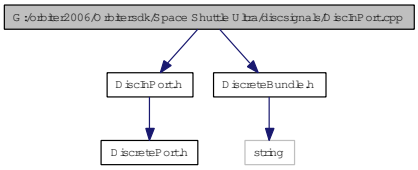
8.19.1.2 const char* STD_CLASS_NAME = "SpaceShuttleUltra" [static]

8.20 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscInPort.cpp File Reference

```
#include "DiscInPort.h"
```

```
#include "DiscreteBundle.h"
```

Include dependency graph for DiscInPort.cpp:



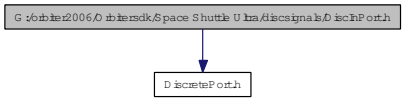
Namespaces

- namespace discsignals

8.21 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscInPort.h File Reference

```
#include "DiscretePort.h"
```

Include dependency graph for DiscInPort.h:



This graph shows which files directly or indirectly include this file:

Namespaces

- namespace discsignals

Classes

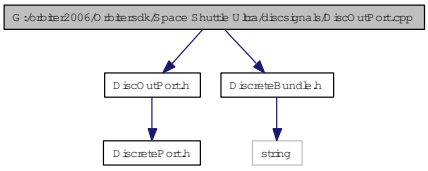
- class discsignals::DiscInPort

8.22 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscOutPort.cpp File Reference

```
#include "DiscOutPort.h"
```

```
#include "DiscreteBundle.h"
```

Include dependency graph for DiscOutPort.cpp:



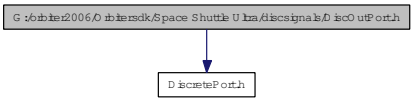
Namespaces

- namespace discsignals

8.23 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscOutPort.h File Reference

```
#include "DiscretePort.h"
```

Include dependency graph for DiscOutPort.h:



This graph shows which files directly or indirectly include this file:

Namespaces

- namespace discsignals

Classes

- class discsignals::DiscOutPort

Defines

- #define AFX_DISCOUTPORT_H__B01F0E2B_BD2E_4A08_B236_7DCF6C1507CD__INCLUDED_

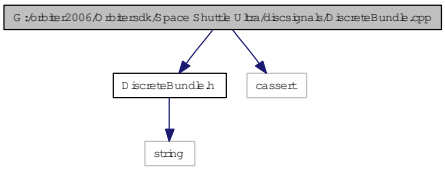
8.23.1 Define Documentation

8.23.1.1 #define AFX_DISCOUTPORT_H__B01F0E2B_BD2E_4A08_B236_7DCF6C1507CD__INCLUDED_

8.24 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscreteBundle.cpp File Reference

```
#include "DiscreteBundle.h"
#include <cassert>
```

Include dependency graph for DiscreteBundle.cpp:



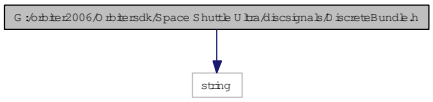
Namespaces

- namespace discsignals

8.25 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscreteBundle.h File Reference

```
#include <string>
```

Include dependency graph for DiscreteBundle.h:



This graph shows which files directly or indirectly include this file:

Namespaces

- namespace discsignals

Classes

- class discsignals::DiscreteBundle

Defines

- #define AFX_DISCRETEBUNDLE_H__A928D10B_004B_495D_87CD_3A0309B3EFDA__INCLUDED_

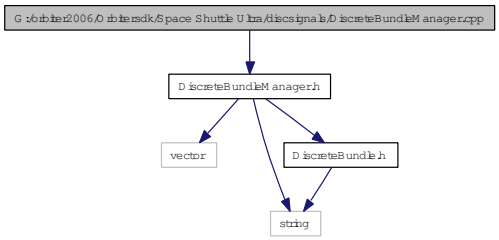
8.25.1 Define Documentation

8.25.1.1 #define AFX_DISCRETEBUNDLE_H__A928D10B_004B_495D_87CD_3A0309B3EFDA__INCLUDED_

8.26 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscreteBundleManager.cpp File Reference

```
#include "DiscreteBundleManager.h"
```

Include dependency graph for DiscreteBundleManager.cpp:



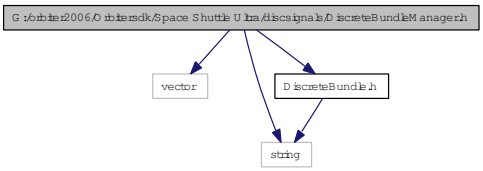
Namespaces

- namespace discsignals

8.27 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscreteBundleManager.h File Reference

```
#include <vector>
#include <string>
#include "DiscreteBundle.h"
```

Include dependency graph for DiscreteBundleManager.h:



This graph shows which files directly or indirectly include this file:

Namespaces

- namespace discsignals

Classes

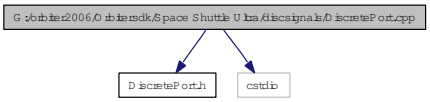
- class discsignals::DiscreteBundleManager

8.28 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscretePort.cpp File Reference

```
#include "DiscretePort.h"
```

```
#include <cstdio>
```

Include dependency graph for DiscretePort.cpp:

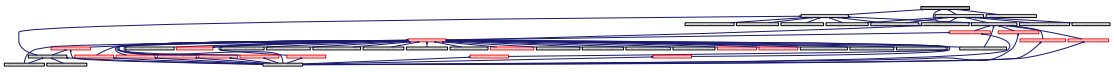


Namespaces

- namespace discsignals

8.29 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscretePort.h File Reference

This graph shows which files directly or indirectly include this file:



Namespaces

- namespace discsignals

Classes

- class discsignals::DiscretePort

Defines

- #define AFX_DISCRETEPORT_H__18D00C82_9652_493D_8A1D_D1CFF223A0B8__INCLUDED_

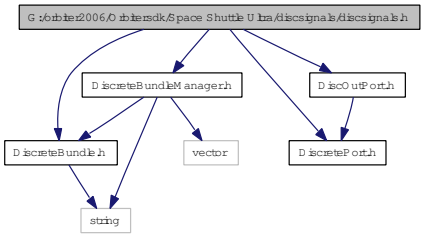
8.29.1 Define Documentation

8.29.1.1 #define AFX_DISCRETEPORT_H__18D00C82_9652_493D_8A1D_D1CFF223A0B8__INCLUDED_

8.30 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/discsignals.h File Reference

```
#include "DiscreteBundle.h"
#include "DiscreteBundleManager.h"
#include "DiscretePort.h"
#include "DiscOutPort.h"
```

Include dependency graph for discsignals.h:

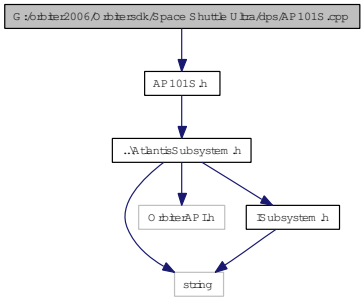


This graph shows which files directly or indirectly include this file:

8.31 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/AP101S.cpp File Reference

```
#include "AP101S.h"
```

Include dependency graph for AP101S.cpp:



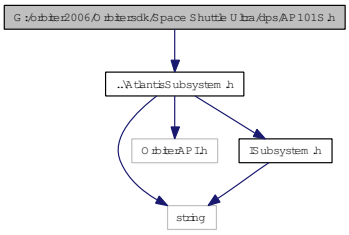
Namespaces

- namespace dps

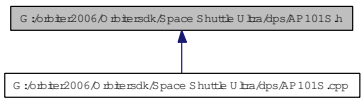
8.32 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/AP101S.h File Reference

`#include "..\AtlantisSubsystem.h"`

Include dependency graph for AP101S.h:



This graph shows which files directly or indirectly include this file:



Namespaces

- namespace dps

Classes

- class dps::AP101S

Defines

- #define AFX_AP101S_H__01BE9789_9E06_4F6B_9FAD_F48EA99C5161__INCLUDED_

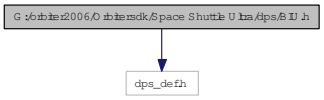
8.32.1 Define Documentation

8.32.1.1 #define AFX_AP101S_H__01BE9789_9E06_4F6B_9FAD_F48EA99C5161__INCLUDED_

8.33 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/BIU.h File Reference

```
#include "dps_def.h"
```

Include dependency graph for BIU.h:



Namespaces

- namespace dps

Classes

- class dps::BIU

8.34 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/dps_defs.h File Reference

This graph shows which files directly or indirectly include this file:

Namespaces

- namespace dps

Classes

- struct dps::DEUCHAR

Enumerations

- enum dps::MAJORFUNCTION { dps::GNC = 0, dps::SM = 1, dps::PL = 2 }

8.35 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/GPCVariables.h File Reference

Defines

- #define NUM_GPC_VARIABLES 32768
- #define GPCV_OP_CURRENT 0
- #define GPCV_OP_NEXT 1
- #define GPCV_MET_HIGH 2
- #define GPCV_MET_LOW 3
- #define GPCV_GMT_HIGH 4
- #define GPCV_GMT_LOW 5

8.35.1 Define Documentation

8.35.1.1 #define GPCV_GMT_HIGH 4

8.35.1.2 #define GPCV_GMT_LOW 5

8.35.1.3 #define GPCV_MET_HIGH 2

8.35.1.4 #define GPCV_MET_LOW 3

8.35.1.5 #define GPCV_OP_CURRENT 0

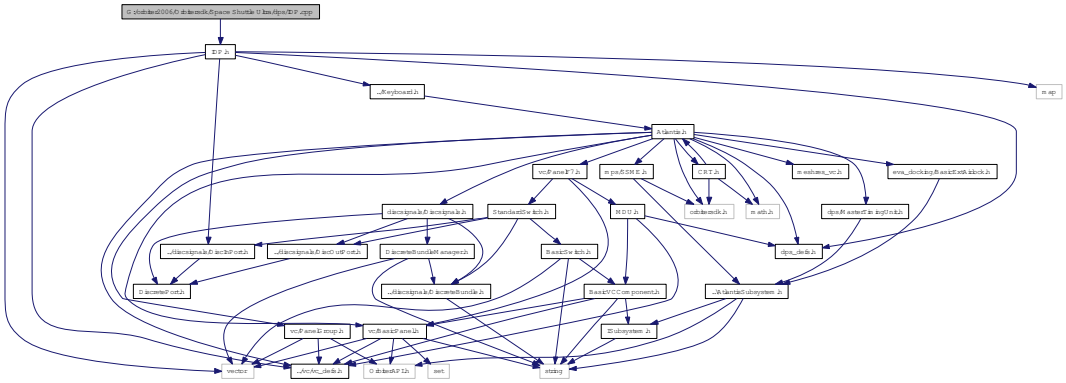
8.35.1.6 #define GPCV_OP_NEXT 1

8.35.1.7 #define NUM_GPC_VARIABLES 32768

8.36 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/IDP.cpp File Reference

```
#include "IDP.h"
```

Include dependency graph for IDP.cpp:



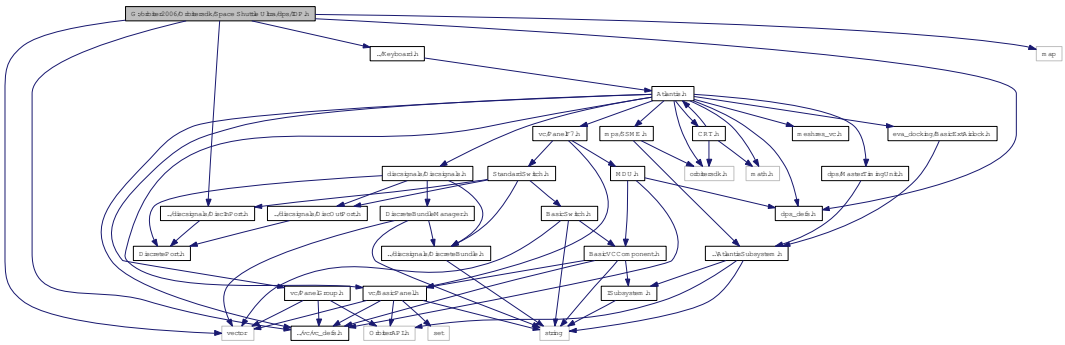
Namespaces

- namespace dps

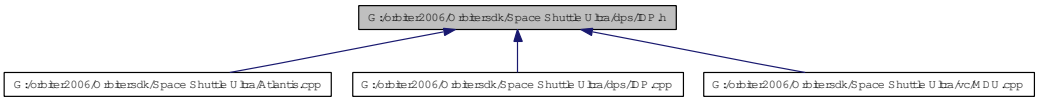
8.37 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/IDP.h File Reference

```
#include <vector>
#include "../vc/vc_defs.h"
#include "dps_defs.h"
#include "../Keyboard.h"
#include "../discsignals/DiscInPort.h"
#include <map>
```

Include dependency graph for IDP.h:



This graph shows which files directly or indirectly include this file:



Namespaces

- namespace dps

Classes

- class dps::IDPSoftware
- class dps::IDP

Variables

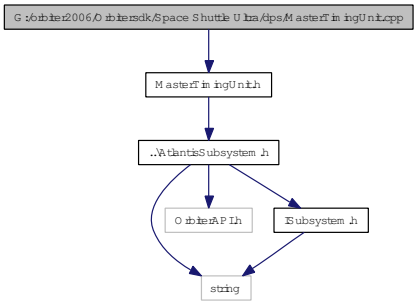
- const char dps::DEUATT_NORMAL = 0

- const char dps::DEUATT_OVERBRIGHT = 1
- const char dps::DEUATT_FLASHING = 2
- const char dps::DEUATT_UNDERLINED = 4

8.38 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/MasterTimingUnit.cpp File Reference

```
#include "MasterTimingUnit.h"
```

Include dependency graph for MasterTimingUnit.cpp:



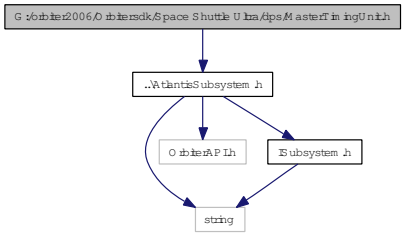
Namespaces

- namespace dps

8.39 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/MasterTimingUnit.h File Reference

`#include "..\AtlantisSubsystem.h"`

Include dependency graph for MasterTimingUnit.h:



This graph shows which files directly or indirectly include this file:

Namespaces

- namespace dps

Classes

- class dps::MasterTimingUnit

Defines

- #define AFX_MASTERTIMINGUNIT_H__FEF22983_033D_412B_B3A9_876EE3A3EC04__INCLUDED_

Typedefs

- typedef short dps::MTU_ACCU_INDEX
- typedef short dps::MTU_EVTTMR_INDEX

Enumerations

- enum dps::EVENT_TIMER_MODE { dps::COUNT_UP, dps::COUNT_DOWN, dps::COUNT_TEST }
- enum dps::EVENT_TIMER_CONTROL { dps::COUNT_STOPPED, dps::COUNT_STARTED }

Variables

- const MTU_EVTTMR_INDEX dps::TIMER_FORWARD = 0
- const MTU_EVTTMR_INDEX dps::TIMER_AFT = 1

8.39.1 Define Documentation

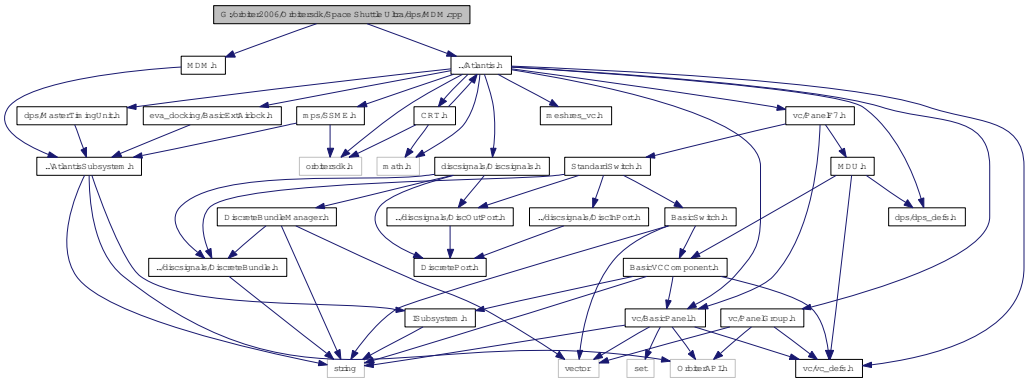
8.39.1.1 #define AFX_MASTERTIMINGUNIT_H__FEF22983_033D_412B_B3A9_-876EE3A3EC04__INCLUDED_

8.40 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/MDM.cpp File Reference

```
#include "../Atlantis.h"
```

```
#include "MDM.h"
```

Include dependency graph for MDM.cpp:



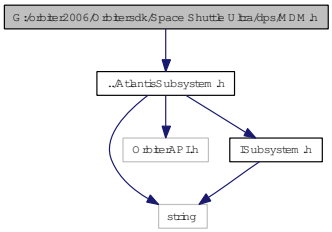
Namespaces

- namespace dps

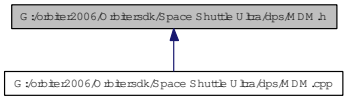
8.41 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/MDM.h File Reference

`#include "../AtlantisSubsystem.h"`

Include dependency graph for MDM.h:



This graph shows which files directly or indirectly include this file:



Namespaces

- namespace dps

Classes

- class dps::MDM

Defines

- #define AFX_MDM_H__BC1B13B9_53D9_4DC9_A6D6_2E7EA9F155A6__INCLUDED_

8.41.1 Define Documentation

8.41.1.1 #define AFX_MDM_H__BC1B13B9_53D9_4DC9_A6D6_2E7EA9F155A6__INCLUDED_

8.42 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/RAM16.h File Reference

Namespaces

- namespace dps

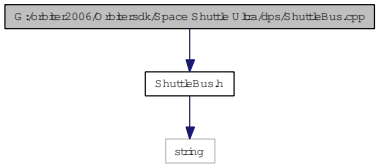
Classes

- class dps::RAM16

8.43 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/ShuttleBus.cpp File Reference

```
#include "ShuttleBus.h"
```

Include dependency graph for ShuttleBus.cpp:



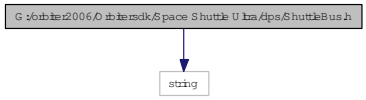
Namespaces

- namespace dps

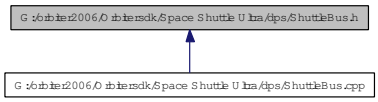
8.44 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/ShuttleBus.h File Reference

```
#include <string>
```

Include dependency graph for ShuttleBus.h:



This graph shows which files directly or indirectly include this file:



Namespaces

- namespace dps

Classes

- class dps::ShuttleBus

Defines

- #define AFX_SHUTTLEBUS_H__5E15D9D6_08AF_49BD_96FB_2835793F608E__INCLUDED_

8.44.1 Define Documentation

8.44.1.1 #define AFX_SHUTTLEBUS_H__5E15D9D6_08AF_49BD_96FB_2835793F608E__INCLUDED_

8.45 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/eps/eps_defs.h File Reference

Namespaces

- namespace eps

Classes

- class eps::IDCElement
- class eps::IACElement

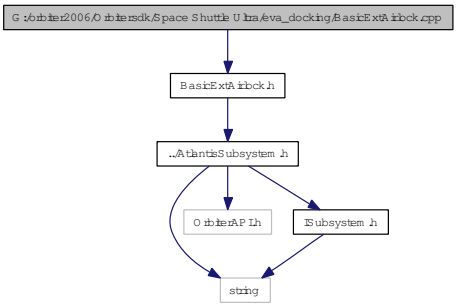
Enumerations

- enum eps::ACPHASE { eps::PHASE1 = 0, eps::PHASE2, eps::PHASE3 }

8.46 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/eva_docking/BasicExtAirlock.cpp File Reference

```
#include "BasicExtAirlock.h"
```

Include dependency graph for BasicExtAirlock.cpp:



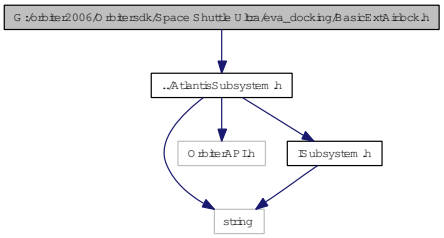
Namespaces

- namespace eva_docking

8.47 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/eva_docking/BasicExtAirlock.h File Reference

```
#include "../AtlantisSubsystem.h"
```

Include dependency graph for BasicExtAirlock.h:



This graph shows which files directly or indirectly include this file:

Namespaces

- namespace eva_docking

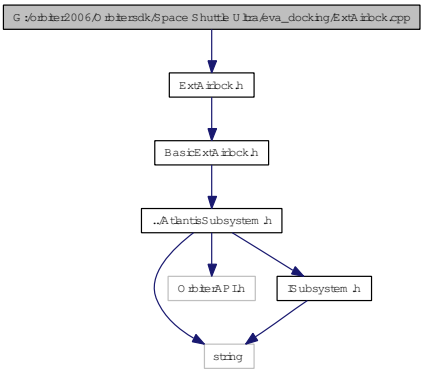
Classes

- class eva_docking::BasicExternalAirlock

8.48 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/eva_docking/ExtAirlock.cpp File Reference

```
#include "ExtAirlock.h"
```

Include dependency graph for ExtAirlock.cpp:



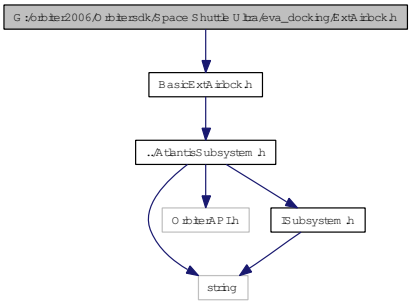
Namespaces

- namespace eva_docking

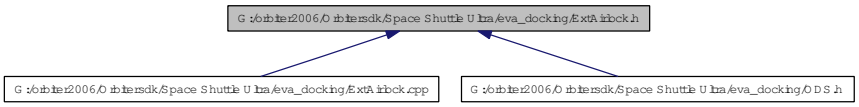
8.49 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/eva_docking/ExtAirlock.h File Reference

```
#include "BasicExtAirlock.h"
```

Include dependency graph for ExtAirlock.h:



This graph shows which files directly or indirectly include this file:



Namespaces

- namespace eva_docking

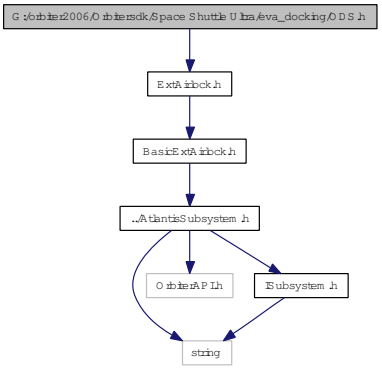
Classes

- class eva_docking::ExtAirlock

8.50 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/eva_docking/ODS.h File Reference

`#include "ExtAirlock.h"`

Include dependency graph for ODS.h:



Namespaces

- namespace eva_docking

Classes

- class eva_docking::ODS

Variables

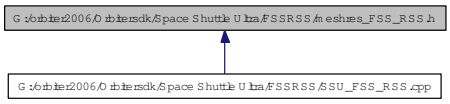
- static const char * ODS_MESHNAME = "ODS"

8.50.1 Variable Documentation

8.50.1.1 const char* ODS_MESHNAME = "ODS" [static]

8.51 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/FSSRSS/meshres_FSS_RSS.h File Reference

This graph shows which files directly or indirectly include this file:



Defines

- #define NGRP 189
- #define NMAT 6
- #define NTEX 2
- #define GRP_Box01 0
- #define GRP_Box02 1
- #define GRP_Box03 2
- #define GRP_Box04 3
- #define GRP_Box06 4
- #define GRP_Box07 5
- #define GRP_Box08 6
- #define GRP_Box09 7
- #define GRP_Box10 8
- #define GRP_Box11 9
- #define GRP_Box12 10
- #define GRP_Box13 11
- #define GRP_Box14 12
- #define GRP_Box15 13
- #define GRP_Box16 14
- #define GRP_Box17 15
- #define GRP_Box18 16
- #define GRP_Box19 17
- #define GRP_Box20 18
- #define GRP_Box21 19
- #define GRP_Box22 20
- #define GRP_Box23 21
- #define GRP_Box24 22
- #define GRP_Cylinder01 23
- #define GRP_Cylinder02 24
- #define GRP_Cylinder03 25
- #define GRP_Cylinder04 26
- #define GRP_Cylinder05 27
- #define GRP_Cylinder06 28
- #define GRP_Cylinder07 29
- #define GRP_Cylinder08 30
- #define GRP_Cylinder09 31

- #define GRP_Cylinder10 32
- #define GRP_Cylinder11 33
- #define GRP_Cylinder12 34
- #define GRP_Cylinder13 35
- #define GRP_Cylinder14 36
- #define GRP_Cylinder15 37
- #define GRP_Cylinder16 38
- #define GRP_Cylinder17 39
- #define GRP_Cylinder18 40
- #define GRP_Cylinder19 41
- #define GRP_Cylinder20 42
- #define GRP_Cylinder21 43
- #define GRP_Cylinder22 44
- #define GRP_Cylinder23 45
- #define GRP_Cylinder24 46
- #define GRP_Cylinder25 47
- #define GRP_Cylinder26 48
- #define GRP_Cylinder27 49
- #define GRP_Cylinder28 50
- #define GRP_Cylinder29 51
- #define GRP_Cylinder30 52
- #define GRP_Cylinder31 53
- #define GRP_Cylinder32 54
- #define GRP_Cylinder33 55
- #define GRP_Cylinder34 56
- #define GRP_Cylinder35 57
- #define GRP_Cylinder36 58
- #define GRP_Cylinder37 59
- #define GRP_Cylinder38 60
- #define GRP_Cylinder39 61
- #define GRP_Cylinder40 62
- #define GRP_Cylinder41 63
- #define GRP_Cylinder42 64
- #define GRP_Cylinder43 65
- #define GRP_Cylinder44 66
- #define GRP_Cylinder45 67
- #define GRP_Cylinder46 68
- #define GRP_Cylinder47 69
- #define GRP_Cylinder48 70
- #define GRP_Cylinder49 71
- #define GRP_Cylinder50 72
- #define GRP_Cylinder51 73
- #define GRP_Cylinder52 74
- #define GRP_Cylinder53 75
- #define GRP_Cylinder54 76
- #define GRP_Cylinder55 77
- #define GRP_Cylinder56 78
- #define GRP_Cylinder57 79
- #define GRP_Cylinder58 80
- #define GRP_Cylinder59 81

- #define GRP_Cylinder60 82
- #define GRP_Cylinder61 83
- #define GRP_Cylinder62 84
- #define GRP_Cylinder63 85
- #define GRP_Cylinder64 86
- #define GRP_Cylinder65 87
- #define GRP_Cylinder66 88
- #define GRP_Cylinder67 89
- #define GRP_Cylinder68 90
- #define GRP_Cylinder69 91
- #define GRP_Cylinder70 92
- #define GRP_Cylinder71 93
- #define GRP_Cylinder72 94
- #define GRP_Cylinder73 95
- #define GRP_Cylinder74 96
- #define GRP_Cylinder75 97
- #define GRP_Cylinder76 98
- #define GRP_Cylinder77 99
- #define GRP_Cylinder78 100
- #define GRP_Cylinder79 101
- #define GRP_Cylinder80 102
- #define GRP_Cylinder81 103
- #define GRP_Cylinder82 104
- #define GRP_Cylinder83 105
- #define GRP_Cylinder84 106
- #define GRP_Cylinder85 107
- #define GRP_Lightning_mast 108
- #define GRP_Lightning_mast_foundation 109
- #define GRP_Line01 110
- #define GRP_MAPU_part_1 111
- #define GRP_MAPU_part_2 112
- #define GRP_WPS_panel 113
- #define GRP_FSS_WPS_Z_bracket 114
- #define GRP_Inner_FSS_WPS_panel_track 115
- #define GRP_Inner_WPS_panel 116
- #define GRP_Outer_WPS_panel 117
- #define GRP_Y_FSS_WPS_struts 118
- #define GRP_WPS_panels_Y_support_struts 119
- #define GRP_Main_FSS 120
- #define GRP_GOX_Cylinder06 121
- #define GRP_GOX_Cylinder342 122
- #define GRP_GOX_vent_arm_truss 123
- #define GRP_North_GN2_vent_pipe 124
- #define GRP_South_GN2_vent_pipe 125
- #define GRP_Vent_hood 126
- #define GRP_Cylinder86 127
- #define GRP_Cylinder122 128
- #define GRP_Cylinder88 129
- #define GRP_Cylinder89 130
- #define GRP_Cylinder121 131

- #define GRP_Cylinder120 132
- #define GRP_Cylinder119 133
- #define GRP_Cylinder118 134
- #define GRP_Cylinder117 135
- #define GRP_Cylinder116 136
- #define GRP_Cylinder115 137
- #define GRP_Cylinder114 138
- #define GRP_Cylinder112 139
- #define GRP_Cylinder113 140
- #define GRP_Cylinder111 141
- #define GRP_Cylinder110 142
- #define GRP_Cylinder109 143
- #define GRP_Cylinder108 144
- #define GRP_Cylinder123 145
- #define GRP_Box25 146
- #define GRP_Box26 147
- #define GRP_Box27 148
- #define GRP_Cylinder104 149
- #define GRP_Box28 150
- #define GRP_Cylinder105 151
- #define GRP_Box29 152
- #define GRP_Box05 153
- #define GRP_Cylinder106 154
- #define GRP_Box31 155
- #define GRP_Cylinder107 156
- #define GRP_Box32 157
- #define GRP_Box33 158
- #define GRP_Box34 159
- #define GRP_Box35 160
- #define GRP_Cylinder103 161
- #define GRP_Box36 162
- #define GRP_Box37 163
- #define GRP_Cylinder102 164
- #define GRP_Cylinder101 165
- #define GRP_Cylinder100 166
- #define GRP_Cylinder99 167
- #define GRP_Cylinder98 168
- #define GRP_Cylinder97 169
- #define GRP_Cylinder96 170
- #define GRP_Cylinder95 171
- #define GRP_Cylinder94 172
- #define GRP_Cylinder93 173
- #define GRP_Cylinder92 174
- #define GRP_Cylinder91 175
- #define GRP_Cylinder90 176
- #define GRP_Cylinder87 177
- #define GRP_Box30 178
- #define GRP_White_Room_Mat 179
- #define GRP_White_Room_Part_2 180
- #define GRP_White_Room_Part_1 181

- #define GRP_Orbiter_Access_Arm 182

- #define GRP_FWD_RSS_drive_truck 183

- #define GRP_Rear_RSS_drive_truck 184

- #define GRP_RSS 185

- #define GRP_RSS_cab 186

- #define GRP_RSS_misc 187

- #define GRP_RSS_sheild 188

8.51.1 Define Documentation

8.51.1.1 #define GRP_Box01 0

8.51.1.2 #define GRP_Box02 1

8.51.1.3 #define GRP_Box03 2

8.51.1.4 #define GRP_Box04 3

8.51.1.5 #define GRP_Box05 153

8.51.1.6 #define GRP_Box06 4

8.51.1.7 #define GRP_Box07 5

8.51.1.8 #define GRP_Box08 6

8.51.1.9 #define GRP_Box09 7

8.51.1.10 #define GRP_Box10 8

8.51.1.11 #define GRP_Box11 9

8.51.1.12 #define GRP_Box12 10

8.51.1.13 #define GRP_Box13 11

8.51.1.14 #define GRP_Box14 12

8.51.1.15 #define GRP_Box15 13

8.51.1.16 #define GRP_Box16 14

8.51.1.17 #define GRP_Box17 15

8.51.1.18 #define GRP_Box18 16

8.51.1.19 #define GRP_Box19 17

8.51.1.20 #define GRP_Box20 18

8.51.1.21 #define GRP_Box21 19

8.51.1.22 #define GRP_Box22 20

8.51.1.23 #define GRP_Box23 21

8.51.1.24 #define GRP_Box24 22

8.51.1.25 #define GRP_Box25 146

8.51.1.26 #define GRP_Box26 147

8.51.1.27 #define GRP_Box27 148

8.51.1.28 #define GRP_Box28 150

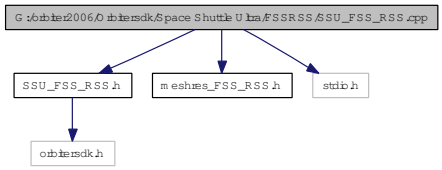
8.51.1.29 #define GRP_Box29 152

8.51.1.30 #define GRP_Box30 178

8.52 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/FSSRSS/SSU_-FSS_RSS.cpp File Reference

```
#include "SSU_FSS_RSS.h"
#include "meshres_FSS_RSS.h"
#include <stdio.h>
```

Include dependency graph for SSU_FSS_RSS.cpp:



Defines

- #define STRICT

Functions

- DLLCLBK VESSEL * ovcInit (OBJHANDLE hvessel, int flightmodel)
- DLLCLBK void ovcExit (VESSEL *vessel)
- DLLCLBK void ovcSetClassCaps (VESSEL *vessel, FILEHANDLE cfg)
- DLLCLBK void ovcLoadStateEx (VESSEL *vessel, FILEHANDLE scn, void *vs)
- DLLCLBK void ovcSaveState (VESSEL *vessel, FILEHANDLE scn)
- DLLCLBK void ovcTimestep (VESSEL *vessel, double simt)
- DLLCLBK int ovcConsumeBufferedKey (VESSEL *vessel, DWORD key, bool down, char *kstate)

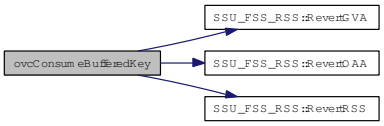
8.52.1 Define Documentation

8.52.1.1 #define STRICT

8.52.2 Function Documentation

8.52.2.1 DLLCLBK int ovcConsumeBufferedKey (VESSEL ∗ *vessel*, DWORD *key*, bool *down*, char ∗ *kstate*)

Here is the call graph for this function:



8.52.2.2 DLLCLBK void ovcExit (VESSEL ∗ *vessel*)

8.52.2.3 DLLCLBK VESSEL∗ ovcInit (OBJHANDLE *hvessel*, int *flightmodel*)

Here is the call graph for this function:

8.52.2.4 DLLCLBK void ovcLoadStateEx (VESSEL ∗ *vessel*, FILEHANDLE *scn*, void ∗ *vs*)

8.52.2.5 DLLCLBK void ovcSaveState (VESSEL ∗ *vessel*, FILEHANDLE *scn*)

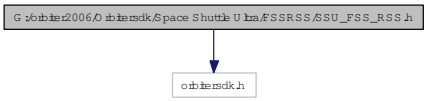
8.52.2.6 DLLCLBK void ovcSetClassCaps (VESSEL ∗ *vessel*, FILEHANDLE *cfg*)

8.52.2.7 DLLCLBK void ovcTimestep (VESSEL ∗ *vessel*, double *simt*)

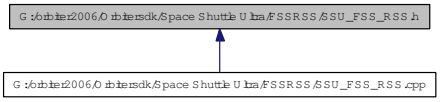
8.53 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/FSSRSS/SSU_-FSS_RSS.h File Reference

```
#include "orbitersdk.h"
```

Include dependency graph for SSU_FSS_RSS.h:



This graph shows which files directly or indirectly include this file:



Classes

- class SSU_FSS_RSS

Variables

- const double SSU_FSS_RSS_SIZE = 66
- const double SSU_FSS_RSS_EMPTY_MASS = 600000
- const double OAA_OPERATING_SPEED = 0.025
- const double GVA_OPERATING_SPEED = 0.022
- const double RSS_OPERATING_SPEED = 0.0004

8.53.1 Variable Documentation

8.53.1.1 const double GVA_OPERATING_SPEED = 0.022

8.53.1.2 const double OAA_OPERATING_SPEED = 0.025

8.53.1.3 const double RSS_OPERATING_SPEED = 0.0004

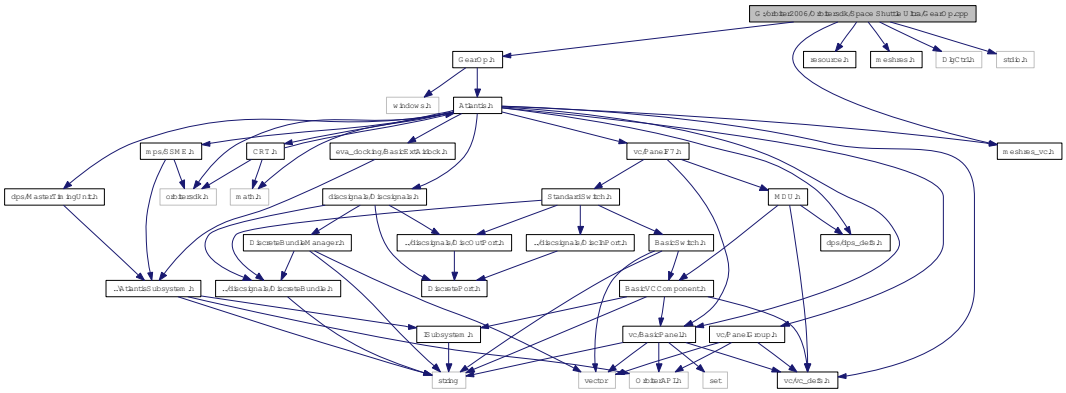
8.53.1.4 const double SSU_FSS_RSS_EMPTY_MASS = 600000

8.53.1.5 const double SSU_FSS_RSS_SIZE = 66

8.54 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/GearOp.cpp File Reference

```
#include "GearOp.h"
#include "resource.h"
#include "meshres.h"
#include "meshres_vc.h"
#include "DlgCtrl.h"
#include <stdio.h>
```

Include dependency graph for GearOp.cpp:



Variables

- GDIParams g_Param
- HELPCONTEXT g_hc
- char ∗ ActionString [5]

8.54.1 Variable Documentation

8.54.1.1 char∗ ActionString[5]

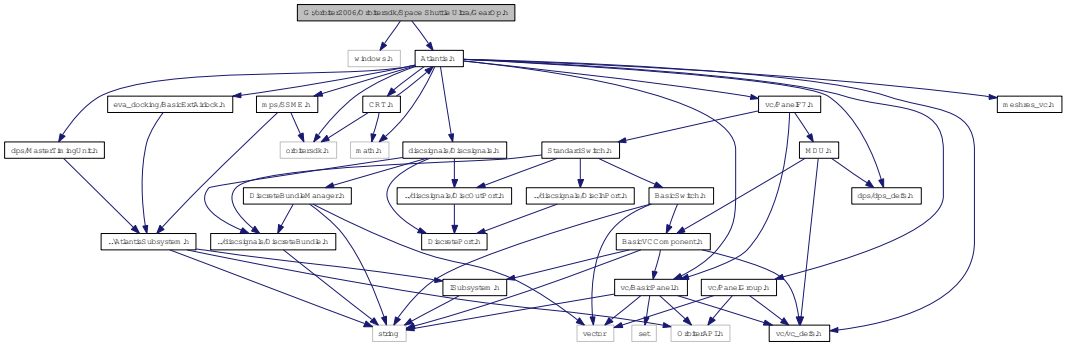
8.54.1.2 HELPCONTEXT g_hc

8.54.1.3 GDIParams g_Param

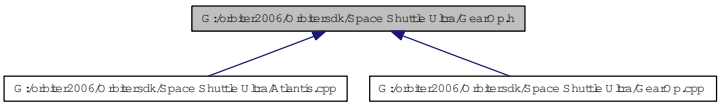
8.55 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/GearOp.h File Reference

```
#include <windows.h>
#include "Atlantis.h"
```

Include dependency graph for GearOp.h:



This graph shows which files directly or indirectly include this file:



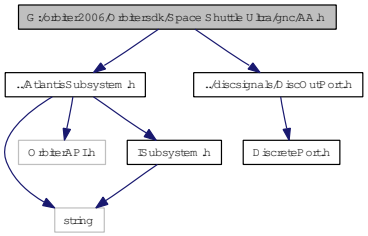
Classes

- class GearOp

8.56 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/gnc/AA.h File Reference

```
#include "../AtlantisSubsystem.h"
#include "../discsignals/DiscOutPort.h"
```

Include dependency graph for AA.h:



Namespaces

- namespace gnc

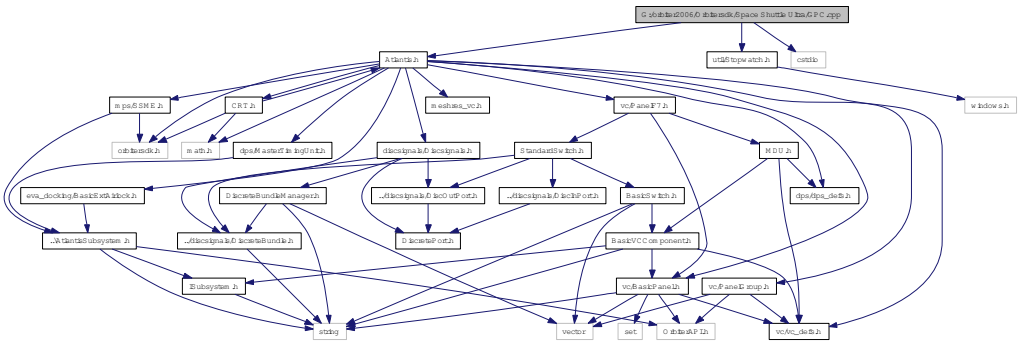
Classes

- class gnc::AccAssembly

8.57 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/GPC.cpp File Reference

```
#include "Atlantis.h"
#include "util/Stopwatch.h"
#include <cstdio>
```

Include dependency graph for GPC.cpp:



Functions

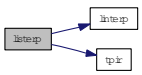
- int tpir (const double *list, int n_items, double target)
- double linterp (double x0, double y0, double x1, double y1, double x)
- double listerp (const double *listx, const double *listy, int n_items, double x)
- double tableterp (const double *table, const double *listrow, int n_row, const double *listcol, int n_col, double rowlookup, double collookup)

8.57.1 Function Documentation

8.57.1.1 double linterp (double *x0*, double *y0*, double *x1*, double *y1*, double *x*)

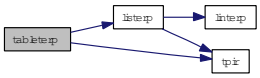
8.57.1.2 double listerp (const double * *listx*, const double * *listy*, int *n_items*, double *x*)

Here is the call graph for this function:



8.57.1.3 double tableterp (const double * *table*, const double * *listrow*, int *n_row*, const double * *listcol*, int *n_col*, double *rowlookup*, double *collookup*)

Here is the call graph for this function:

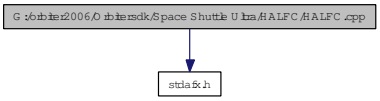


8.57.1.4 int tpir (const double * *list*, int *n_items*, double *target*)

8.58 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/HALFC/HALFC.cpp File Reference

```
#include "stdafx.h"
```

Include dependency graph for HALFC.cpp:



Functions

- int main (int argc, char *argv[ ])

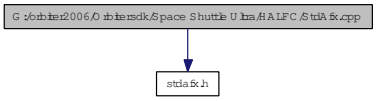
8.58.1 Function Documentation

8.58.1.1 int main (int *argc*, char * *argv*[ ])

8.59 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/HALFC/StdAfx.cpp File Reference

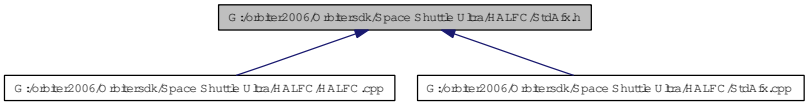
```
#include "stdafx.h"
```

Include dependency graph for StdAfx.cpp:



8.60 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/HALFC/StdAfx.h File Reference

This graph shows which files directly or indirectly include this file:



Defines

- #define AFX_STDAFX_H__C669E737_B06A_482C_B99B_41C360D81979__INCLUDED_

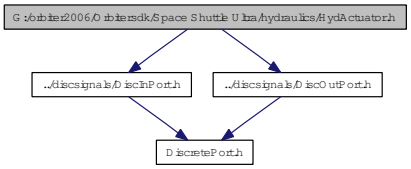
8.60.1 Define Documentation

8.60.1.1 #define AFX_STDAFX_H__C669E737_B06A_482C_B99B_41C360D81979__INCLUDED_

8.61 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/hydraulics/HydActuator.h File Reference

```
#include "../discsignals/DiscInPort.h"
#include "../discsignals/DiscOutPort.h"
```

Include dependency graph for HydActuator.h:



Namespaces

- namespace hydraulics

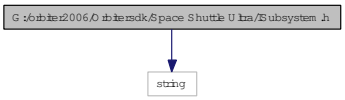
Classes

- class hydraulics::HydActuator

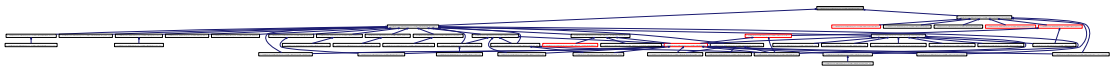
8.62 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/ISubsystem.h File Reference

```
#include <string>
```

Include dependency graph for ISubsystem.h:



This graph shows which files directly or indirectly include this file:



Classes

- class ISubsystem

Defines

- #define AFX_ISUBSYSTEM_H__A4896D4F_46E0_4D68_AEA4_F6B9CF259711__-INCLUDED_

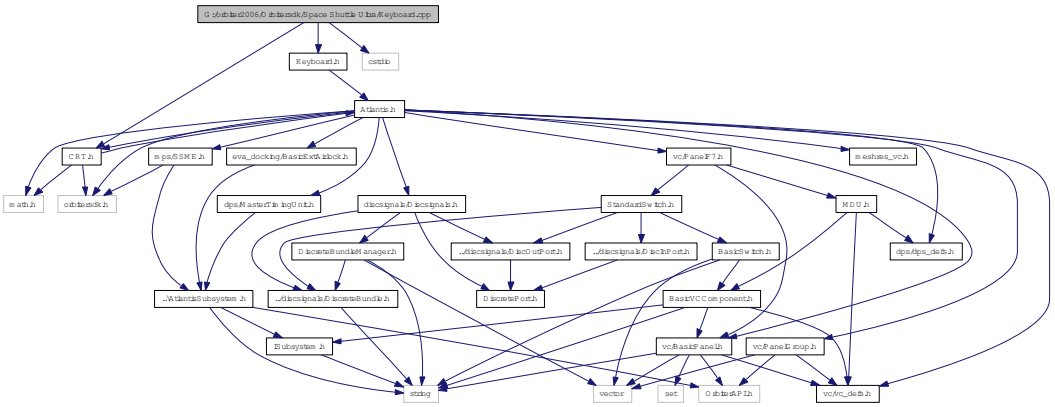
8.62.1 Define Documentation

8.62.1.1 #define AFX_ISUBSYSTEM_H__A4896D4F_46E0_4D68_AEA4_F6B9CF259711__-INCLUDED_

8.63 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Keyboard.cpp File Reference

```
#include "Keyboard.h"
#include "CRT.h"
#include <cstdio>
```

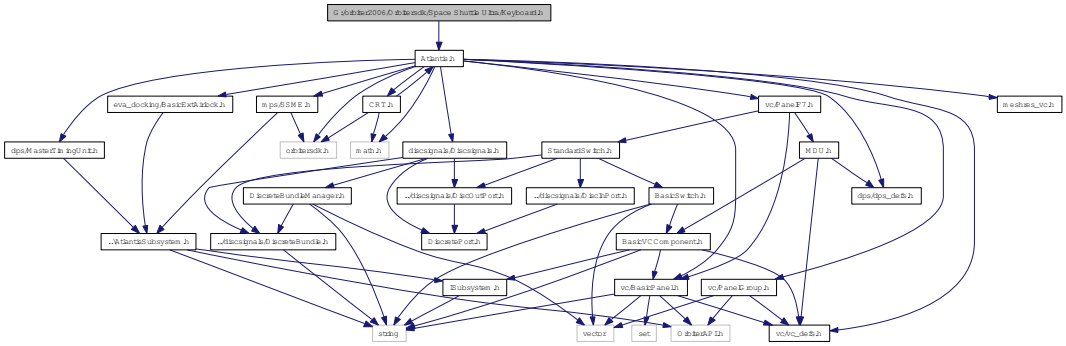
Include dependency graph for Keyboard.cpp:



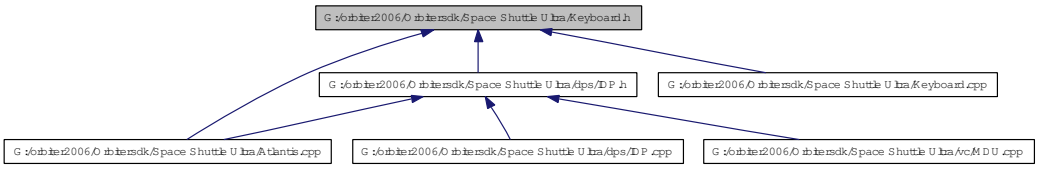
8.64 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Keyboard.h File Reference

```
#include "Atlantis.h"
```

Include dependency graph for Keyboard.h:



This graph shows which files directly or indirectly include this file:



Classes

- class Keyboard

Variables

- const int DEU_KEY_RESUME = 9

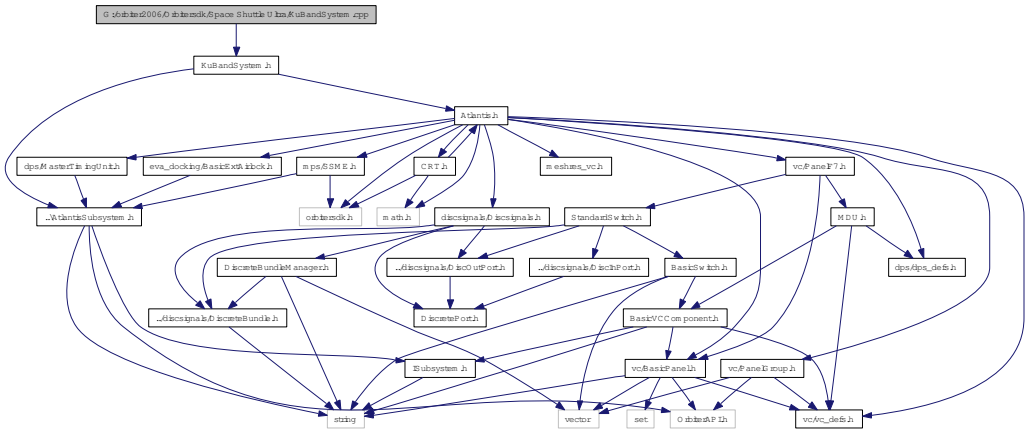
8.64.1 Variable Documentation

8.64.1.1 const int DEU_KEY_RESUME = 9

8.65 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/KuBandSystem.cpp File Reference

```
#include "KuBandSystem.h"
```

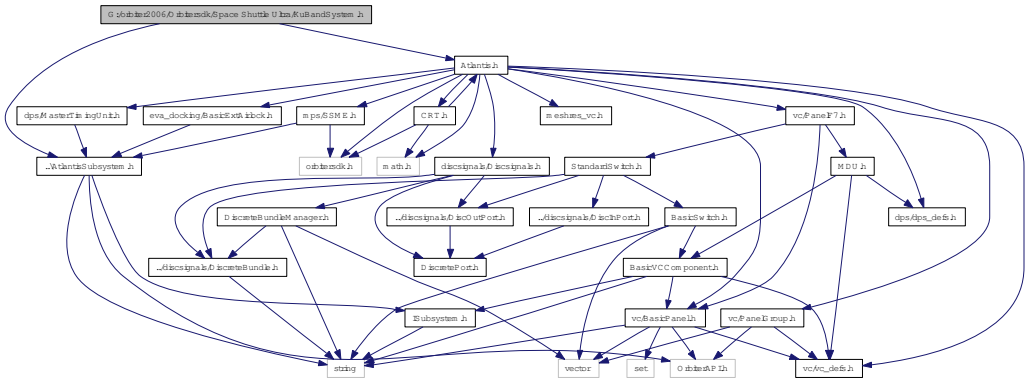
Include dependency graph for KuBandSystem.cpp:



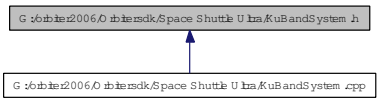
8.66 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/KuBandSystem.h File Reference

```
#include "Atlantis.h"
#include "AtlantisSubsystem.h"
```

Include dependency graph for KuBandSystem.h:



This graph shows which files directly or indirectly include this file:



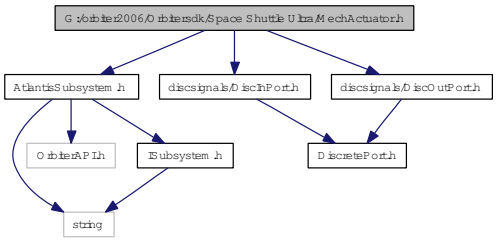
Classes

- class KuBandSystem

8.67 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/MechActuator.h File Reference

```
#include "AtlantisSubsystem.h"
#include "discsignals/DiscInPort.h"
#include "discsignals/DiscOutPort.h"
```

Include dependency graph for MechActuator.h:



Classes

- class MechActuator

8.68 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/meshres.h File Reference

This graph shows which files directly or indirectly include this file:

Defines

- #define NGRP 68
- #define NMAT 6
- #define NTEX 2
- #define GRP_FUSELAGE 0
- #define GRP_BODYFLAP 1
- #define GRP_CARGODOOROUTR 2
- #define GRP_CARGODOOROUTL 3
- #define GRP_FLAPR 4
- #define GRP_FLAPL 5
- #define GRP_AILERONR 6
- #define GRP_AILERONL 7
- #define GRP_ETUMBDOORR 8
- #define GRP_ETUMBDOORL 9
- #define GRP_OMSL 10
- #define GRP_OMSR 11
- #define GRP_SSMER 12
- #define GRP_SSMEL 13
- #define GRP_SSMET 14
- #define GRP_NOSEDOORL 15
- #define GRP_GEARDOORL 16
- #define GRP_GEARDOORR 17
- #define GRP_NOSEDOORR 18
- #define GRP_NOSEWHEEL 19
- #define GRP_NOSEGEAR 20
- #define GRP_WHEELR 21
- #define GRP_GEARR 22
- #define GRP_WHEELL 23
- #define GRP_GEARL 24
- #define GRP_TRACKBOX 25
- #define GRP_RUDDERR 26
- #define GRP_RUDDERL 27
- #define GRP_PROBEL 28
- #define GRP_PROBER 29
- #define GRP_CARGOBAY 30
- #define GRP_CARGODOORINR 31
- #define GRP_CARGODOORINL 32
- #define GRP_KUBAND1 33
- #define GRP_KUBAND2 34

- #define GRP_CAMERAPIVOTFL 35
- #define GRP_CAMERAFL 36
- #define GRP_CAMERABL 37
- #define GRP_CAMERAPIVOTBL 38
- #define GRP_CAMERAFR 39
- #define GRP_CAMERAPIVOTFR 40
- #define GRP_CAMERAPIVOTBR 41
- #define GRP_CAMERABR 42
- #define GRP_RADIATORFL 43
- #define GRP_RADIATORFR 44
- #define GRP_RADIATORBR 45
- #define GRP_RADIATORBL 46
- #define GRP_STARTRACKERS 47
- #define GRP_TRACKERPLATES 48
- #define GRP_ZSTARTRACKDOOR 49
- #define GRP_YSTARTRACKDOOR 50
- #define GRP_LEFTFLAPPERIN 51
- #define GRP_LEFTFLAPPEROUT 52
- #define GRP_RIGHTFLAPPERIN 53
- #define GRP_RIGHTFLAPPEROUT 54
- #define GRP_PORTPUSH_ROD 55
- #define GRP_PORT_CLAMP 56
- #define GRP_PORTDOOR_PULL 57
- #define GRP_STBDPUSH_ROD 58
- #define GRP_STBD_CLAMP 59
- #define GRP_STBDDOOR_PULL 60
- #define GRP_AFT_HOOKS 61
- #define GRP_MID_AFT_HOOKS 62
- #define GRP_MID_FWD_HOOKS 63
- #define GRP_FWD_HOOKS 64
- #define GRP_HUDL 65
- #define GRP_HUDR 66
- #define GRP_WINDOWS 67

8.68.1 Define Documentation

8.68.1.1 #define GRP_AFT_HOOKS 61

8.68.1.2 #define GRP_AILERONL 7

8.68.1.3 #define GRP_AILERONR 6

8.68.1.4 #define GRP_BODYFLAP 1

8.68.1.5 #define GRP_CAMERABL 37

8.68.1.6 #define GRP_CAMERABR 42

8.68.1.7 #define GRP_CAMERAFL 36

8.68.1.8 #define GRP_CAMERAFR 39

8.68.1.9 #define GRP_CAMERAPIVOTBL 38

8.68.1.10 #define GRP_CAMERAPIVOTBR 41

8.68.1.11 #define GRP_CAMERAPIVOTFL 35

8.68.1.12 #define GRP_CAMERAPIVOTFR 40

8.68.1.13 #define GRP_CARGOBAY 30

8.68.1.14 #define GRP_CARGODOORINL 32

8.68.1.15 #define GRP_CARGODOORINR 31

8.68.1.16 #define GRP_CARGODOOROUTL 3

8.68.1.17 #define GRP_CARGODOOROUTR 2

8.68.1.18 #define GRP_ETUMBDOORL 9

8.68.1.19 #define GRP_ETUMBDOORR 8

8.68.1.20 #define GRP_FLAPL 5

8.68.1.21 #define GRP_FLAPR 4

8.68.1.22 #define GRP_FUSELAGE 0

8.68.1.23 #define GRP_FWD_HOOKS 64

8.68.1.24 #define GRP_GEARDOORL 16

8.68.1.25 #define GRP_GEARDOORR 17

8.68.1.26 #define GRP_GEARL 24

8.68.1.27 #define GRP_GEARR 22

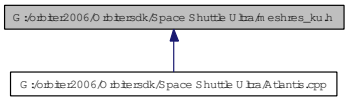
8.68.1.28 #define GRP_HUDL 65

8.68.1.29 #define GRP_HUDR 66

8.68.1.30 #define GRP_KUBAND1 33

8.69 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/meshres_ku.h File Reference

This graph shows which files directly or indirectly include this file:



Defines

- #define NGRP_KU 5
- #define NMAT_KU 2
- #define NTEX_KU 1
- #define GRP_base_KU 0
- #define GRP_cable_one_KU 1
- #define GRP_KUBAND_BOX_KU 2
- #define GRP_KUGIMBAL_KU 3
- #define GRP_KUDISH_KU 4

8.69.1 Define Documentation

8.69.1.1 #define GRP_base_KU 0

8.69.1.2 #define GRP_cable_one_KU 1

8.69.1.3 #define GRP_KUBAND_BOX_KU 2

8.69.1.4 #define GRP_KUDISH_KU 4

8.69.1.5 #define GRP_KUGIMBAL_KU 3

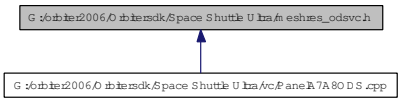
8.69.1.6 #define NGRP_KU 5

8.69.1.7 #define NMAT_KU 2

8.69.1.8 #define NTEX_KU 1

8.70 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/meshres_odsvc.h File Reference

This graph shows which files directly or indirectly include this file:



Defines

- #define NGRPODSVC 87
- #define NMATODSVC 4
- #define NTEXODSVC 2
- #define GRP_OPEN_Hooks_ODSVC 0
- #define GRP_OPEN_Latches_ODSVC 1
- #define GRP_Undocking_ODSVC 2
- #define GRP_Act_Hooks_Firing_ODSVC 3
- #define GRP_Pas_Hooks_Firing_ODSVC 4
- #define GRP_Pyro_Circ_Prot_ON_ODSVC 5
- #define GRP_Pyro_Circ_Prot_OFF_ODSVC 6
- #define GRP_A8A3S1_ODSVC 7
- #define GRP_A8A3S2_ODSVC 8
- #define GRP_A8A3S3_ODSVC 9
- #define GRP_A8A3S6_ODSVC 10
- #define GRP_A8A3S5_ODSVC 11
- #define GRP_A8A3S4_ODSVC 12
- #define GRP_A8A3S7_ODSVC 13
- #define GRP_A8A3S8_ODSVC 14
- #define GRP_A8A3S9_ODSVC 15
- #define GRP_A8A3S12_ODSVC 16
- #define GRP_A8A3S11_ODSVC 17
- #define GRP_A8A3S10_ODSVC 18
- #define GRP_Power_ON_ODSVC 19
- #define GRP_APDS_Circ_Prot_OFF_ODSVC 20
- #define GRP_Ring_Initial_Pos_ODSVC 21
- #define GRP_Ring_Aligned_ODSVC 22
- #define GRP_Fixers_OFF_ODSVC 23
- #define GRP_Hooks1_OPEN_ODSVC 24
- #define GRP_Latches_CLOSED_ODSVC 25
- #define GRP_Hooks2_OPEN_ODSVC 26
- #define GRP_Undock_Complet_ODSVC 27
- #define GRP_Ring_Final_Pos_ODSVC 28
- #define GRP_Hooks2_CLOSED_ODSVC 29
- #define GRP_Latches_OPEN_ODSVC 30
- #define GRP_Hooks1_CLOSED_ODSVC 31

- #define GRP_Interp_Sealed_ODSVC 32
- #define GRP_Ring_Forward_Pos_ODSVC 33
- #define GRP_READY_To_Hook_ODSVC 34
- #define GRP_CAPTURE_ODSVC 35
- #define GRP_Initial_Contact_ODSVC 36
- #define GRP_APDSCC_PowerOn_ODSVC 37
- #define GRP_APDSCC_PowerOff_ODSVC 38
- #define GRP_APDSCC_RingOut_ODSVC 39
- #define GRP_APDSCC_RingIn_ODSVC 40
- #define GRP_APDSCC_FixerOff_ODSVC 41
- #define GRP_APDSCC_CloseLatches_ODSVC 42
- #define GRP_APDSCC_CloseHooks_ODSVC 43
- #define GRP_APDSCC_APDS_Circ_Prot_OFF_ODSVC 44
- #define GRP_ADS_ODSVC 45
- #define GRP_BDS_ODSVC 46
- #define GRP_CDS_ODSVC 47
- #define GRP_LampTest_ODSVC 48
- #define GRP_guards_ODSVC 49
- #define GRP_plateguard2_ODSVC 50
- #define GRP_plateguard1_ODSVC 51
- #define GRP_A8A3panel_ODSVC 52
- #define GRP_A7A3CB1_ODSVC 53
- #define GRP_A7A3panel_ODSVC 54
- #define GRP_A7A3S2_ODSVC 55
- #define GRP_A7A3S1_ODSVC 56
- #define GRP_A7A3S3_ODSVC 57
- #define GRP_A7A3S4_ODSVC 58
- #define GRP_A7A3S6_ODSVC 59
- #define GRP_A7A3S8_ODSVC 60
- #define GRP_A7A3S5_ODSVC 61
- #define GRP_A7A3S7_ODSVC 62
- #define GRP_A7A3S10_ODSVC 63
- #define GRP_A7A3S16_ODSVC 64
- #define GRP_A7A3S11_ODSVC 65
- #define GRP_A7A3S9_ODSVC 66
- #define GRP_A7A3S12_ODSVC 67
- #define GRP_A7A3S15_ODSVC 68
- #define GRP_A7A3S13_ODSVC 69
- #define GRP_A7A3S14_ODSVC 70
- #define GRP_A7A3CB11_ODSVC 71
- #define GRP_A7A3CB2_ODSVC 72
- #define GRP_A7A3CB3_ODSVC 73
- #define GRP_A7A3CB4_ODSVC 74
- #define GRP_A7A3CB5_ODSVC 75
- #define GRP_A7A3CB6_ODSVC 76
- #define GRP_A7A3CB7_ODSVC 77
- #define GRP_A7A3CB8_ODSVC 78
- #define GRP_A7A3CB9_ODSVC 79
- #define GRP_A7A3CB10_ODSVC 80
- #define GRP_A7A3CB12_ODSVC 81

- #define GRP_A7A3CB13_ODSVC 82

- #define GRP_A7A3CB14_ODSVC 83

- #define GRP_A7A3CB15_ODSVC 84

- #define GRP_A7A3CB16_ODSVC 85

- #define GRP_A7A3Guard_ODSVC 86

8.70.1 Define Documentation

8.70.1.1 #define GRP_A7A3CB10_ODSVC 80

8.70.1.2 #define GRP_A7A3CB11_ODSVC 71

8.70.1.3 #define GRP_A7A3CB12_ODSVC 81

8.70.1.4 #define GRP_A7A3CB13_ODSVC 82

8.70.1.5 #define GRP_A7A3CB14_ODSVC 83

8.70.1.6 #define GRP_A7A3CB15_ODSVC 84

8.70.1.7 #define GRP_A7A3CB16_ODSVC 85

8.70.1.8 #define GRP_A7A3CB1_ODSVC 53

8.70.1.9 #define GRP_A7A3CB2_ODSVC 72

8.70.1.10 #define GRP_A7A3CB3_ODSVC 73

8.70.1.11 #define GRP_A7A3CB4_ODSVC 74

8.70.1.12 #define GRP_A7A3CB5_ODSVC 75

8.70.1.13 #define GRP_A7A3CB6_ODSVC 76

8.70.1.14 #define GRP_A7A3CB7_ODSVC 77

8.70.1.15 #define GRP_A7A3CB8_ODSVC 78

8.70.1.16 #define GRP_A7A3CB9_ODSVC 79

8.70.1.17 #define GRP_A7A3Guard_ODSVC 86

8.70.1.18 #define GRP_A7A3panel_ODSVC 54

8.70.1.19 #define GRP_A7A3S10_ODSVC 63

8.70.1.20 #define GRP_A7A3S11_ODSVC 65

8.70.1.21 #define GRP_A7A3S12_ODSVC 67

8.70.1.22 #define GRP_A7A3S13_ODSVC 69

8.70.1.23 #define GRP_A7A3S14_ODSVC 70

8.70.1.24 #define GRP_A7A3S15_ODSVC 68

8.70.1.25 #define GRP_A7A3S16_ODSVC 64

8.70.1.26 #define GRP_A7A3S1_ODSVC 56

8.70.1.27 #define GRP_A7A3S2_ODSVC 55

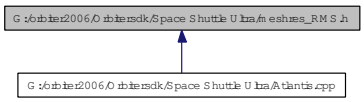
8.70.1.28 #define GRP_A7A3S3_ODSVC 57

8.70.1.29 #define GRP_A7A3S4_ODSVC 58

8.70.1.30 #define GRP_A7A3S5_ODSVC 61

8.71 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/meshres_RMS.h File Reference

This graph shows which files directly or indirectly include this file:



Defines

- #define NGRP_RMS 11
- #define NMAT_RMS 2
- #define NTEX_RMS 1
- #define GRP_RMS_MPMs 0
- #define GRP_base 1
- #define GRP_Shoulder_Yaw 2
- #define GRP_Humerus 3
- #define GRP_Radii 4
- #define GRP_elbowcam 5
- #define GRP_camswivel 6
- #define GRP_cambase 7
- #define GRP_Wristpitch 8
- #define GRP_Wrist_Yaw 9

8.71.1 Define Documentation

8.71.1.1 #define GRP_base 1

8.71.1.2 #define GRP_cambase 7

8.71.1.3 #define GRP_camswivel 6

8.71.1.4 #define GRP_elbowcam 5

8.71.1.5 #define GRP_Humerus 3

8.71.1.6 #define GRP_Radii 4

8.71.1.7 #define GRP_RMS_MPMs 0

8.71.1.8 #define GRP_Shoulder_Yaw 2

8.71.1.9 #define GRP_Wrist_Yaw 9

8.71.1.10 #define GRP_Wristpitch 8

8.71.1.11 #define NGRP_RMS 11

8.71.1.12 #define NMAT_RMS 2

8.71.1.13 #define NTEX_RMS 1

8.72 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/meshres_vc.h File Reference

This graph shows which files directly or indirectly include this file:

Defines

- #define NGRP_VC 853
- #define NMAT_VC 12
- #define NTEX_VC 34
- #define GRP_R13L_VC 0
- #define GRP_B_VC 1
- #define GRP_five_VC 2
- #define GRP_six_VC 3
- #define GRP_clear_VC 4
- #define GRP_GPC_VC 5
- #define GRP_three_VC 6
- #define GRP_D_VC 7
- #define GRP_Fault_VC 8
- #define GRP_ACK_VC 9
- #define GRP_C_VC 10
- #define GRP_IORESET_VC 11
- #define GRP_KEYB_FOUR_VC 12
- #define GRP_KEYB_F_VC 13
- #define GRP_KEYB_DOT_VC 14
- #define GRP_KEYB_A_VC 15
- #define GRP_KEYB_EXEC_VC 16
- #define GRP_KEYB_OPS_VC 17
- #define GRP_KEYB_SPEC_VC 18
- #define GRP_KEYB_ONE_VC 19
- #define GRP_KEYB_TWO_VC 20
- #define GRP_KEYB_RESUME_VC 21
- #define GRP_nine_VC 22
- #define GRP_ITEM_VC 23
- #define GRP_E_VC 24
- #define GRP_eight_VC 25
- #define GRP_MSG_VC 26
- #define GRP_seven_VC 27
- #define GRP_Sys_VC 28
- #define GRP_plus_VC 29
- #define GRP_zero_VC 30
- #define GRP_PRO_VC 31
- #define GRP_minus_VC 32
- #define GRP_CDR1_VC 33
- #define GRP_CDR2_VC 34
- #define GRP_PLT1_VC 35

- #define GRP_PLT2_VC 36
- #define GRP_MFD1_VC 37
- #define GRP_MFD2_VC 38
- #define GRP_MFD3_VC 39
- #define GRP_MFD4_VC 40
- #define GRP_MFD5_VC 41
- #define GRP_MFD_aft_VC 42
- #define GRP_MFD_VC 43
- #define GRP_CDR1LABEL_VC 44
- #define GRP_CDR2LABEL_VC 45
- #define GRP_PLT1LABEL_VC 46
- #define GRP_PLT2LABEL_VC 47
- #define GRP_MFD1LABEL_VC 48
- #define GRP_MFD2LABEL_VC 49
- #define GRP_MFD3LABEL_VC 50
- #define GRP_MFD4LABEL_VC 51
- #define GRP_MFD5LABEL_VC 52
- #define GRP_aft_VC 53
- #define GRP_080_VC 54
- #define GRP_081_VC 55
- #define GRP_082_VC 56
- #define GRP_146_VC 57
- #define GRP_147_VC 58
- #define GRP_VirtualHUD_VC 59
- #define GRP_SwitchR13L_1_VC 60
- #define GRP_SwitchR13L_2_VC 61
- #define GRP_SwitchR13L_3_VC 62
- #define GRP_SwitchR13L_4_VC 63
- #define GRP_SwitchR13L_5_VC 64
- #define GRP_SwitchR13L_6_VC 65
- #define GRP_SwitchR13L_7_VC 66
- #define GRP_SwitchR13L_8_VC 67
- #define GRP_SwitchR13L_9_VC 68
- #define GRP_SwitchR13L_10_VC 69
- #define GRP_SwitchR13L_11_VC 70
- #define GRP_SwitchR13L_12_VC 71
- #define GRP_SwitchR13L_13_VC 72
- #define GRP_Guardb_VC 73
- #define GRP_R2guards_VC 74
- #define GRP_R2panel_VC 75
- #define GRP_R2b39_VC 76
- #define GRP_R2b38_VC 77
- #define GRP_R2b37_VC 78
- #define GRP_R2b43_VC 79
- #define GRP_R2b42_VC 80
- #define GRP_R2b44_VC 81
- #define GRP_R2b45_VC 82
- #define GRP_R2b47_VC 83
- #define GRP_R2b46_VC 84
- #define GRP_R2b54_VC 85

- #define GRP_R2b56_VC 86
- #define GRP_R2b55_VC 87
- #define GRP_R2b53_VC 88
- #define GRP_R2b52_VC 89
- #define GRP_R2b51_VC 90
- #define GRP_R2b50_VC 91
- #define GRP_R2b49_VC 92
- #define GRP_R2b48_VC 93
- #define GRP_R2b30_VC 94
- #define GRP_R2b31_VC 95
- #define GRP_R2b32_VC 96
- #define GRP_R2b33_VC 97
- #define GRP_R2b36_VC 98
- #define GRP_R2b35_VC 99
- #define GRP_R2b34_VC 100
- #define GRP_R2b29_VC 101
- #define GRP_R2b17_VC 102
- #define GRP_R2b16_VC 103
- #define GRP_R2b40_VC 104
- #define GRP_R2b41_VC 105
- #define GRP_R2b24_VC 106
- #define GRP_R2b25_VC 107
- #define GRP_R2b26_VC 108
- #define GRP_R2b27_VC 109
- #define GRP_R2b28_VC 110
- #define GRP_R2b18_VC 111
- #define GRP_R2b19_VC 112
- #define GRP_R2b20_VC 113
- #define GRP_R2b21_VC 114
- #define GRP_R2b22_VC 115
- #define GRP_R2b23_VC 116
- #define GRP_R2b15_VC 117
- #define GRP_R2b14_VC 118
- #define GRP_R2b13_VC 119
- #define GRP_R2b12_VC 120
- #define GRP_R2b9_VC 121
- #define GRP_R2b10_VC 122
- #define GRP_R2b11_VC 123
- #define GRP_R2b8_VC 124
- #define GRP_R2b7_VC 125
- #define GRP_R2b6_VC 126
- #define GRP_R2b3_VC 127
- #define GRP_R2b4_VC 128
- #define GRP_R2b5_VC 129
- #define GRP_R2b2_VC 130
- #define GRP_R2b1_VC 131
- #define GRP_A8Guardb_VC 132
- #define GRP_Dial_VC 133
- #define GRP_Dialsmall_VC 135
- #define GRP_A8bot_VC 138

- #define GRP_A8top_VC 139
- #define GRP_C3cover1_VC 161
- #define GRP_C3cover2_VC 162
- #define GRP_C3cover3_VC 163
- #define GRP_C3cover4_VC 164
- #define GRP_C3cover5_VC 165
- #define GRP_C3_VC 166
- #define GRP_C3Dial_VC 167
- #define GRP_C3b1_VC 168
- #define GRP_C3b2_VC 169
- #define GRP_C3b3_VC 170
- #define GRP_C3b4_VC 171
- #define GRP_C3b5_VC 172
- #define GRP_C3b6_VC 173
- #define GRP_C3b7_VC 174
- #define GRP_C3b8_VC 175
- #define GRP_C3b9_VC 176
- #define GRP_C3b10_VC 177
- #define GRP_C3b11_VC 178
- #define GRP_C3b12_VC 179
- #define GRP_C3b13_VC 180
- #define GRP_C3b14v_VC 181
- #define GRP_C3b15_VC 182
- #define GRP_C3b16_VC 183
- #define GRP_C3b17_VC 184
- #define GRP_C3b18v_VC 185
- #define GRP_C3b19_VC 186
- #define GRP_C3b20_VC 187
- #define GRP_C3b21_VC 188
- #define GRP_C3b22_VC 189
- #define GRP_C3b23_VC 190
- #define GRP_C3b24_VC 191
- #define GRP_C3b25_VC 192
- #define GRP_C3b26_VC 193
- #define GRP_C3b27_VC 194
- #define GRP_C3b28_VC 195
- #define GRP_C3b29_VC 196
- #define GRP_C3b30_VC 197
- #define GRP_C3b31_VC 198
- #define GRP_C3b32_VC 199
- #define GRP_C3b33_VC 200
- #define GRP_C3b34_VC 201
- #define GRP_C3b35_VC 202
- #define GRP_C3b36_VC 203
- #define GRP_C3b37_VC 204
- #define GRP_C2b8v_VC 205
- #define GRP_C2guards_VC 206
- #define GRP_C2tog4_VC 207
- #define GRP_C2tog3_VC 208
- #define GRP_C2tog2_VC 209

- #define GRP_C2tog1_VC 210
- #define GRP_Clock1_VC 211
- #define GRP_Clock2_VC 212
- #define GRP_Clock3_VC 213
- #define GRP_Clock4_VC 214
- #define GRP_C2b1_VC 215
- #define GRP_C2b7v_VC 216
- #define GRP_C2b2_VC 217
- #define GRP_C2b3_VC 218
- #define GRP_C2b4_VC 219
- #define GRP_C2b5_VC 220
- #define GRP_C2b6_VC 221
- #define GRP_C2b9_VC 222
- #define GRP_C2b10_VC 223
- #define GRP_C2b11_VC 224
- #define GRP_C2_VC 225
- #define GRP_R13talk1_VC 226
- #define GRP_R13talk2_VC 227
- #define GRP_R13talk3_VC 228
- #define GRP_R13talk4_VC 229
- #define GRP_R13talk5_VC 230
- #define GRP_R13talk6_VC 231
- #define GRP_R13talk7_VC 232
- #define GRP_R13talk8_VC 233
- #define GRP_L2panel_VC 234
- #define GRP_L2b11v_VC 235
- #define GRP_L2b17v_VC 236
- #define GRP_L2b12_VC 237
- #define GRP_L2b10_VC 238
- #define GRP_L2b7_VC 239
- #define GRP_L2b9_VC 240
- #define GRP_L2b8_VC 241
- #define GRP_L2b5_VC 242
- #define GRP_L2b6_VC 243
- #define GRP_L2b2_VC 244
- #define GRP_L2b1_VC 245
- #define GRP_L2b4_VC 246
- #define GRP_L2b3_VC 247
- #define GRP_L2b13_VC 248
- #define GRP_L2b14_VC 249
- #define GRP_L2b16_VC 250
- #define GRP_L2b15_VC 251
- #define GRP_L2b24_VC 252
- #define GRP_L2b23_VC 253
- #define GRP_L2b21_VC 254
- #define GRP_L2b18_VC 255
- #define GRP_L2b20_VC 256
- #define GRP_L2b22_VC 257
- #define GRP_L2b19_VC 258
- #define GRP_L2b25_VC 259

- #define GRP_guards_VC 260
- #define GRP_L1dial3_VC 261
- #define GRP_L1dial2_VC 262
- #define GRP_L1dial1_VC 263
- #define GRP_L1b1_VC 264
- #define GRP_L1b2_VC 265
- #define GRP_L1b3_VC 266
- #define GRP_L1b4_VC 267
- #define GRP_L1b5_VC 268
- #define GRP_L1b6_VC 269
- #define GRP_L1b7_VC 270
- #define GRP_L1b8_VC 271
- #define GRP_L1b9_VC 272
- #define GRP_L1b10_VC 273
- #define GRP_L1b11_VC 274
- #define GRP_L1b12_VC 275
- #define GRP_L1b13_VC 276
- #define GRP_L1b14_VC 277
- #define GRP_L1b15_VC 278
- #define GRP_L1b16_VC 279
- #define GRP_L1b17_VC 280
- #define GRP_L1b18_VC 281
- #define GRP_L1b19_VC 282
- #define GRP_L1b20_VC 283
- #define GRP_L1b21_VC 284
- #define GRP_L1b23_VC 285
- #define GRP_L1b24_VC 286
- #define GRP_L1b25_VC 287
- #define GRP_L1b26_VC 288
- #define GRP_L1b27_VC 289
- #define GRP_L1b28_VC 290
- #define GRP_L1b29_VC 291
- #define GRP_L1b30_VC 292
- #define GRP_L1b31_VC 293
- #define GRP_L1b32_VC 294
- #define GRP_L1b34_VC 295
- #define GRP_L1b35_VC 296
- #define GRP_L1b36_VC 297
- #define GRP_L1b37_VC 298
- #define GRP_L1b38_VC 299
- #define GRP_L1b39_VC 300
- #define GRP_L1b40_VC 301
- #define GRP_L1b41_VC 302
- #define GRP_L1b42_VC 303
- #define GRP_L1b43_VC 304
- #define GRP_L1b44_VC 305
- #define GRP_L1b45_VC 306
- #define GRP_L1b46_VC 307
- #define GRP_L1b47_VC 308
- #define GRP_L1Panel_VC 309

- #define GRP_L1cover3_VC 310
- #define GRP_L1cover1_VC 311
- #define GRP_L1cover2_VC 312
- #define GRP_L1b33_VC 314
- #define GRP_L1b22_VC 315
- #define GRP_R1b1_VC 318
- #define GRP_R1b2_VC 319
- #define GRP_R1b3_VC 320
- #define GRP_R1b4_VC 321
- #define GRP_R1b5_VC 322
- #define GRP_Coas_VC 383
- #define GRP_F6panel1_VC 384
- #define GRP_F6Guard_VC 385
- #define GRP_F6TalkBack3_VC 386
- #define GRP_F6TalkBack2_VC 387
- #define GRP_F6TalkBack1_VC 388
- #define GRP_F6Dial_VC 389
- #define GRP_F6switch_9_VC 390
- #define GRP_F6switch_8_VC 391
- #define GRP_F6switch_7_VC 392
- #define GRP_F6switch_6_VC 393
- #define GRP_F6switch_5_VC 394
- #define GRP_F6switch_4_VC 395
- #define GRP_F6switch_3_VC 396
- #define GRP_F6switch_2_VC 397
- #define GRP_F6switch_1_VC 398
- #define GRP_F6button7_VC 399
- #define GRP_F6button5_VC 400
- #define GRP_F6button4_VC 401
- #define GRP_F6button6_VC 402
- #define GRP_F6button3_VC 403
- #define GRP_F6button2_VC 404
- #define GRP_F6button1_VC 405
- #define GRP_F6cover2_VC 406
- #define GRP_F6cover1_VC 407
- #define GRP_F8pan1_VC 408
- #define GRP_F8Guard_VC 409
- #define GRP_F8switch9_VC 410
- #define GRP_F8switch6_VC 411
- #define GRP_F8switch8_VC 412
- #define GRP_F8switch7_VC 413
- #define GRP_F8switch3_VC 414
- #define GRP_F8switch2_VC 415
- #define GRP_F8switch1_VC 416
- #define GRP_F8switch5_VC 417
- #define GRP_F8switch4_VC 418
- #define GRP_F8TalkBack1_VC 419
- #define GRP_F8TalkBack2_VC 420
- #define GRP_F8TalkBack3_VC 421
- #define GRP_F8button1_VC 422

- #define GRP_F8button2_VC 423
- #define GRP_F8button3_VC 424
- #define GRP_F8cover1_VC 425
- #define GRP_F8cover2_VC 426
- #define GRP_Eventtimer_VC 427
- #define GRP_F7_Event_M10_VC 428
- #define GRP_F7_Event_M_VC 429
- #define GRP_F7_Event_S10_VC 430
- #define GRP_F7_Event_S_VC 431
- #define GRP_A4MissionSwitch_VC 433
- #define GRP_A4_Event_S_VC 434
- #define GRP_A4_Event_S10_VC 435
- #define GRP_A4_Event_M_VC 436
- #define GRP_A4_Event_M10_VC 437
- #define GRP_A4_MET_S2_VC 438
- #define GRP_A4_MET_S1_VC 439
- #define GRP_A4_MET_M2_VC 440
- #define GRP_A4_MET_M1_VC 441
- #define GRP_A4_MET_H2_VC 442
- #define GRP_A4_MET_H1_VC 443
- #define GRP_A4_MET_DAY3_VC 444
- #define GRP_A4_MET_DAY2_VC 445
- #define GRP_A4_MET_DAY1_VC 446
- #define GRP_panelA4_VC 447
- #define GRP_measure_strips_VC 448
- #define GRP_stripbases_VC 449
- #define GRP_O3_guards_VC 450
- #define GRP_cable_VC 451
- #define GRP_O1S1_VC 452
- #define GRP_3_PRPLT_LEFT1_VC 453
- #define GRP_O3_PRPLT_LEFT2_VC 454
- #define GRP_O3_PRPLT_FWD1_VC 455
- #define GRP_O3_PRPLT_FWD2_VC 456
- #define GRP_O3_PRPLT_RIGHT1_VC 457
- #define GRP_O3_PRPLT_RIGHT2_VC 458
- #define GRP_O3_MET_DAY1_VC 459
- #define GRP_O3_MET_DAY2_VC 460
- #define GRP_O3_MET_DAY3_VC 461
- #define GRP_O3_MET_H1_VC 462
- #define GRP_O3_MET_H2_VC 463
- #define GRP_O3_MET_M2_VC 464
- #define GRP_O3_MET_M1_VC 465
- #define GRP_O3_MET_S1_VC 466
- #define GRP_O3_MET_S2_VC 467
- #define GRP_O1S4_VC 468
- #define GRP_O1S3_VC 469
- #define GRP_O1S6_VC 470
- #define GRP_O3S12_VC 471
- #define GRP_O2S9_VC 472
- #define GRP_O3S11_VC 473

- #define GRP_O3S10_VC 474
- #define GRP_O2S8_VC 475
- #define GRP_O2S7_VC 476
- #define GRP_O1S5_VC 477
- #define GRP_O1S2_VC 478
- #define GRP_O2_VC 479
- #define GRP_O3_VC 480
- #define GRP_O1_VC 481
- #define GRP_GPC_Status_VC 482
- #define GRP_O1M1T_VC 483
- #define GRP_O1M1P_VC 484
- #define GRP_O1M2F_VC 485
- #define GRP_O1M2T_VC 486
- #define GRP_O1M3P_VC 487
- #define GRP_O1M3F_VC 488
- #define GRP_O1M4P_VC 489
- #define GRP_O1M4ppo_VC 490
- #define GRP_O2M5T_VC 491
- #define GRP_O2M6P_VC 492
- #define GRP_O2M6Q_VC 493
- #define GRP_O2M7P_VC 494
- #define GRP_O2M7Q_VC 495
- #define GRP_O2M8T_VC 496
- #define GRP_O3M9O_VC 497
- #define GRP_O3M9F_VC 498
- #define GRP_O3M10O_VC 499
- #define GRP_O3M10F_VC 500
- #define GRP_O3M11O_VC 501
- #define GRP_O3M11F_VC 502
- #define GRP_covers_VC 503
- #define GRP_A1RS1_VC 504
- #define GRP_A1RS32_VC 505
- #define GRP_A1RS34_VC 506
- #define GRP_A1RS29_VC 507
- #define GRP_A1RS27_VC 508
- #define GRP_A1RS25_VC 509
- #define GRP_A1RS18_VC 510
- #define GRP_A1RS20_VC 511
- #define GRP_A1RS22_VC 512
- #define GRP_A1RS35_VC 513
- #define GRP_A1RS33_VC 514
- #define GRP_A1RS31_VC 515
- #define GRP_A1RS24_VC 516
- #define GRP_A1RS26_VC 517
- #define GRP_A1RS28_VC 518
- #define GRP_A1RS30_VC 519
- #define GRP_A1RS23_VC 520
- #define GRP_A1RS21_VC 521
- #define GRP_A1RS19_VC 522
- #define GRP_A1RS17_VC 523

- #define GRP_A1RS12_VC 524
- #define GRP_A1RS11_VC 525
- #define GRP_A1RS13_VC 526
- #define GRP_A1RS9_VC 527
- #define GRP_A1RS7_VC 528
- #define GRP_A1RS6_VC 529
- #define GRP_A1RS5_VC 530
- #define GRP_A1RS3_VC 531
- #define GRP_A1RS2_VC 532
- #define GRP_A1RRS15_VC 533
- #define GRP_A1RRS16_VC 534
- #define GRP_A1RRS14_VC 535
- #define GRP_A1RRS10_VC 536
- #define GRP_A1RRS4_VC 537
- #define GRP_A1RRS8_VC 538
- #define GRP_A1R_VC 539
- #define GRP_A1LS17_VC 540
- #define GRP_A1LS4_VC 541
- #define GRP_A1LS15_VC 542
- #define GRP_A1LS14_VC 543
- #define GRP_A1LS13_VC 544
- #define GRP_A1LS1_VC 545
- #define GRP_A1LS2_VC 546
- #define GRP_A1US10_VC 547
- #define GRP_A1LS5_VC 548
- #define GRP_A1LS11_VC 549
- #define GRP_A1LS12_VC 550
- #define GRP_A1LS16_VC 551
- #define GRP_A1LS8_VC 552
- #define GRP_A1LS9_VC 553
- #define GRP_A1LS20_VC 554
- #define GRP_A1LS19_VC 555
- #define GRP_A1LS18_VC 556
- #define GRP_A1LS21_VC 557
- #define GRP_A1LS22_VC 558
- #define GRP_A1LS26_VC 559
- #define GRP_A1LS24_VC 560
- #define GRP_A1LS25_VC 561
- #define GRP_A1LS23_VC 562
- #define GRP_A1LRS6_VC 563
- #define GRP_A1Lpanel_VC 564
- #define GRP_A1URS7_VC 565
- #define GRP_A1URS15_VC 566
- #define GRP_A1URS16_VC 567
- #define GRP_A1US8_VC 568
- #define GRP_A1US12_VC 569
- #define GRP_A1US13_VC 570
- #define GRP_A1US14_VC 571
- #define GRP_A1US18_VC 572
- #define GRP_A1US6_VC 573

- #define GRP_A1US2_VC 574
- #define GRP_A1US3_VC 575
- #define GRP_A1US4_VC 576
- #define GRP_A1US1_VC 577
- #define GRP_A1U_VC 578
- #define GRP_A1UM1_VC 579
- #define GRP_A1Guard_VC 580
- #define GRP_A1_VC 581
- #define GRP_F8HUDbrt_VC 582
- #define GRP_F8HUDtest_VC 583
- #define GRP_F8HUDdim_VC 584
- #define GRP_F3S6_VC 585
- #define GRP_F3S4_VC 586
- #define GRP_F3S5_VC 587
- #define GRP_F3S2_VC 588
- #define GRP_F3S1_VC 589
- #define GRP_F3S3_VC 590
- #define GRP_F6HUDdim_VC 591
- #define GRP_F6HUDtest_VC 592
- #define GRP_F6HUDbrt_VC 593
- #define GRP_F7CW_VC 594
- #define GRP_F6panel_VC 595
- #define GRP_F8panel_VC 596
- #define GRP_Fcovers_VC 598
- #define GRP_A6US37_VC 600
- #define GRP_A6US4_VC 601
- #define GRP_A6URS5_VC 602
- #define GRP_A6URS4_VC 603
- #define GRP_A6URS3_VC 604
- #define GRP_A6URS1_VC 605
- #define GRP_A6URS2_VC 606
- #define GRP_A6U39_VC 607
- #define GRP_A6U40_VC 608
- #define GRP_A6U41_VC 609
- #define GRP_A6U45_VC 610
- #define GRP_A6U44_VC 611
- #define GRP_A6U43_VC 612
- #define GRP_A6U42_VC 613
- #define GRP_A6U36_VC 614
- #define GRP_A6U47_VC 615
- #define GRP_A6U46_VC 616
- #define GRP_A6U7_VC 617
- #define GRP_A6U5_VC 618
- #define GRP_A6U1_VC 619
- #define GRP_A6U34_VC 620
- #define GRP_A6U35_VC 621
- #define GRP_A6U33_VC 622
- #define GRP_A6U32_VC 623
- #define GRP_A6U2_VC 624
- #define GRP_A6U6_VC 625

- #define GRP_A6Upanel_VC 626
- #define GRP_Clock8_VC 627
- #define GRP_Clock7_VC 628
- #define GRP_Clock6_VC 629
- #define GRP_Clock5_VC 630
- #define GRP_A6Utog1_VC 631
- #define GRP_A6Utog2_VC 632
- #define GRP_A6Utog3_VC 633
- #define GRP_A6Utog4_VC 634
- #define GRP_A7U11_VC 635
- #define GRP_A7U44_VC 636
- #define GRP_A7U43_VC 637
- #define GRP_A7U56_VC 638
- #define GRP_A7U40_VC 639
- #define GRP_A7U42_VC 640
- #define GRP_A7U41_VC 641
- #define GRP_A7U52_VC 642
- #define GRP_A7U12_VC 643
- #define GRP_A7U18_VC 644
- #define GRP_A7U16_VC 645
- #define GRP_A7U17_VC 646
- #define GRP_A7U15_VC 647
- #define GRP_A7U14_VC 648
- #define GRP_A7U2_VC 649
- #define GRP_A7U1_VC 650
- #define GRP_A7U3_VC 651
- #define GRP_A7U4_VC 652
- #define GRP_A7U6_VC 653
- #define GRP_A7U5_VC 654
- #define GRP_A7U7_VC 655
- #define GRP_A7U8_VC 656
- #define GRP_A7U54_VC 657
- #define GRP_A7U53_VC 658
- #define GRP_A7U10_VC 659
- #define GRP_A7U9_VC 660
- #define GRP_A7U13_VC 662
- #define GRP_MFD11_VC 664
- #define GRP_A2panel_VC 665
- #define GRP_A2U1_VC 666
- #define GRP_A2U2_VC 667
- #define GRP_translational_VC 668
- #define GRP_transguard_VC 669
- #define GRP_transboot_VC 671
- #define GRP_transRMS_VC 672
- #define GRP_transRMSboot_VC 673
- #define GRP_A2AZH_VC 674
- #define GRP_A2ELH_VC 675
- #define GRP_A2ELV_VC 676
- #define GRP_A2AZV_VC 677
- #define GRP_AftMonitor_VC 678

- #define GRP_MDUs_VC 679
- #define GRP_O6S50_VC 680
- #define GRP_O6S49_VC 681
- #define GRP_O6S48_VC 682
- #define GRP_O6S47_VC 683
- #define GRP_O6S45_VC 684
- #define GRP_O6S28_VC 685
- #define GRP_O6S26_VC 686
- #define GRP_O6S29_VC 687
- #define GRP_O6S27_VC 688
- #define GRP_O6S25_VC 689
- #define GRP_O6S20_VC 690
- #define GRP_O6S22_VC 691
- #define GRP_O6S24_VC 692
- #define GRP_O6S21_VC 693
- #define GRP_O6S23_VC 694
- #define GRP_O6S18_VC 695
- #define GRP_O6S16_VC 696
- #define GRP_O6S19_VC 697
- #define GRP_O6S17_VC 698
- #define GRP_O6S52_VC 699
- #define GRP_O6R2_VC 700
- #define GRP_O6RS6_VC 701
- #define GRP_O6R7_VC 702
- #define GRP_O6R1_VC 703
- #define GRP_O6R5_VC 704
- #define GRP_O6R6_VC 705
- #define GRP_O6R4_VC 706
- #define GRP_O6S35_VC 707
- #define GRP_O6S36_VC 708
- #define GRP_O6S37_VC 709
- #define GRP_O6S38_VC 710
- #define GRP_O6S39_VC 711
- #define GRP_O6S2_VC 712
- #define GRP_O6S3_VC 713
- #define GRP_O6S34_VC 714
- #define GRP_O6S33_VC 715
- #define GRP_O6S32_VC 716
- #define GRP_O6S31_VC 717
- #define GRP_O6S30_VC 718
- #define GRP_O6S15_VC 719
- #define GRP_O6S51_VC 720
- #define GRP_O6S14_VC 721
- #define GRP_O6S12_VC 722
- #define GRP_O6S13_VC 723
- #define GRP_O6S6_VC 724
- #define GRP_O6S1_VC 725
- #define GRP_O6S5_VC 726
- #define GRP_O6switchA_VC 727
- #define GRP_O6switchB_VC 728

- #define GRP_O6switchC_VC 729
- #define GRP_O6R3_VC 730
- #define GRP_O6S4_VC 731
- #define GRP_O6panel_VC 732
- #define GRP_O6Guard_VC 733
- #define GRP_O6S46_VC 734
- #define GRP_O7RS5_VC 735
- #define GRP_O7RS3_VC 736
- #define GRP_O7RS1_VC 737
- #define GRP_O7S36_VC 738
- #define GRP_O7S35_VC 739
- #define GRP_O7S34_VC 740
- #define GRP_O7S33_VC 741
- #define GRP_O7S32_VC 742
- #define GRP_O7S27_VC 743
- #define GRP_O7S29_VC 744
- #define GRP_O7S30_VC 745
- #define GRP_O7S28_VC 746
- #define GRP_O7S31_VC 747
- #define GRP_O7S26_VC 748
- #define GRP_O7S23_VC 749
- #define GRP_O7S25_VC 750
- #define GRP_O7S24_VC 751
- #define GRP_O7S22_VC 752
- #define GRP_O7S16b_VC 753
- #define GRP_O7S17b_VC 754
- #define GRP_O7S18b_VC 755
- #define GRP_O7S18a_VC 756
- #define GRP_O7S17a_VC 757
- #define GRP_O7S16a_VC 758
- #define GRP_O7S14_VC 759
- #define GRP_O7S13_VC 760
- #define GRP_O7S11_VC 761
- #define GRP_O7S10_VC 762
- #define GRP_O7S6_VC 763
- #define GRP_O7S4_VC 764
- #define GRP_O7S2_VC 765
- #define GRP_O7panel_VC 766
- #define GRP_O7Guard_VC 767
- #define GRP_O8Guard_VC 768
- #define GRP_O8S33_VC 769
- #define GRP_O8S31_VC 770
- #define GRP_O8S30_VC 771
- #define GRP_O8S34_VC 772
- #define GRP_O8S32_VC 773
- #define GRP_O8S28_VC 774
- #define GRP_O8S26_VC 775
- #define GRP_O8S27_VC 776
- #define GRP_O8S29_VC 777
- #define GRP_O8S22_VC 778

- #define GRP_O8S20_VC 779
- #define GRP_O8S7_VC 780
- #define GRP_O8S6_VC 781
- #define GRP_O8S19_VC 782
- #define GRP_O8S21_VC 783
- #define GRP_O8S23_VC 784
- #define GRP_O8S24_VC 785
- #define GRP_O8S16_VC 786
- #define GRP_O8S17_VC 787
- #define GRP_O8S15_VC 788
- #define GRP_O8S13_VC 789
- #define GRP_O8S2a_VC 790
- #define GRP_O8S1a_VC 791
- #define GRP_O8S12_VC 792
- #define GRP_O8S14_VC 793
- #define GRP_O8S18_VC 794
- #define GRP_O8R6_VC 795
- #define GRP_O8S1_VC 796
- #define GRP_O8S3_VC 797
- #define GRP_O8S10_VC 798
- #define GRP_O8S8_VC 799
- #define GRP_O8S9_VC 800
- #define GRP_O8R1_VC 801
- #define GRP_O8R4_VC 802
- #define GRP_O8R5_VC 803
- #define GRP_O8R3_VC 804
- #define GRP_O8S5_VC 805
- #define GRP_O8R2_VC 806
- #define GRP_O8S4_VC 807
- #define GRP_O8panel_VC 808
- #define GRP_F9needle3_VC 809
- #define GRP_F9needle2_VC 810
- #define GRP_F9needle1_VC 811
- #define GRP_topdial_VC 812
- #define GRP_bottomdial_VC 813
- #define GRP_bottomcup_VC 814
- #define GRP_topcup_VC 815
- #define GRP_invisibletop_VC 816
- #define GRP_invisiblebottom_VC 817
- #define GRP_F9panel_VC 818
- #define GRP_leftface_VC 819
- #define GRP_rightface_VC 820
- #define GRP_topface_VC 821
- #define GRP_panelR4_VC 822
- #define GRP_R4Guard_VC 823
- #define GRP_R4S26_VC 824
- #define GRP_R4S23_VC 825
- #define GRP_R4S29_VC 826
- #define GRP_R4S9_VC 827
- #define GRP_R4S7_VC 828

- #define GRP_R4S13_VC 829
- #define GRP_R4S19_VC 830
- #define GRP_R4S11_VC 831
- #define GRP_R4S12_VC 832
- #define GRP_R4S14_VC 833
- #define GRP_R4S16_VC 834
- #define GRP_R4S15_VC 835
- #define GRP_R4S21_VC 836
- #define GRP_R4S17_VC 837
- #define GRP_R4S18_VC 838
- #define GRP_R4S27_VC 839
- #define GRP_R4S25_VC 840
- #define GRP_R4S22_VC 841
- #define GRP_R4S24_VC 842
- #define GRP_R4S30_VC 843
- #define GRP_R4S28_VC 844
- #define GRP_R4S6_VC 845
- #define GRP_R4S8_VC 846
- #define GRP_R4S10_VC 847
- #define GRP_R4S5_VC 848
- #define GRP_R4S4_VC 849
- #define GRP_R4S3_VC 850
- #define GRP_R4S2_VC 851
- #define GRP_R4S1_VC 852

8.72.1 Define Documentation

8.72.1.1 #define GRP_080_VC 54

8.72.1.2 #define GRP_081_VC 55

8.72.1.3 #define GRP_082_VC 56

8.72.1.4 #define GRP_146_VC 57

8.72.1.5 #define GRP_147_VC 58

8.72.1.6 #define GRP_3_PRPLT_LEFT1_VC 453

8.72.1.7 #define GRP_A1_VC 581

8.72.1.8 #define GRP_A1Guard_VC 580

8.72.1.9 #define GRP_A1Lpanel_VC 564

8.72.1.10 #define GRP_A1LRS6_VC 563

8.72.1.11 #define GRP_A1LS11_VC 549

8.72.1.12 #define GRP_A1LS12_VC 550

8.72.1.13 #define GRP_A1LS13_VC 544

8.72.1.14 #define GRP_A1LS14_VC 543

8.72.1.15 #define GRP_A1LS15_VC 542

8.72.1.16 #define GRP_A1LS16_VC 551

8.72.1.17 #define GRP_A1LS17_VC 540

8.72.1.18 #define GRP_A1LS18_VC 556

8.72.1.19 #define GRP_A1LS19_VC 555

8.72.1.20 #define GRP_A1LS1_VC 545

8.72.1.21 #define GRP_A1LS20_VC 554

8.72.1.22 #define GRP_A1LS21_VC 557

8.72.1.23 #define GRP_A1LS22_VC 558

8.72.1.24 #define GRP_A1LS23_VC 562

8.72.1.25 #define GRP_A1LS24_VC 560

8.72.1.26 #define GRP_A1LS25_VC 561

8.72.1.27 #define GRP_A1LS26_VC 559

8.72.1.28 #define GRP_A1LS2_VC 546

8.72.1.29 #define GRP_A1LS4_VC 541

8.72.1.30 #define GRP_A1LS5_VC 548

8.73 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/meshres_vc_-additions.h File Reference

This graph shows which files directly or indirectly include this file:

Defines

- #define GRP_A8b1_VC 140
- #define GRP_A8b2_VC 141
- #define GRP_A8b3_VC 142
- #define GRP_A8b4_VC 143
- #define GRP_A8b5_VC 144
- #define GRP_A8b6_VC 145
- #define GRP_A8b7_VC 146
- #define GRP_A8b8_VC 147
- #define GRP_A8b9_VC 148
- #define GRP_A8b10_VC 149
- #define GRP_A8b11_VC 150
- #define GRP_A8b12_VC 151
- #define GRP_A8b13_VC 152
- #define GRP_A8b14_VC 153
- #define GRP_A8b15_VC 154
- #define GRP_A8b16_VC 155
- #define GRP_A8b17_VC 156
- #define GRP_C2b8_VC 205
- #define GRP_C2Evt1SW_VC 207
- #define GRP_C2Evt10SW_VC 208
- #define GRP_C2Evt1MW_VC 209
- #define GRP_C2Evt10MW_VC 210
- #define GRP_C2b1_VC 215
- #define GRP_C2b7_VC 216
- #define GRP_C2b2_VC 217
- #define GRP_C2b3_VC 218
- #define GRP_C2b4_VC 219
- #define GRP_C2b5_VC 220
- #define GRP_C2b6_VC 221
- #define GRP_C2b9_VC 222
- #define GRP_C2b10_VC 223
- #define GRP_C2b12_VC 224
- #define GRP_Eventtimer_VC 427
- #define GRP_F7_Event_M10_VC 428
- #define GRP_F7_Event_M_VC 429
- #define GRP_F7_Event_S10_VC 430
- #define GRP_F7_Event_S_VC 431
- #define GRP_A4MissionSwitch_VC 433

- #define GRP_A4_Event_S_VC 434
- #define GRP_A4_Event_S10_VC 435
- #define GRP_A4_Event_M_VC 436
- #define GRP_A4_Event_M10_VC 437
- #define GRP_A4_MET_S2_VC 438
- #define GRP_A4_MET_S1_VC 439
- #define GRP_A4_MET_M2_VC 440
- #define GRP_A4_MET_M1_VC 441
- #define GRP_A4_MET_H2_VC 442
- #define GRP_A4_MET_H1_VC 443
- #define GRP_A4_MET_DAY3_VC 444
- #define GRP_A4_MET_DAY2_VC 445
- #define GRP_A4_MET_DAY1_VC 446
- #define GRP_panelA4_VC 447
- #define TEX_ATLANTIS 2
- #define TEX_FPANELS_VC 1
- #define TEX_A1SHADOW_VC 2
- #define TEX_PANELA1U_VC 3
- #define TEX_PANELA1L_VC 4
- #define TEX_PANELA1R_VC 5
- #define TEX_PANELO1_VC 6
- #define TEX_PANELO3_VC 7
- #define TEX_DIGITS_VC 8
- #define TEX_AFTTIMER_VC 9
- #define TEX_TIMER_VC 10
- #define TEX_TALKBACK_VC 11
- #define TEX_PANELF8_VC 12
- #define TEX_A8TOP_VC 13
- #define TEX_PANELF6_VC 14
- #define TEX_COAS_VC 15
- #define TEX_R1L1_VC 16
- #define TEX_PANELC3_VC 17
- #define TEX_PANELR2_VC 18
- #define TEX_PANELC2_VC 19
- #define TEX_CLOCKNUMS_VC 20
- #define TEX_A8BOTTOM_VC 21
- #define TEX_STSVC03_VC 22
- #define TEX_STSVC02_VC 23
- #define TEX_STSVC01_VC 24
- #define TEX_LABEL_VC 25
- #define TEX_GPC_VC 26
- #define TEX_R13L_VC 27

8.73.1 Define Documentation

8.73.1.1 #define GRP_A4_Event_M10_VC 437

8.73.1.2 #define GRP_A4_Event_M_VC 436

8.73.1.3 #define GRP_A4_Event_S10_VC 435

8.73.1.4 #define GRP_A4_Event_S_VC 434

8.73.1.5 #define GRP_A4_MET_DAY1_VC 446

8.73.1.6 #define GRP_A4_MET_DAY2_VC 445

8.73.1.7 #define GRP_A4_MET_DAY3_VC 444

8.73.1.8 #define GRP_A4_MET_H1_VC 443

8.73.1.9 #define GRP_A4_MET_H2_VC 442

8.73.1.10 #define GRP_A4_MET_M1_VC 441

8.73.1.11 #define GRP_A4_MET_M2_VC 440

8.73.1.12 #define GRP_A4_MET_S1_VC 439

8.73.1.13 #define GRP_A4_MET_S2_VC 438

8.73.1.14 #define GRP_A4MissionSwitch_VC 433

8.73.1.15 #define GRP_A8b10_VC 149

8.73.1.16 #define GRP_A8b11_VC 150

8.73.1.17 #define GRP_A8b12_VC 151

8.73.1.18 #define GRP_A8b13_VC 152

8.73.1.19 #define GRP_A8b14_VC 153

8.73.1.20 #define GRP_A8b15_VC 154

8.73.1.21 #define GRP_A8b16_VC 155

8.73.1.22 #define GRP_A8b17_VC 156

8.73.1.23 #define GRP_A8b1_VC 140

8.73.1.24 #define GRP_A8b2_VC 141

8.73.1.25 #define GRP_A8b3_VC 142

8.73.1.26 #define GRP_A8b4_VC 143

8.73.1.27 #define GRP_A8b5_VC 144

8.73.1.28 #define GRP_A8b6_VC 145

8.73.1.29 #define GRP_A8b7_VC 146

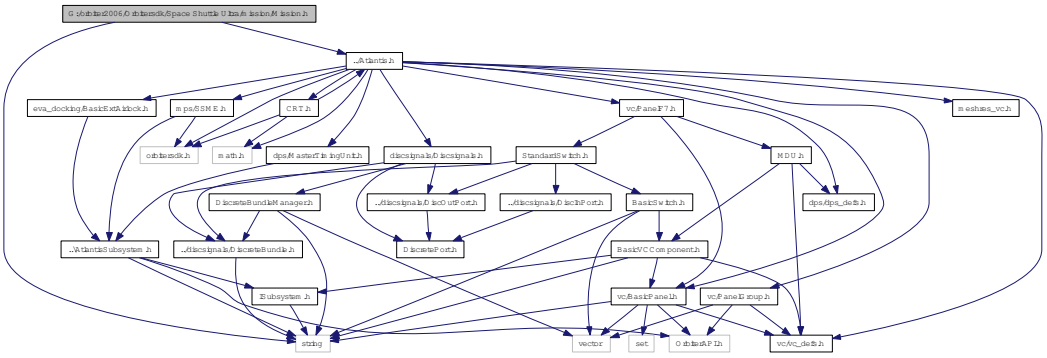
8.73.1.30 #define GRP_A8b8_VC 147

8.74 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mission/Mission.h File Reference

```
#include "../Atlantis.h"
```

```
#include <string>
```

Include dependency graph for Mission.h:



Namespaces

- namespace mission

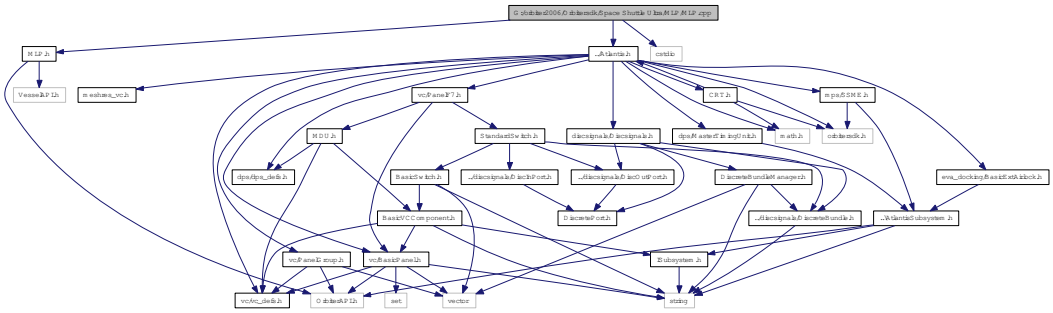
Classes

- class mission::Mission

8.75 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/MLP/MLP.cpp File Reference

```
#include "../Atlantis.h"
#include "MLP.h"
#include <cstdio>
```

Include dependency graph for MLP.cpp:



Functions

- DLLCLBK void InitModule (HINSTANCE hDLL)
- DLLCLBK void ExitModule (HINSTANCE hDLL)
- DLLCLBK VESSEL ∗ ovcInit (OBJHANDLE hVessel, int iFlightModel)
- DLLCLBK void ovcExit (VESSEL ∗pVessel)

8.75.1 Function Documentation

8.75.1.1 DLLCLBK void ExitModule (HINSTANCE *hDLL*)

8.75.1.2 DLLCLBK void InitModule (HINSTANCE *hDLL*)

8.75.1.3 DLLCLBK void ovcExit (VESSEL ∗ *pVessel*)

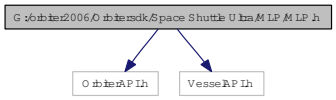
8.75.1.4 DLLCLBK VESSEL∗ ovcInit (OBJHANDLE *hVessel*, int *iFlightModel*)

Here is the call graph for this function:

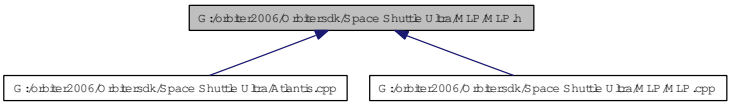
8.76 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/MLP/MLP.h File Reference

```
#include "OrbiterAPI.h"
#include "VesselAPI.h"
```

Include dependency graph for MLP.h:



This graph shows which files directly or indirectly include this file:



Classes

- class MLP

Variables

- const VECTOR3 POS_MPS_SMOKE = _V(0.0, -15.0, -20.0)
- const VECTOR3 POS_SRB_SMOKE = _V(0.0, -15.0, +20.0)
- const VECTOR3 DIR_MPS_SMOKE = _V(0.0, 0.0, -1.0)
- const VECTOR3 DIR_SRB_SMOKE = _V(0.0, 0.0, 1.0)
- const VECTOR3 HDP_POS = _V(0, -0.86, -4.25)

8.76.1 Variable Documentation

8.76.1.1 const VECTOR3 DIR_MPS_SMOKE = _V(0.0, 0.0, -1.0)

8.76.1.2 const VECTOR3 DIR_SRB_SMOKE = _V(0.0, 0.0, 1.0)

8.76.1.3 const VECTOR3 HDP_POS = _V(0, -0.86, -4.25)

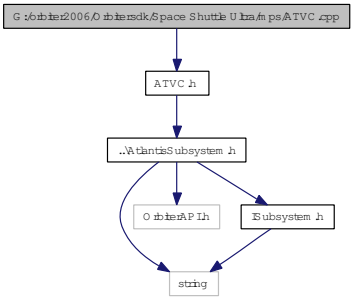
8.76.1.4 const VECTOR3 POS_MPS_SMOKE = _V(0.0, -15.0, -20.0)

8.76.1.5 const VECTOR3 POS_SRB_SMOKE = _V(0.0, -15.0, +20.0)

8.77 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/ATVC.cpp File Reference

```
#include "ATVC.h"
```

Include dependency graph for ATVC.cpp:



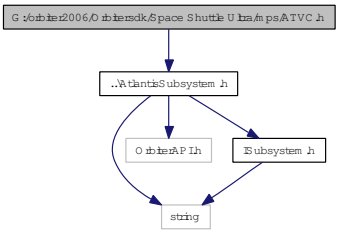
Namespaces

- namespace mps

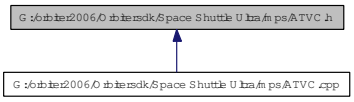
8.78 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/ATVC.h File Reference

```
#include "..\AtlantisSubsystem.h"
```

Include dependency graph for ATVC.h:



This graph shows which files directly or indirectly include this file:



Namespaces

- namespace mps

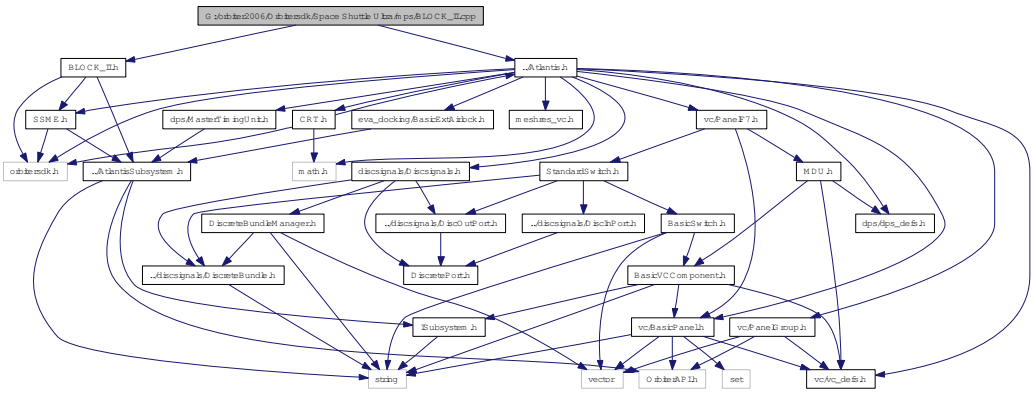
Classes

- class mps::ATVC

8.79 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/BLOCK_-II.cpp File Reference

```
#include "BLOCK_II.h"
#include "../Atlantis.h"
```

Include dependency graph for BLOCK_II.cpp:



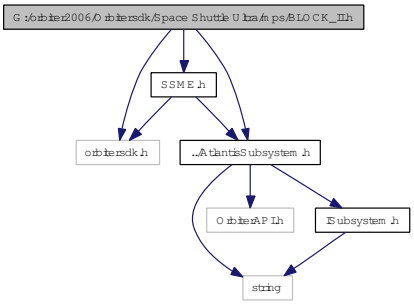
Namespaces

- namespace mps

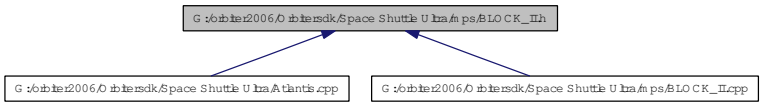
8.80 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/BLOCK_-II.h File Reference

```
#include <orbitersdk.h>
#include "../AtlantisSubsystem.h"
#include "SSME.h"
```

Include dependency graph for BLOCK_II.h:



This graph shows which files directly or indirectly include this file:



Namespaces

- namespace mps

Classes

- class mps::BLOCK_II

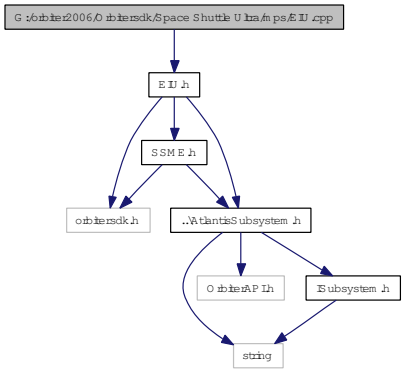
Variables

- const double mps::BLOCK_II_RPL_THRUST = 2090664.159
- const double mps::BLOCK_II_FPL_THRUST = 2275094.273
- const double mps::BLOCK_II_ISP0 = 453∗9.80665
- const double mps::BLOCK_II_ISP1 = 363∗9.80665

8.81 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/EIU.cpp File Reference

```
#include "EIU.h"
```

Include dependency graph for EIU.cpp:



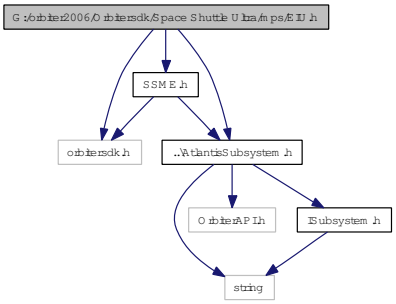
Namespaces

- namespace mps

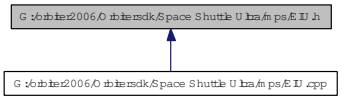
8.82 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/EIU.h File Reference

```
#include "SSME.h"
#include <orbitersdk.h>
#include "..\AtlantisSubsystem.h"
```

Include dependency graph for EIU.h:



This graph shows which files directly or indirectly include this file:



Namespaces

- namespace mps

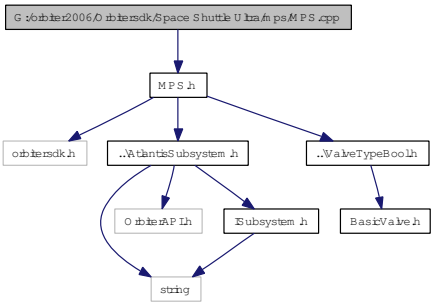
Classes

- class mps::EIU

8.83 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/MPS.cpp File Reference

```
#include "MPS.h"
```

Include dependency graph for MPS.cpp:



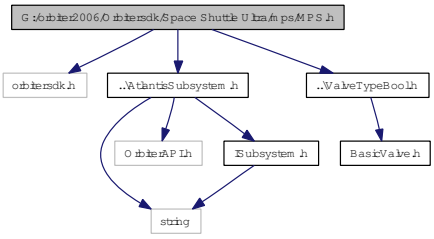
Namespaces

- namespace mps

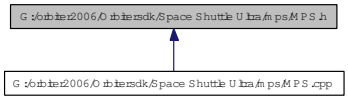
8.84 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/MPS.h File Reference

```
#include <orbitersdk.h>
#include "..\AtlantisSubsystem.h"
#include "..\ValveTypeBool.h"
```

Include dependency graph for MPS.h:



This graph shows which files directly or indirectly include this file:



Namespaces

- namespace mps

Classes

- class mps::MPS

Typedefs

- typedef enum mps::_MPS_VALVE mps::MPS_VALVE

Enumerations

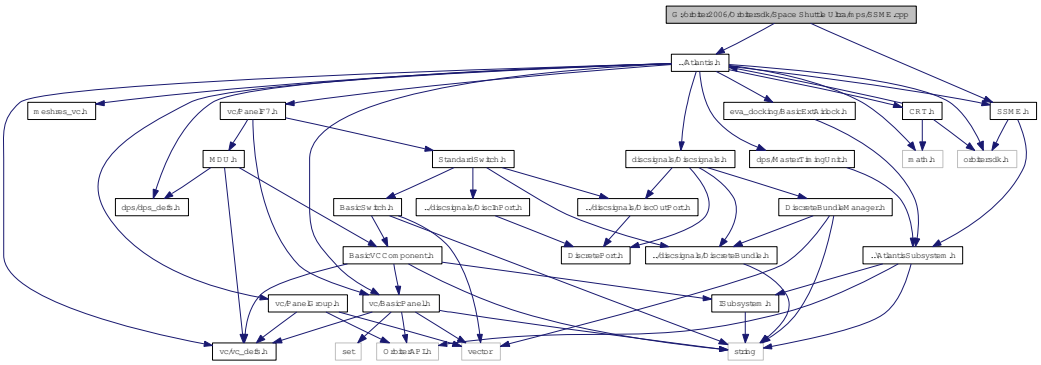
- enum mps::_MPS_VALVE {
  mps::PV1, mps::PV2, mps::PV3, mps::PV4,
  mps::PV5, mps::PV6, mps::PV7, mps::PV8,

mps::PV9, mps::PV10, mps::PV11, mps::PV12,
mps::PV13, mps::PV17, mps::PV18, mps::PV19,
mps::PV20, mps::PV21, mps::PV22, mps::PD1,
mps::PD2, mps::PD3 }

8.85 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/SSME.cpp File Reference

```
#include "SSME.h"
#include "../Atlantis.h"
```

Include dependency graph for SSME.cpp:



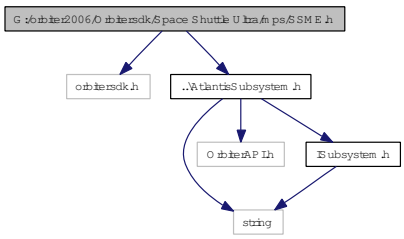
Namespaces

- namespace mps

8.86 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/SSME.h File Reference

```
#include <orbitersdk.h>
#include "..\AtlantisSubsystem.h"
```

Include dependency graph for SSME.h:



This graph shows which files directly or indirectly include this file:

Namespaces

- namespace mps

Classes

- struct mps::_DATA_TABLE
- class mps::SSME

Typedefs

- typedef enum mps::_ENGINE_STATUS mps::ENGINE_STATUS
- typedef enum mps::_DCU mps::DCU
- typedef struct mps::_DATA_TABLE mps::DATA_TABLE

Enumerations

- enum mps::_ENGINE_STATUS {

  mps::CHECKOUT_STANDBY, mps::CHECKOUT_COMPONENTCHECKOUT, mps::STARTPREPARATION_PURGESEQUENCE1, mps::STARTPREPARATION_-PURGESEQUENCE2,

  mps::STARTPREPARATION_PURGESEQUENCE3, mps::STARTPREPARATION_-PURGESEQUENCE4, mps::STARTPREPARATION_ENGINEREADY, mps::STARTMAINSTAGE_STARTPHASESTARTINITIATION,

mps::STARTMAINSTAGE_STARTPHASETHRUSTBUILDUP, mps::STARTMAINSTAGE_-MAINSTAGEPHASENORMALCONTROL, mps::STARTMAINSTAGE_FIXEDDENSITY, mps::STARTMAINSTAGE_THRUSTLIMITING,

mps::STARTMAINSTAGE_ELECTRICALLOCKUP, mps::STARTMAINSTAGE_-HYDRAULICLOCKUP, mps::SHUTDOWN_THROTTLINGTOZEROTHRUST, mps::SHUTDOWN_PROPELLANTVALVESCLOSED,

mps::SHUTDOWN_FAILSAFEPNEUMATIC, mps::POSTSHUTDOWN_-STANDBY, mps::POSTSHUTDOWN_OXIDIZERDUMP, mps::POSTSHUTDOWN_-TERMINATESEQUENCE,

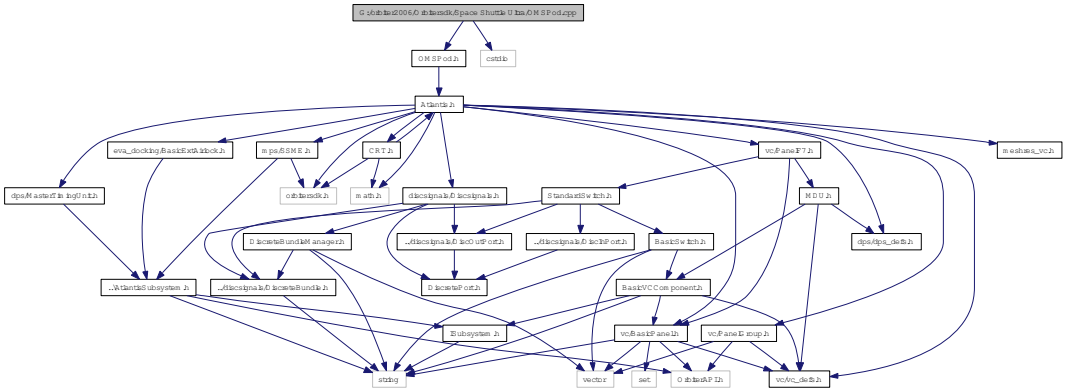
mps::PROM_STANDBY }

- enum mps::_DCU { mps::DCU_A, mps::DCU_B }

8.87 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/OMSPod.cpp File Reference

```
#include "OMSPod.h"
#include <cstdio>
```

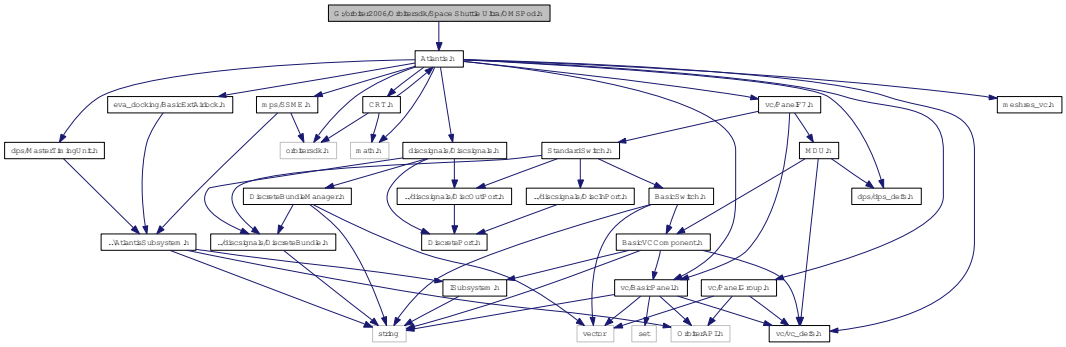
Include dependency graph for OMSPod.cpp:



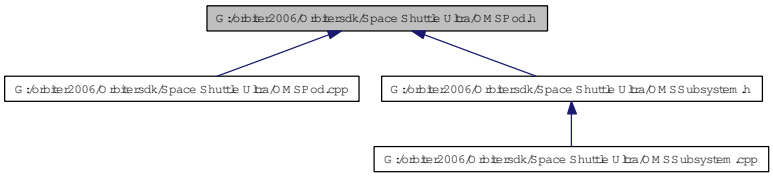
8.88 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/OMSPod.h File Reference

```
#include "Atlantis.h"
```

Include dependency graph for OMSPod.h:



This graph shows which files directly or indirectly include this file:



Classes

- class OMSPod

Variables

- const double N2_TANK_VOLUME = 300.0 / 1728
- const double N2_MOLARMASS = 28.0134/LBM
- const double GAS_CONSTANT = 10.7316
- const double N2_PIPE_AREA = 1.824E-6
- const double L_ISOLVLV = 1.0

8.88.1 Variable Documentation

8.88.1.1 const double GAS_CONSTANT = 10.7316

8.88.1.2 const double L_ISOLVLV = 1.0

8.88.1.3 const double N2_MOLARMASS = 28.0134/LBM

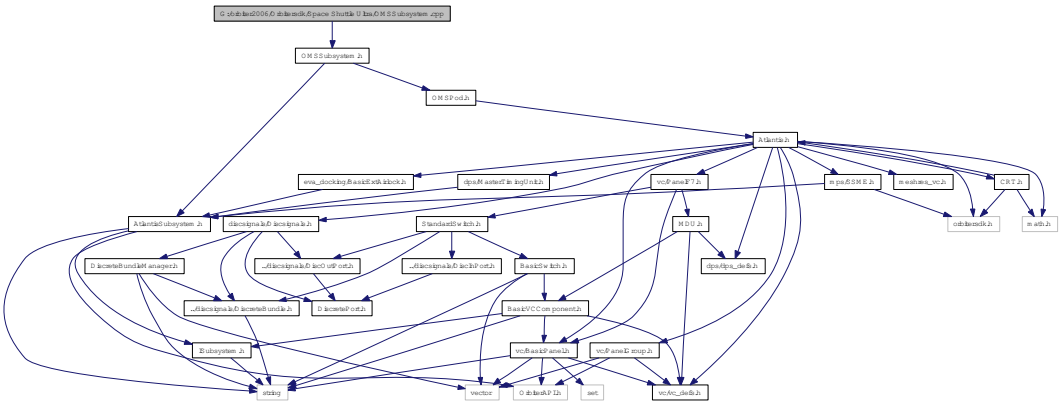
8.88.1.4 const double N2_PIPE_AREA = 1.824E-6

8.88.1.5 const double N2_TANK_VOLUME = 300.0 / 1728

8.89 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/OMSSubsystem.cpp File Reference

```
#include "OMSSubsystem.h"
```

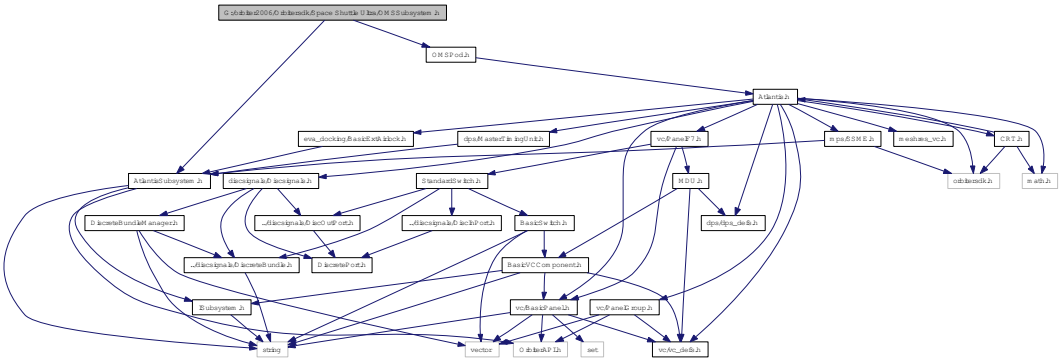
Include dependency graph for OMSSubsystem.cpp:



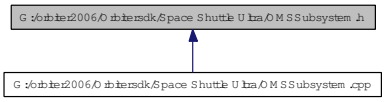
8.90 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/OMSSubsystem.h File Reference

```
#include "AtlantisSubsystem.h"
#include "OMSPod.h"
```

Include dependency graph for OMSSubsystem.h:



This graph shows which files directly or indirectly include this file:



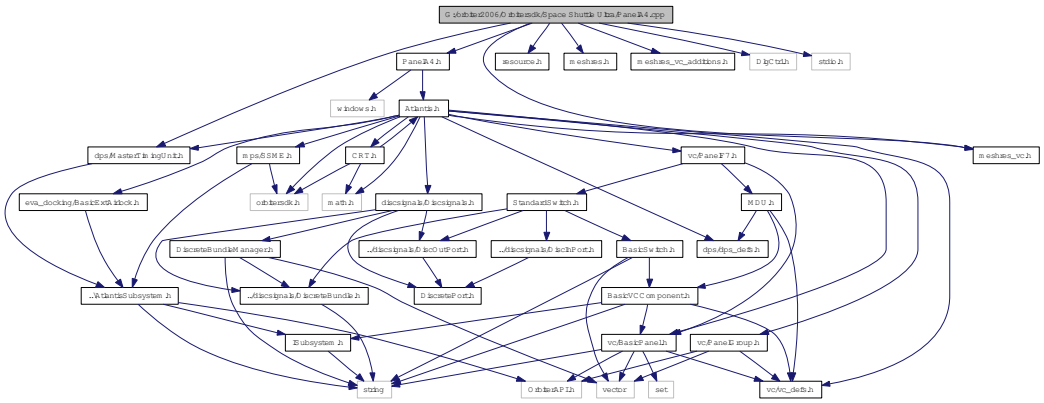
Classes

- class OMSSubsystem

8.91 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelA4.cpp File Reference

```
#include "PanelA4.h"
#include "resource.h"
#include "meshres.h"
#include "meshres_vc.h"
#include "meshres_vc_additions.h"
#include "DlgCtrl.h"
#include <stdio.h>
#include "dps/MasterTimingUnit.h"
```

Include dependency graph for PanelA4.cpp:



Variables

- GDIParams g_Param
- HELPCONTEXT g_hc
- char ∗ ActionString [5]

8.91.1 Variable Documentation

8.91.1.1 char∗ ActionString[5]

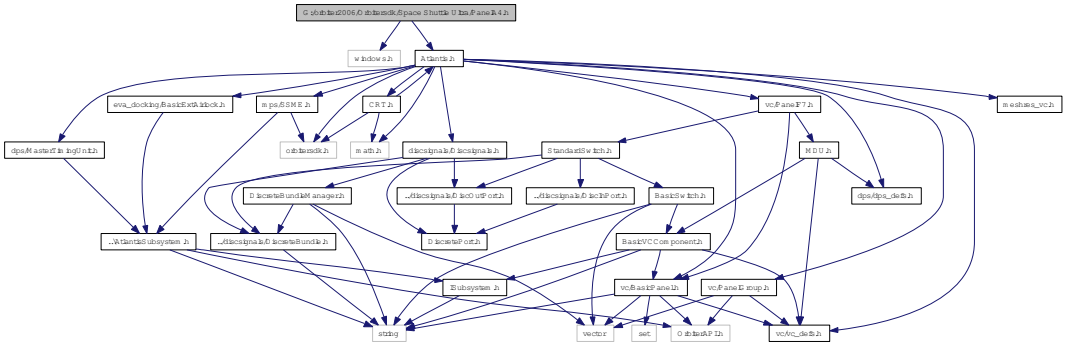
8.91.1.2 HELPCONTEXT g_hc

8.91.1.3 GDIParams g_Param

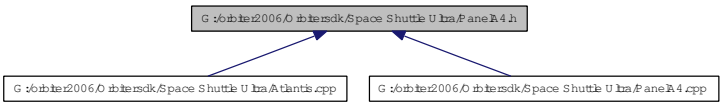
8.92 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelA4.h File Reference

```
#include <windows.h>
#include "Atlantis.h"
```

Include dependency graph for PanelA4.h:



This graph shows which files directly or indirectly include this file:



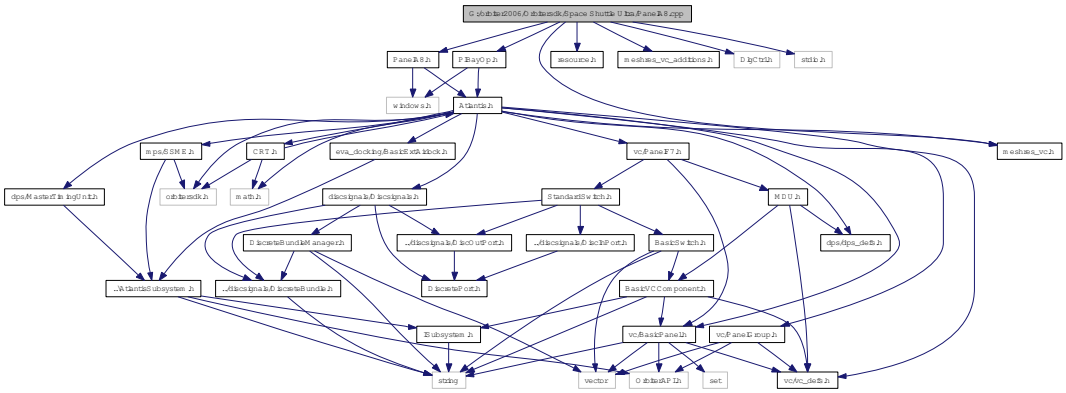
Classes

- class PanelA4

8.93 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelA8.cpp File Reference

```
#include "PanelA8.h"
#include "PlBayOp.h"
#include "resource.h"
#include "meshres_vc.h"
#include "meshres_vc_additions.h"
#include "DlgCtrl.h"
#include <stdio.h>
```

Include dependency graph for PanelA8.cpp:



Variables

- GDIParams g_Param
- HELPCONTEXT g_hc
- char ∗ ActionString [5]

8.93.1 Variable Documentation

8.93.1.1 char∗ ActionString[5]

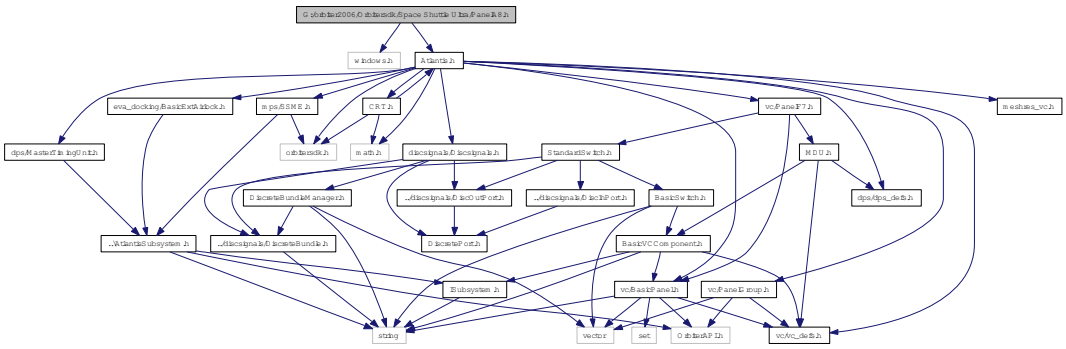
8.93.1.2 HELPCONTEXT g_hc

8.93.1.3 GDIParams g_Param

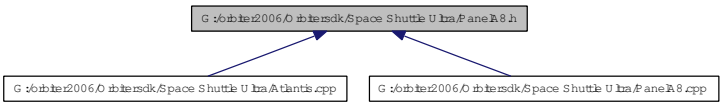
8.94 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelA8.h File Reference

```
#include <windows.h>
#include "Atlantis.h"
```

Include dependency graph for PanelA8.h:



This graph shows which files directly or indirectly include this file:



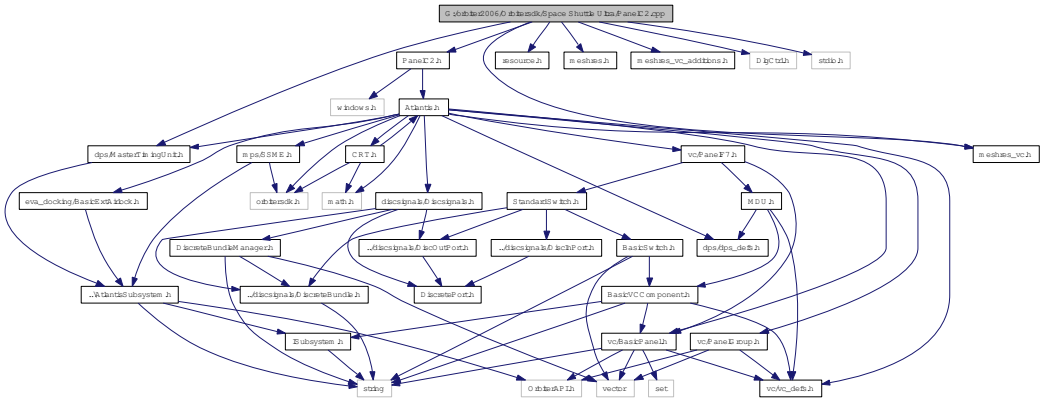
Classes

- class PanelA8

8.95 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelC2.cpp File Reference

```
#include "PanelC2.h"
#include "resource.h"
#include "meshres.h"
#include "meshres_vc.h"
#include "meshres_vc_additions.h"
#include "DlgCtrl.h"
#include <stdio.h>
#include "dps/MasterTimingUnit.h"
```

Include dependency graph for PanelC2.cpp:



Variables

- GDIParams g_Param
- HELPCONTEXT g_hc
- char * ActionString [5]
- const double EVTTMR_WHEELMAX [4] = {3.0, 5.0, 3.0, 5.0}
- const UINT EVTTMR_WINDOW_GROUPS [4] = {211, 212, 213, 214}

8.95.1 Variable Documentation

8.95.1.1 char* ActionString[5]

8.95.1.2 const double EVTTMR_WHEELMAX[4] = {3.0, 5.0, 3.0, 5.0}

8.95.1.3 const UINT EVTTMR_WINDOW_GROUPS[4] = {211, 212, 213, 214}

8.95.1.4 HELPCONTEXT g_hc

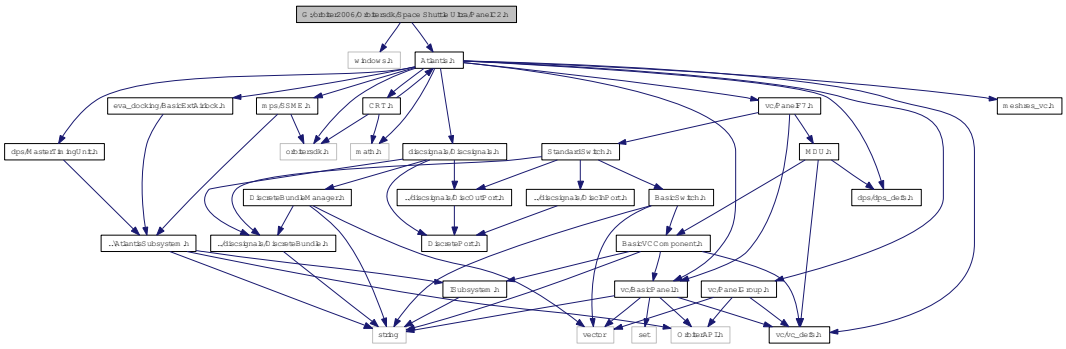
8.95.1.5 GDIParams g_Param

8.96 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelC2.h File Reference

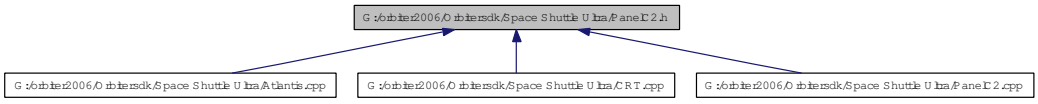
```
#include <windows.h>
```

```
#include "Atlantis.h"
```

Include dependency graph for PanelC2.h:



This graph shows which files directly or indirectly include this file:



Classes

- class PanelC2

Variables

- const VECTOR3 REF_C2S1 = _V(-0.10, 1.911417, 14.50198)
- const VECTOR3 REF_C2S2 = _V(-0.07, 1.911417, 14.50198)
- const VECTOR3 REF_C2S3 = _V(-0.02, 1.911417, 14.50198)
- const VECTOR3 REF_C2S4 = _V(0.01, 1.911417, 14.50198)
- const VECTOR3 REF_C2S5 = _V(0.07, 1.911417, 14.50198)
- const VECTOR3 REF_C2S6 = _V(-0.10, 1.911417, 14.50198)
- const VECTOR3 REF_C2S7 = _V(-0.100152, 1.859752, 14.45152)
- const VECTOR3 REF_C2S8 = _V(0.100152, 1.859752, 14.45152)
- const VECTOR3 REF_C2S9 = _V(0.0, 1.801141, 14.3961)
- const VECTOR3 REF_C2S10 = _V(-0.07, 1.801141, 14.3961)
- const VECTOR3 REF_C2S11 = _V(0.131577, 1.80661, 14.42035)
- const VECTOR3 REF_C2S12 = _V(0.0994235, 1.801141, 14.3961)

8.96.1 Variable Documentation

8.96.1.1 const VECTOR3 REF_C2S1 = _V(-0.10, 1.911417, 14.50198)

8.96.1.2 const VECTOR3 REF_C2S10 = _V(-0.07, 1.801141, 14.3961)

8.96.1.3 const VECTOR3 REF_C2S11 = _V(0.131577, 1.80661, 14.42035)

8.96.1.4 const VECTOR3 REF_C2S12 = _V(0.0994235, 1.801141, 14.3961)

8.96.1.5 const VECTOR3 REF_C2S2 = _V(-0.07, 1.911417, 14.50198)

8.96.1.6 const VECTOR3 REF_C2S3 = _V(-0.02, 1.911417, 14.50198)

8.96.1.7 const VECTOR3 REF_C2S4 = _V(0.01, 1.911417, 14.50198)

8.96.1.8 const VECTOR3 REF_C2S5 = _V(0.07, 1.911417, 14.50198)

8.96.1.9 const VECTOR3 REF_C2S6 = _V(-0.10, 1.911417, 14.50198)

8.96.1.10 const VECTOR3 REF_C2S7 = _V(-0.100152, 1.859752, 14.45152)

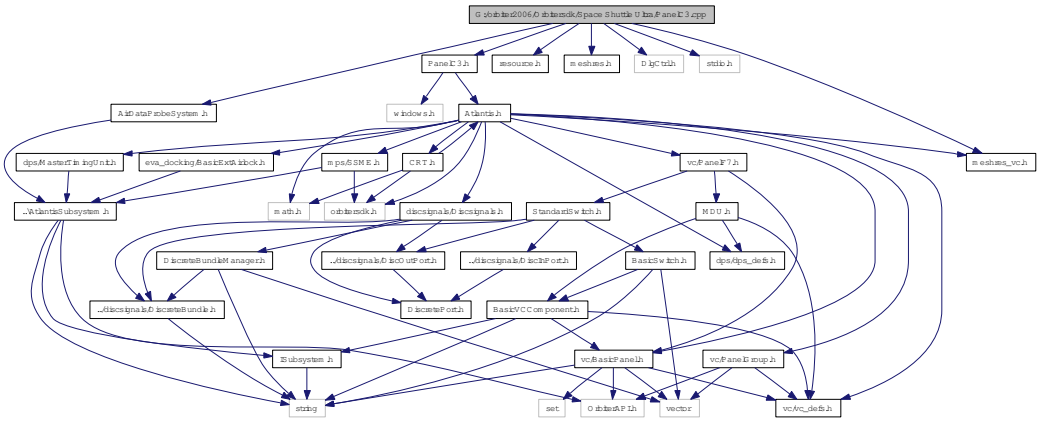
8.96.1.11 const VECTOR3 REF_C2S8 = _V(0.100152, 1.859752, 14.45152)

8.96.1.12 const VECTOR3 REF_C2S9 = _V(0.0, 1.801141, 14.3961)

8.97 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelC3.cpp File Reference

```
#include "PanelC3.h"
#include "resource.h"
#include "meshres.h"
#include "meshres_vc.h"
#include "DlgCtrl.h"
#include <stdio.h>
#include "AirDataProbeSystem.h"
```

Include dependency graph for PanelC3.cpp:



Variables

- GDIParams g_Param
- HELPCONTEXT g_hc
- char ∗ ActionString [5]

8.97.1 Variable Documentation

8.97.1.1 char∗ ActionString[5]

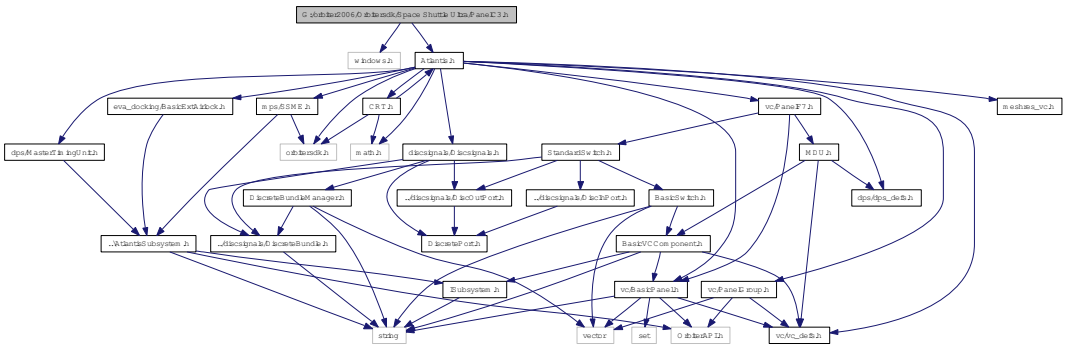
8.97.1.2 HELPCONTEXT g_hc

8.97.1.3 GDIParams g_Param

8.98 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelC3.h File Reference

```
#include <windows.h>
#include "Atlantis.h"
```

Include dependency graph for PanelC3.h:



This graph shows which files directly or indirectly include this file:

Classes

- class PanelC3

Variables

- const VECTOR3 REF_C3S1 = _V(-0.2114868, 1.728119, 14.29085)
- const VECTOR3 REF_C3S2 = _V(-0.1716415, 1.728119, 14.29085)
- const VECTOR3 REF_C3S19 = _V(-0.2114868, 1.715764, 14.18536)
- const VECTOR3 REF_C3S20 = _V(-0.1716415, 1.715764, 14.18536)
- const VECTOR3 REF_C3S8 = _V(-0.2114868, 1.680126, 13.8549)
- const VECTOR3 REF_C3S9 = _V(-0.1716415, 1.680126, 13.8549)

8.98.1 Variable Documentation

8.98.1.1 const VECTOR3 REF_C3S1 = _V(-0.2114868, 1.728119, 14.29085)

8.98.1.2 const VECTOR3 REF_C3S19 = _V(-0.2114868, 1.715764, 14.18536)

8.98.1.3 const VECTOR3 REF_C3S2 = _V(-0.1716415, 1.728119, 14.29085)

8.98.1.4 const VECTOR3 REF_C3S20 = _V(-0.1716415, 1.715764, 14.18536)

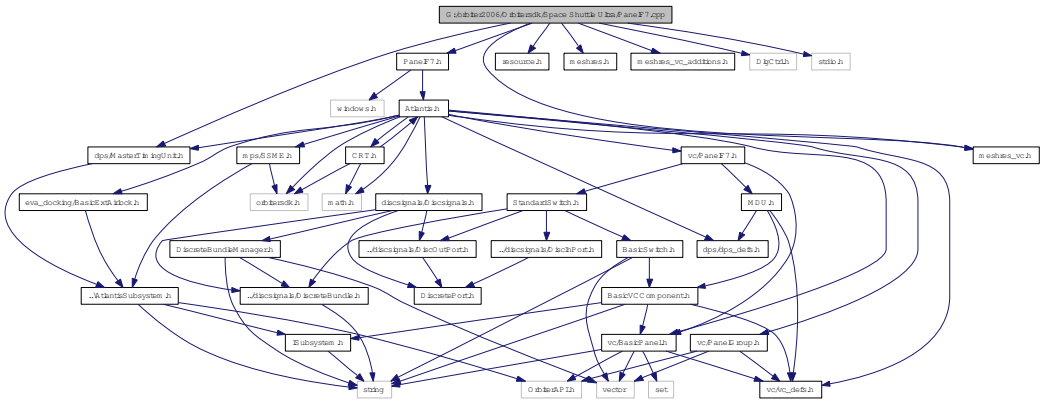
8.98.1.5 const VECTOR3 REF_C3S8 = _V(-0.2114868, 1.680126, 13.8549)

8.98.1.6 const VECTOR3 REF_C3S9 = _V(-0.1716415, 1.680126, 13.8549)

8.99 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelF7.cpp File Reference

```
#include "PanelF7.h"
#include "resource.h"
#include "meshres.h"
#include "meshres_vc.h"
#include "meshres_vc_additions.h"
#include "DlgCtrl.h"
#include <stdio.h>
#include "dps/MasterTimingUnit.h"
```

Include dependency graph for PanelF7.cpp:



Variables

- GDIParams g_Param
- HELPCONTEXT g_hc
- char * ActionString [5]

8.99.1 Variable Documentation

8.99.1.1 char* ActionString[5]

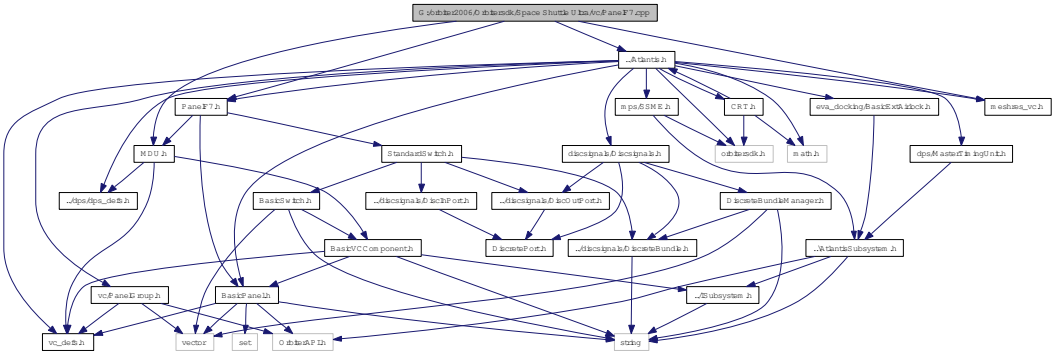
8.99.1.2 HELPCONTEXT g_hc

8.99.1.3 GDIParams g_Param

8.100 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelF7.cpp File Reference

```
#include "PanelF7.h"
#include "../Atlantis.h"
#include "../meshres_vc.h"
#include "MDU.h"
```

Include dependency graph for PanelF7.cpp:



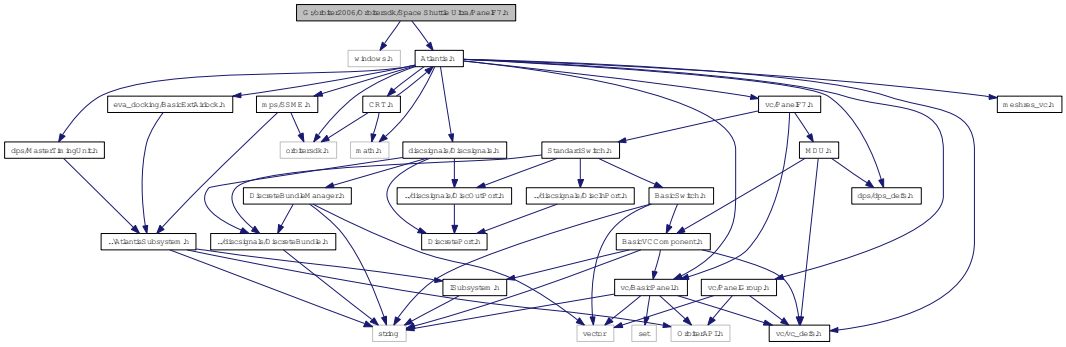
Namespaces

- namespace vc

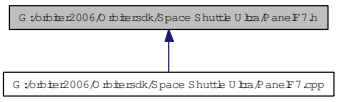
8.101 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelF7.h File Reference

```
#include <windows.h>
#include "Atlantis.h"
```

Include dependency graph for PanelF7.h:



This graph shows which files directly or indirectly include this file:



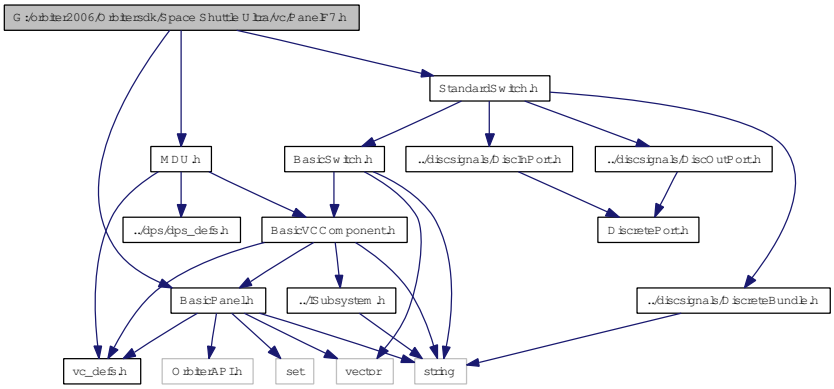
Classes

- class PanelF7

8.102 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelF7.h File Reference

```
#include "BasicPanel.h"
#include "MDU.h"
#include "StandardSwitch.h"
```

Include dependency graph for PanelF7.h:



This graph shows which files directly or indirectly include this file:

Namespaces

- namespace vc

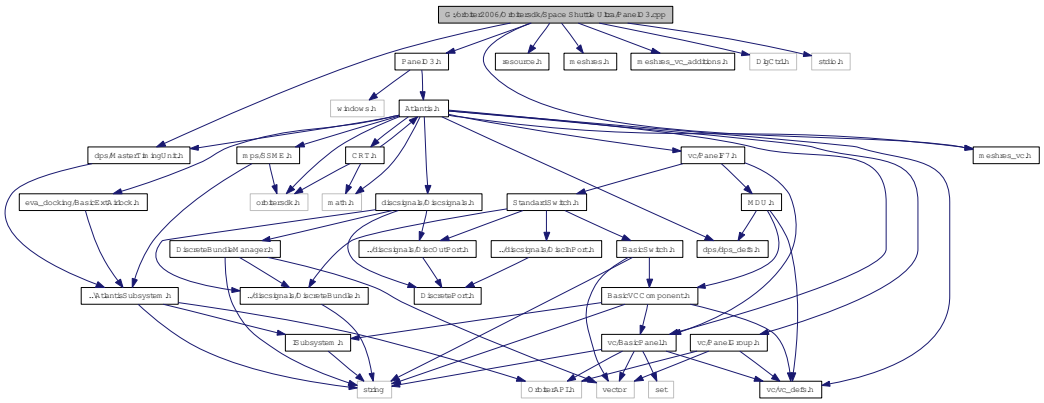
Classes

- class vc::PanelF7

8.103 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelO3.cpp File Reference

```
#include "PanelO3.h"
#include "resource.h"
#include "meshres.h"
#include "meshres_vc.h"
#include "meshres_vc_additions.h"
#include "DlgCtrl.h"
#include <stdio.h>
#include "dps/MasterTimingUnit.h"
```

Include dependency graph for PanelO3.cpp:



Variables

- GDIParams g_Param
- HELPCONTEXT g_hc
- char ∗ ActionString [5]

8.103.1 Variable Documentation

8.103.1.1 char∗ ActionString[5]

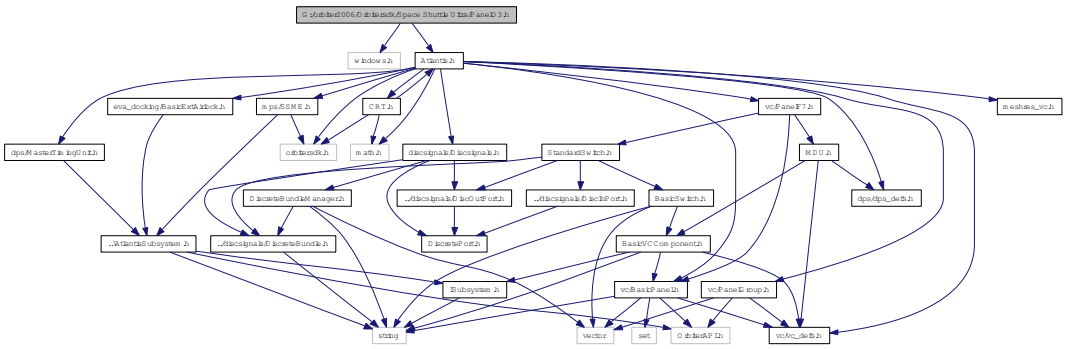
8.103.1.2 HELPCONTEXT g_hc

8.103.1.3 GDIParams g_Param

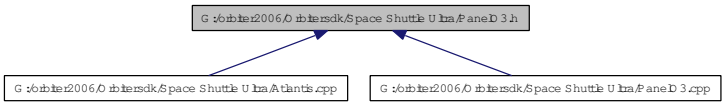
8.104 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelO3.h File Reference

```
#include <windows.h>
#include "Atlantis.h"
```

Include dependency graph for PanelO3.h:



This graph shows which files directly or indirectly include this file:



Classes

- class PanelO3

Defines

- #define NUM_O3_SWITCHES 20
- #define NUM_O3_ANIMATIONS 10

Variables

- const VECTOR3 REF_O3S12 = _V(0.5121638, 2.799416, 14.35993)
- const int SWITCH_O3S12 = 0

8.104.1 Define Documentation

8.104.1.1 #define NUM_O3_ANIMATIONS 10

8.104.1.2 #define NUM_O3_SWITCHES 20

8.104.2 Variable Documentation

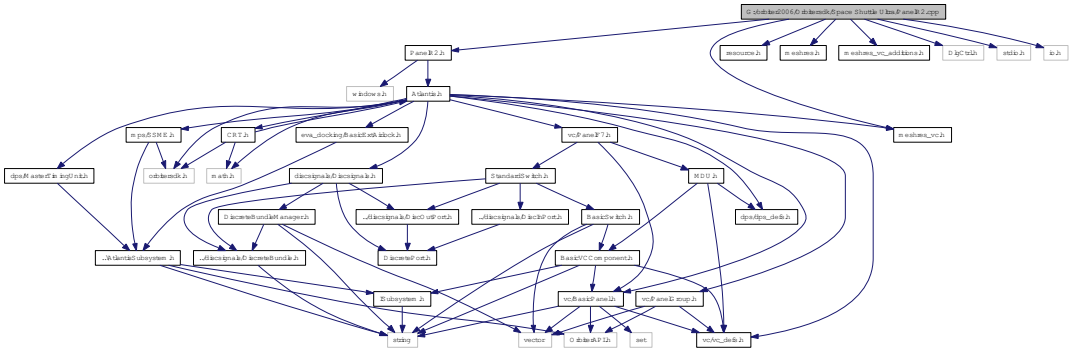
8.104.2.1 const VECTOR3 REF_O3S12 = _V(0.5121638, 2.799416, 14.35993)

8.104.2.2 const int SWITCH_O3S12 = 0

8.105 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelR2.cpp File Reference

```
#include "PanelR2.h"
#include "resource.h"
#include "meshres.h"
#include "meshres_vc.h"
#include "meshres_vc_additions.h"
#include "DlgCtrl.h"
#include <stdio.h>
#include <io.h>
```

Include dependency graph for PanelR2.cpp:



Variables

- GDIParams g_Param
- HELPCONTEXT g_hc
- char ∗ ActionString [5]

8.105.1 Variable Documentation

8.105.1.1 char∗ ActionString[5]

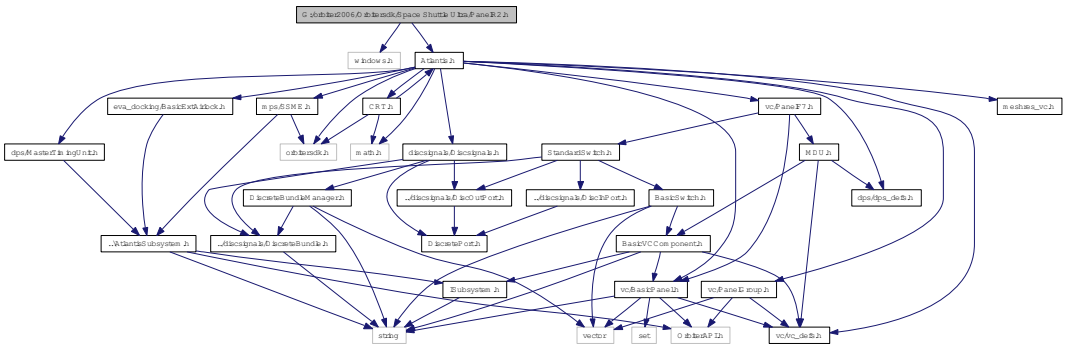
8.105.1.2 HELPCONTEXT g_hc

8.105.1.3 GDIParams g_Param

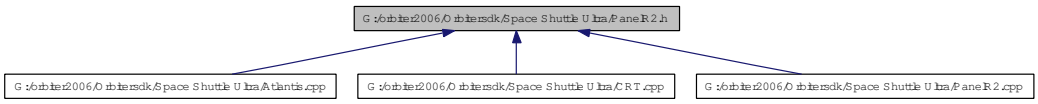
8.106 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelR2.h File Reference

```
#include <windows.h>
#include "Atlantis.h"
```

Include dependency graph for PanelR2.h:



This graph shows which files directly or indirectly include this file:



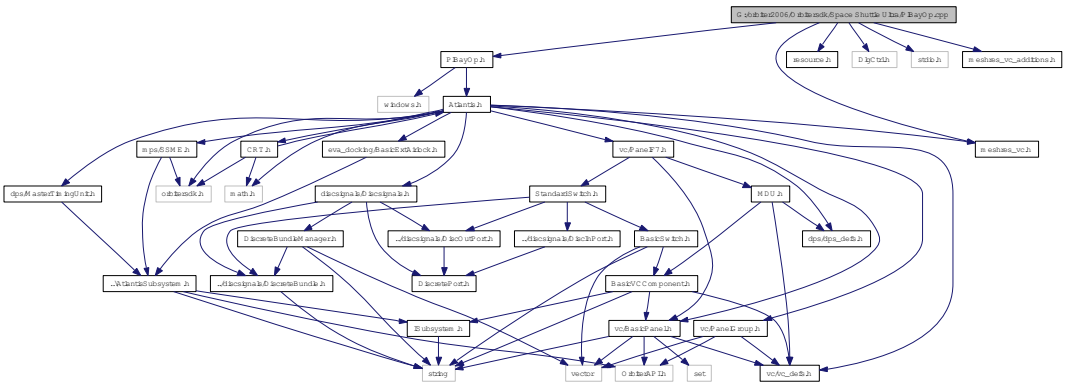
Classes

- class PanelR2

8.107 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PlBayOp.cpp File Reference

```
#include "PlBayOp.h"
#include "resource.h"
#include "meshres_vc.h"
#include "DlgCtrl.h"
#include <stdio.h>
#include "meshres_vc_additions.h"
```

Include dependency graph for PlBayOp.cpp:



Functions

- BOOL CALLBACK PlOp_DlgProc (HWND hWnd, UINT uMsg, WPARAM wParam, LPARAM lParam)

Variables

- GDIParams g_Param
- HELPCONTEXT g_hc
- char * ActionString [5]
- static Atlantis * sts_dlg

8.107.1 Function Documentation

8.107.1.1 BOOL CALLBACK PlOp_DlgProc (HWND *hWnd*, UINT *uMsg*, WPARAM *wParam*, LPARAM *lParam*)

8.107.2 Variable Documentation

8.107.2.1 char* ActionString[5]

8.107.2.2 HELPCONTEXT g_hc

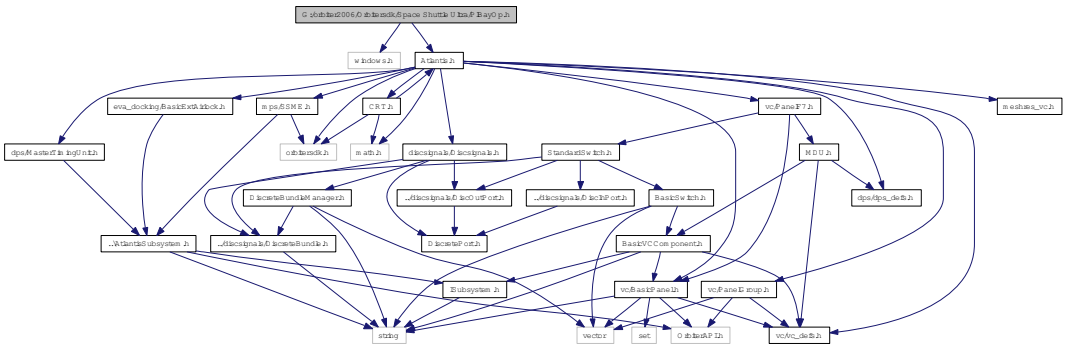
8.107.2.3 GDIParams g_Param

8.107.2.4 Atlantis* sts_dlg [static]

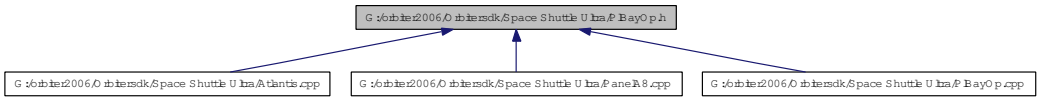
8.108 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PlBayOp.h File Reference

```
#include <windows.h>
#include "Atlantis.h"
```

Include dependency graph for PlBayOp.h:



This graph shows which files directly or indirectly include this file:



Classes

- class PayloadBayOp

8.109 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/resource.h File Reference

This graph shows which files directly or indirectly include this file:

Defines

- #define IDD_RMS 101
- #define IDI_UP 105
- #define IDI_DOWN 106
- #define IDI_LEFT 107
- #define IDI_RIGHT 108
- #define IDI_RRIGHT 109
- #define IDI_RLEFT 110
- #define IDD_PLBAY 111
- #define IDD_CTRL 112
- #define IDB_TKBKLABEL 113
- #define IDD_PAYLOADCAM 114
- #define IDB_CLOCKDIGITS 115
- #define IDB_7SEGDIGITS 116
- #define IDB_DEUCHARACTERS 117
- #define IDC_BUTTON1 1000
- #define IDC_WRIST_ROLLRIGHT 1000
- #define IDC_PLBAYOP 1000
- #define IDC_BUTTON2 1001
- #define IDC_WRIST_YAWLEFT 1001
- #define IDC_RMSOP 1001
- #define IDC_WRIST_PITCHUP 1002
- #define IDC_WRIST_PITCHDOWN 1003
- #define IDC_BUTTON5 1004
- #define IDC_WRIST_YAWRIGHT 1004
- #define IDC_BUTTON6 1005
- #define IDC_WRIST_ROLLLEFT 1005
- #define IDC_DEG01 1006
- #define IDC_STOW 1007
- #define IDC_GRAPPLE 1008
- #define IDC_SHOWGRAPPLE 1009
- #define IDC_RELEASE 1010
- #define IDC_PAYLOAD 1011
- #define IDC_BUTTON9 1024
- #define IDC_ELBOW_PITCHUP 1024
- #define IDC_BUTTON10 1025
- #define IDC_ELBOW_PITCHDOWN 1025
- #define IDC_BUTTON14 1029
- #define IDC_SHOULDER_YAWLEFT 1029
- #define IDC_BUTTON15 1030

- #define IDC_SHOULDER_PITCHUP 1030
- #define IDC_BUTTON16 1031
- #define IDC_SHOULDER_PITCHDOWN 1031
- #define IDC_BUTTON17 1032
- #define IDC_SHOULDER_YAWRIGHT 1032
- #define IDC_PLBD_TLKBK 1033
- #define IDC_RADS_TLKBK 1034
- #define IDC_DEG1 1035
- #define IDC_RADP_TLKBK 1035
- #define IDC_DEG10 1036
- #define IDC_LATCHS_TLKBK 1036
- #define IDC_LATCHP_TLKBK 1037
- #define IDC_LATCHA 1038
- #define IDC_LATCHB 1039
- #define IDC_RADA 1040
- #define IDC_RADB 1041
- #define IDC_MECH1 1042
- #define IDC_MECH2 1043
- #define IDC_PLBD1 1044
- #define IDC_PLBD2 1045
- #define IDC_PLBD 1046
- #define IDC_KU_TLKBK 1047
- #define IDC_KU 1048
- #define IDC_KU_DIRECT 1049
- #define IDC_TRANS_PX 1050
- #define IDC_TRANS_PY 1051
- #define IDC_TRANS_PZ 1052
- #define IDC_CAM_SEL 1052
- #define IDC_TRANS_MX 1053
- #define IDC_CAM_BL 1053
- #define IDC_TRANS_MY 1054
- #define IDC_CAM_FL 1054
- #define IDC_TRANS_MZ 1055
- #define IDC_CAM_BR 1055
- #define IDC_CAM_FR 1056
- #define IDC_CAM_ROT 1057
- #define IDC_CAM_UP 1058
- #define IDC_CAM_DOWN 1059
- #define IDC_CAM_LEFT 1060
- #define IDC_CAM_RIGHT 1061
- #define IDC_PAYCAM 1063
- #define IDC_CAM_LOWRATE 1064
- #define IDC_CAM_HIGHRATE 1065

8.109.1 Define Documentation

8.109.1.1 #define IDB_7SEGDIGITS 116

8.109.1.2 #define IDB_CLOCKDIGITS 115

8.109.1.3 #define IDB_DEUCHARACTERS 117

8.109.1.4 #define IDB_TKBKLABEL 113

8.109.1.5 #define IDC_BUTTON1 1000

8.109.1.6 #define IDC_BUTTON10 1025

8.109.1.7 #define IDC_BUTTON14 1029

8.109.1.8 #define IDC_BUTTON15 1030

8.109.1.9 #define IDC_BUTTON16 1031

8.109.1.10 #define IDC_BUTTON17 1032

8.109.1.11 #define IDC_BUTTON2 1001

8.109.1.12 #define IDC_BUTTON5 1004

8.109.1.13 #define IDC_BUTTON6 1005

8.109.1.14 #define IDC_BUTTON9 1024

8.109.1.15 #define IDC_CAM_BL 1053

8.109.1.16 #define IDC_CAM_BR 1055

8.109.1.17 #define IDC_CAM_DOWN 1059

8.109.1.18 #define IDC_CAM_FL 1054

8.109.1.19 #define IDC_CAM_FR 1056

8.109.1.20 #define IDC_CAM_HIGHRATE 1065

8.109.1.21 #define IDC_CAM_LEFT 1060

8.109.1.22 #define IDC_CAM_LOWRATE 1064

8.109.1.23 #define IDC_CAM_RIGHT 1061

8.109.1.24 #define IDC_CAM_ROT 1057

8.109.1.25 #define IDC_CAM_SEL 1052

8.109.1.26 #define IDC_CAM_UP 1058

8.109.1.27 #define IDC_DEG01 1006

8.109.1.28 #define IDC_DEG1 1035

8.109.1.29 #define IDC_DEG10 1036

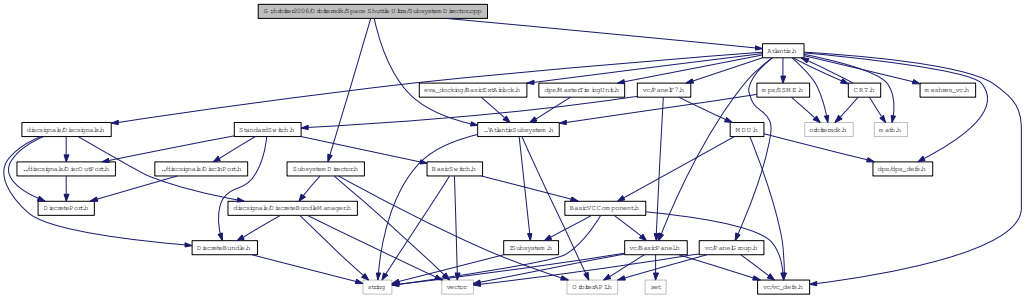
8.109.1.30 #define IDC_ELBOW_PITCHDOWN 1025

8.110 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/resource_-RMS.h File Reference

8.111 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/SubsystemDirector.cpp File Reference

```
#include "SubsystemDirector.h"
#include "Atlantis.h"
#include "AtlantisSubsystem.h"
```

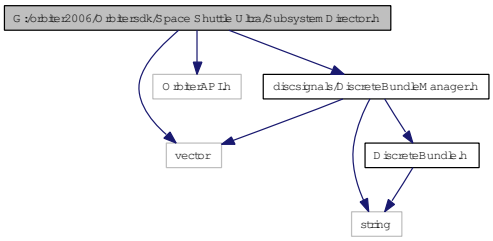
Include dependency graph for SubsystemDirector.cpp:



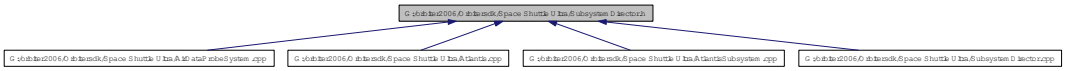
8.112 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/SubsystemDirector.h File Reference

```
#include <vector>
#include "OrbiterAPI.h"
#include "discsignals/DiscreteBundleManager.h"
```

Include dependency graph for SubsystemDirector.h:



This graph shows which files directly or indirectly include this file:



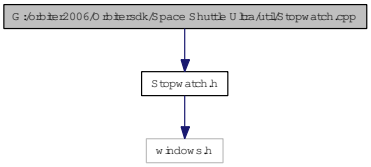
Classes

- class SubsystemDirector

8.113 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/util/Stopwatch.cpp File Reference

```
#include "Stopwatch.h"
```

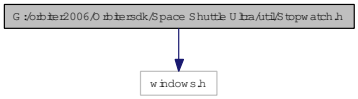
Include dependency graph for Stopwatch.cpp:



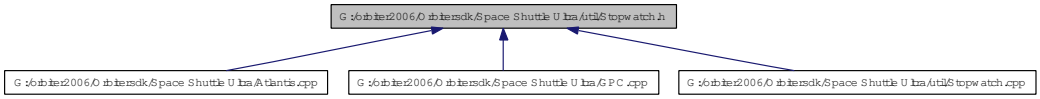
8.114 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/util/Stopwatch.h File Reference

```
#include "windows.h"
```

Include dependency graph for Stopwatch.h:



This graph shows which files directly or indirectly include this file:



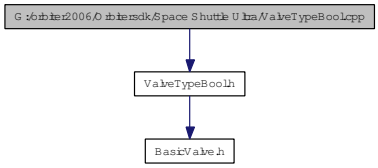
Classes

- class Stopwatch

8.115 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/ValveTypeBool.cpp File Reference

```
#include "ValveTypeBool.h"
```

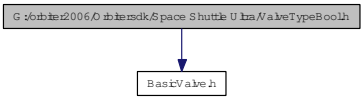
Include dependency graph for ValveTypeBool.cpp:



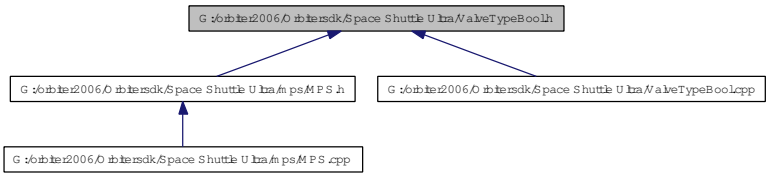
8.116 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/ValveTypeBool.h File Reference

```
#include "BasicValve.h"
```

Include dependency graph for ValveTypeBool.h:



This graph shows which files directly or indirectly include this file:



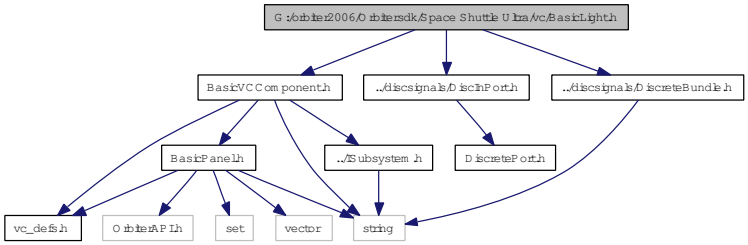
Classes

- class ValveTypeBool

8.117 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicLight.h File Reference

```
#include "BasicVCComponent.h"
#include "../discsignals/DiscInPort.h"
#include "../discsignals/DiscreteBundle.h"
```

Include dependency graph for BasicLight.h:



Namespaces

- namespace vc

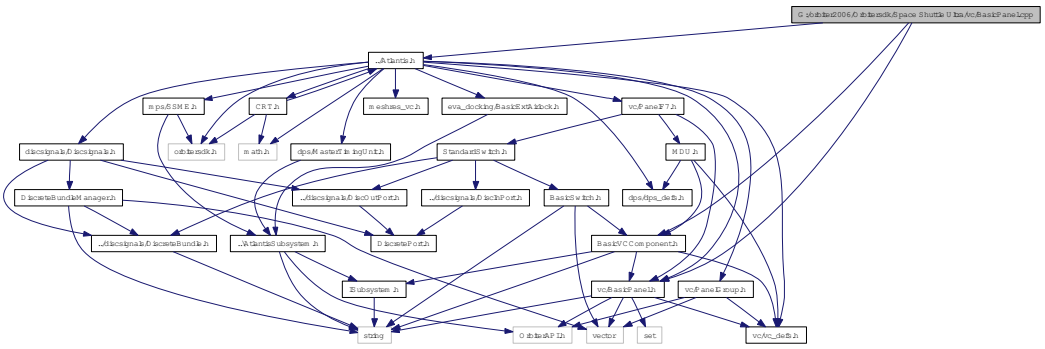
Classes

- class vc::BasicLight

8.118 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicPanel.cpp File Reference

```
#include "../Atlantis.h"
#include "BasicPanel.h"
#include "BasicVCComponent.h"
```

Include dependency graph for BasicPanel.cpp:



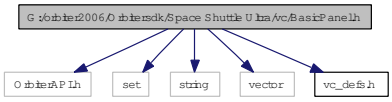
Namespaces

- namespace vc

8.119 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicPanel.h File Reference

```
#include "OrbiterAPI.h"
#include <set>
#include <string>
#include <vector>
#include "vc_defs.h"
```

Include dependency graph for BasicPanel.h:



This graph shows which files directly or indirectly include this file:

Namespaces

- namespace vc

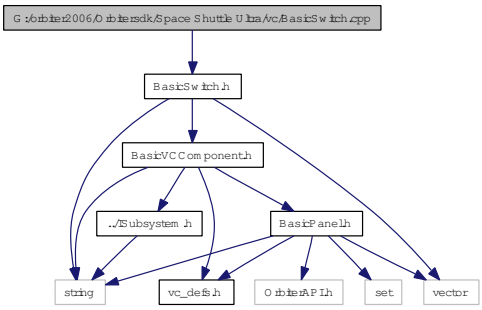
Classes

- class vc::BasicPanel

8.120 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicSwitch.cpp File Reference

```
#include "BasicSwitch.h"
```

Include dependency graph for BasicSwitch.cpp:



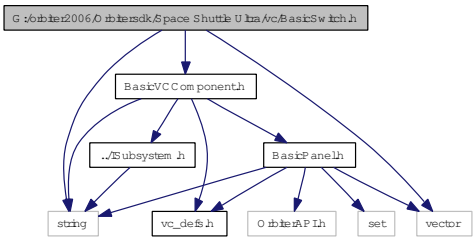
Namespaces

- namespace vc

8.121 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicSwitch.h File Reference

```
#include "BasicVCComponent.h"
#include <vector>
#include <string>
```

Include dependency graph for BasicSwitch.h:



This graph shows which files directly or indirectly include this file:

Namespaces

- namespace vc

Classes

- class vc::BasicSwitch

Defines

- #define AFX_BASICSWITCH_H__3790BCEE_A9B9_492A_A7F4_B0DA5E454AFB__INCLUDED_

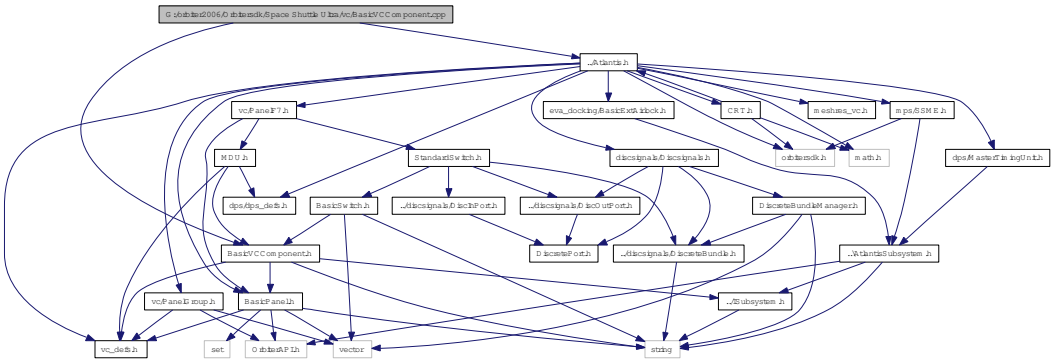
8.121.1 Define Documentation

8.121.1.1 #define AFX_BASICSWITCH_H__3790BCEE_A9B9_492A_A7F4_B0DA5E454AFB__INCLUDED_

8.122 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicVCComponent.cpp File Reference

```
#include "BasicVCComponent.h"
#include "../Atlantis.h"
```

Include dependency graph for BasicVCComponent.cpp:



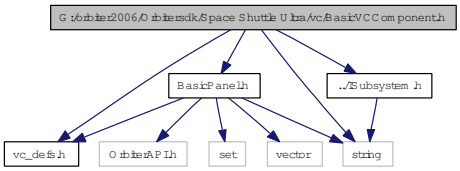
Namespaces

- namespace vc

8.123 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicVCComponent.h File Reference

```
#include "vc_defs.h"
#include "BasicPanel.h"
#include "../ISubsystem.h"
#include <string>
```

Include dependency graph for BasicVCComponent.h:



This graph shows which files directly or indirectly include this file:

Namespaces

- namespace vc

Classes

- class vc::BasicVCComponent

Defines

- #define AFX_BASICVCCOMPONENT_H__B74A68D7_2919_4C3F_AB0C_4B4FBA3AA2B7_-_INCLUDED_

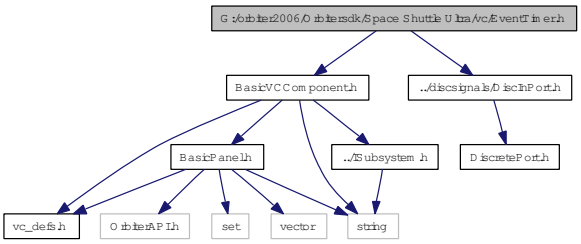
8.123.1 Define Documentation

8.123.1.1 #define AFX_BASICVCCOMPONENT_H__B74A68D7_2919_4C3F_AB0C_-4B4FBA3AA2B7__INCLUDED_

8.124 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/EventTimer.h File Reference

```
#include "BasicVCComponent.h"
#include "../discsignals/DiscInPort.h"
```

Include dependency graph for EventTimer.h:



Namespaces

- namespace vc

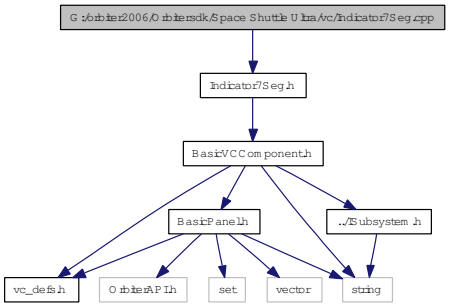
Classes

- class vc::EventTimer

8.125 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/Indicator7Seg.cpp File Reference

```
#include "Indicator7Seg.h"
```

Include dependency graph for Indicator7Seg.cpp:



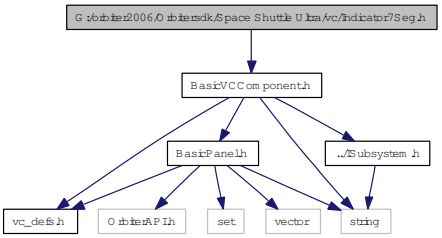
Namespaces

- namespace vc

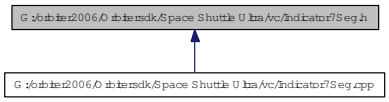
8.126 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/Indicator7Seg.h File Reference

`#include "BasicVCComponent.h"`

Include dependency graph for Indicator7Seg.h:



This graph shows which files directly or indirectly include this file:



Namespaces

- namespace vc

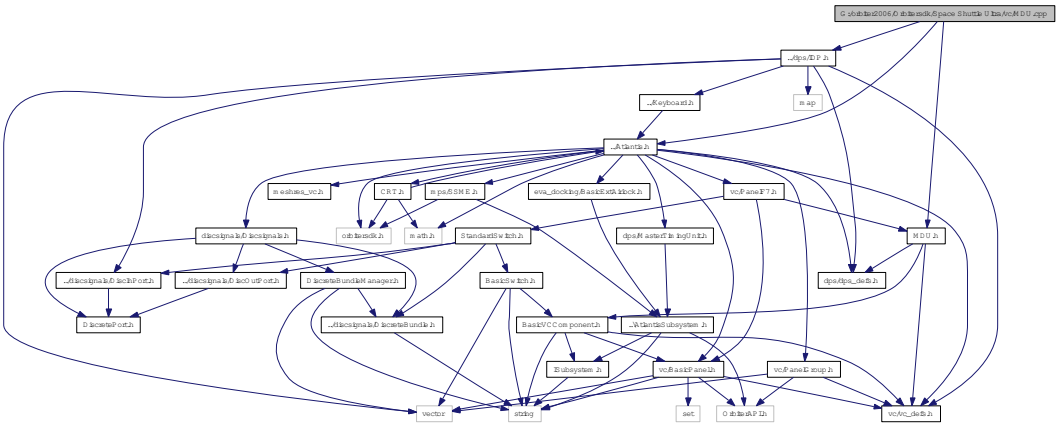
Classes

- class vc::Indicator7Seg

8.127 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/MDU.cpp File Reference

```
#include "../Atlantis.h"
#include "MDU.h"
#include "../dps/IDP.h"
```

Include dependency graph for MDU.cpp:



Namespaces

- namespace vc

Functions

- void vc::BitmapLocation (char ch, int &x, int &y)

Variables

- GDIParams g_Param

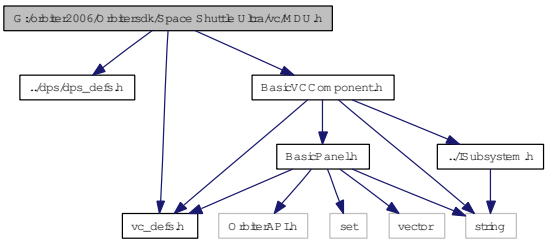
8.127.1 Variable Documentation

8.127.1.1 GDIParams g_Param

8.128 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/MDU.h File Reference

```
#include "../dps/dps_defs.h"
#include "vc_defs.h"
#include "BasicVCComponent.h"
```

Include dependency graph for MDU.h:



This graph shows which files directly or indirectly include this file:

Namespaces

- namespace vc

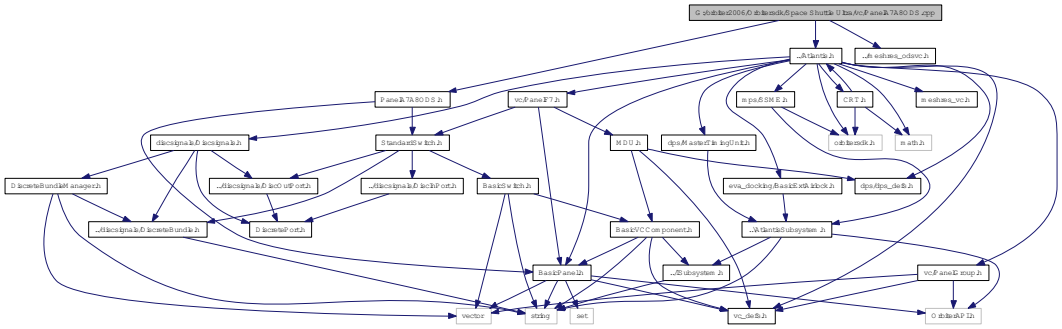
Classes

- class vc::MDU

8.129 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelA7A8ODS.cpp File Reference

```
#include "PanelA7A8ODS.h"
#include "../Atlantis.h"
#include "../meshres_odsvc.h"
```

Include dependency graph for PanelA7A8ODS.cpp:



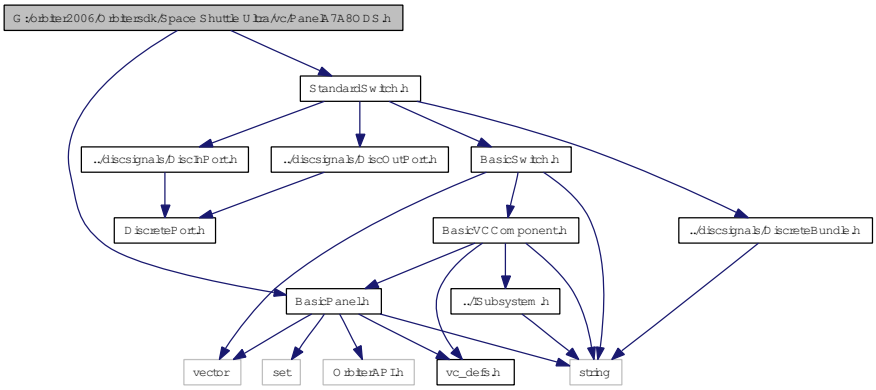
Namespaces

- namespace vc

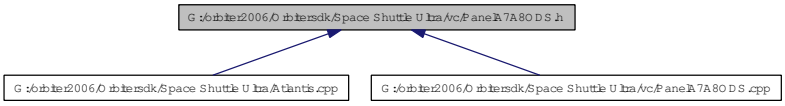
8.130 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelA7A8ODS.h File Reference

```
#include "BasicPanel.h"
#include "StandardSwitch.h"
```

Include dependency graph for PanelA7A8ODS.h:



This graph shows which files directly or indirectly include this file:



Namespaces

- namespace vc

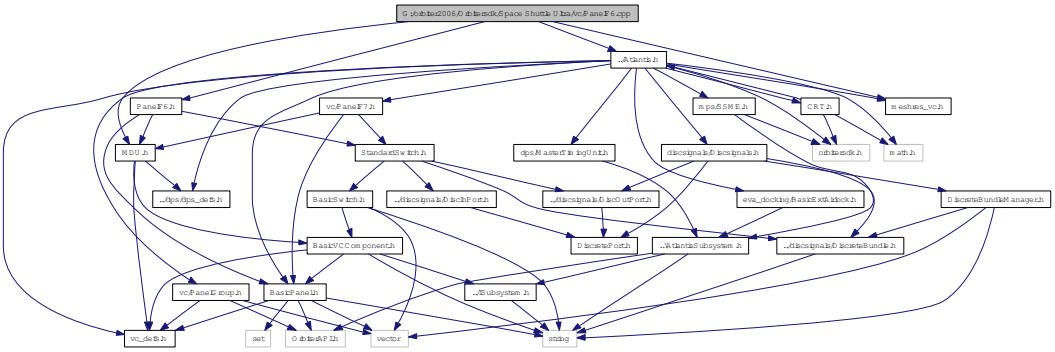
Classes

- class vc::PanelA7A8ODS

8.131 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelF6.cpp File Reference

```
#include "PanelF6.h"
#include "../Atlantis.h"
#include "../meshres_vc.h"
#include "MDU.h"
```

Include dependency graph for PanelF6.cpp:



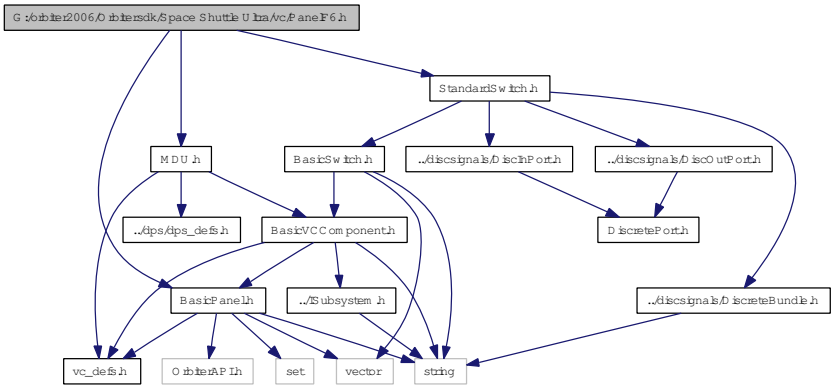
Namespaces

- namespace vc

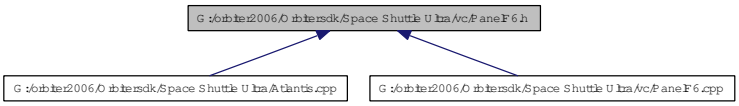
8.132 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelF6.h File Reference

```
#include "BasicPanel.h"
#include "MDU.h"
#include "StandardSwitch.h"
```

Include dependency graph for PanelF6.h:



This graph shows which files directly or indirectly include this file:



Namespaces

- namespace vc

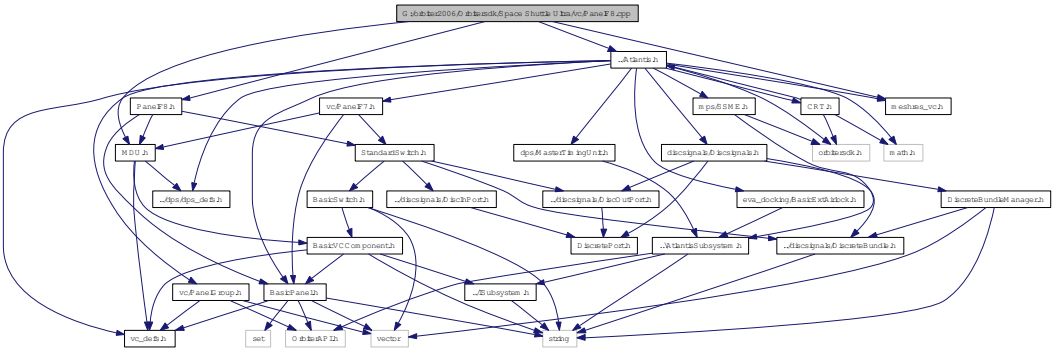
Classes

- class vc::PanelF6

8.133 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelF8.cpp File Reference

```
#include "PanelF8.h"
#include "../Atlantis.h"
#include "../meshres_vc.h"
#include "MDU.h"
```

Include dependency graph for PanelF8.cpp:



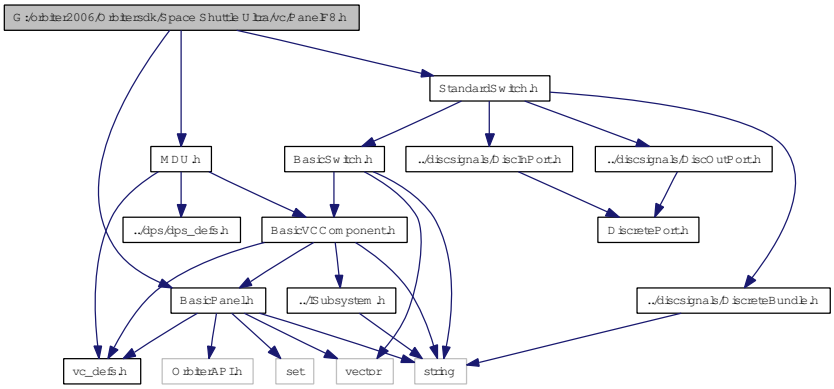
Namespaces

- namespace vc

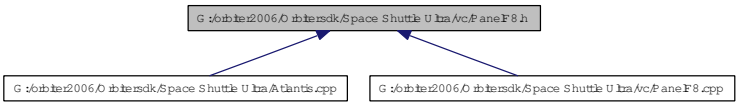
8.134 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelF8.h File Reference

```
#include "BasicPanel.h"
#include "MDU.h"
#include "StandardSwitch.h"
```

Include dependency graph for PanelF8.h:



This graph shows which files directly or indirectly include this file:



Namespaces

- namespace vc

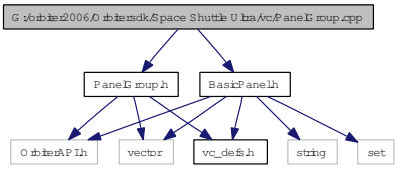
Classes

- class vc::PanelF8

8.135 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelGroup.cpp File Reference

```
#include "PanelGroup.h"
#include "BasicPanel.h"
```

Include dependency graph for PanelGroup.cpp:



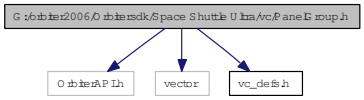
Namespaces

- namespace vc

8.136 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelGroup.h File Reference

```
#include "OrbiterAPI.h"
#include <vector>
#include "vc_defs.h"
```

Include dependency graph for PanelGroup.h:



This graph shows which files directly or indirectly include this file:

Namespaces

- namespace vc

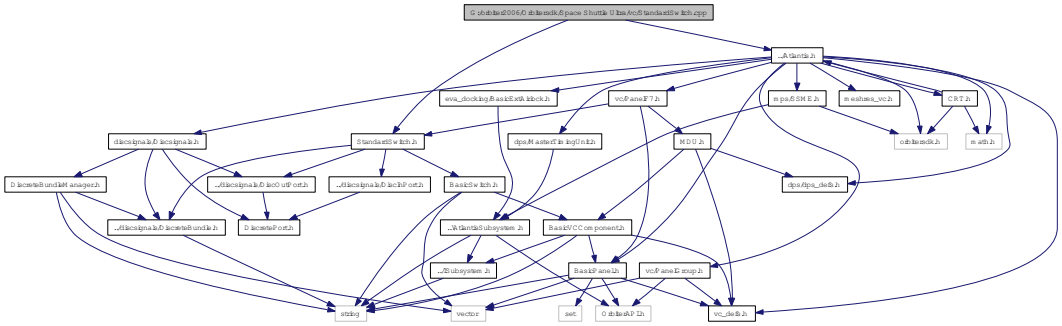
Classes

- class vc::PanelGroup

8.137 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/StandardSwitch.cpp File Reference

```
#include "StandardSwitch.h"
#include "../Atlantis.h"
```

Include dependency graph for StandardSwitch.cpp:



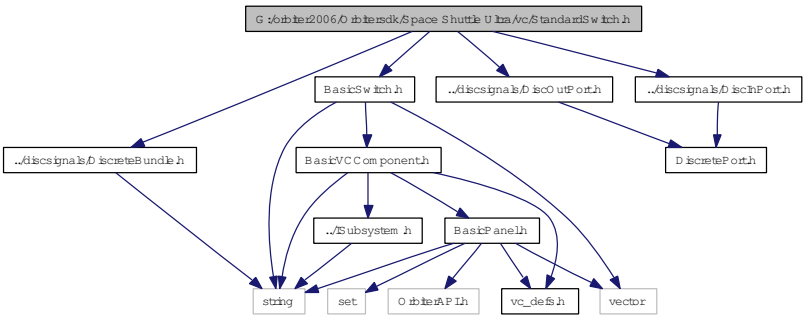
Namespaces

- namespace vc

8.138 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/StandardSwitch.h File Reference

```
#include "../discsignals/DiscreteBundle.h"
#include "../discsignals/DiscOutPort.h"
#include "../discsignals/DiscInPort.h"
#include "BasicSwitch.h"
```

Include dependency graph for StandardSwitch.h:



This graph shows which files directly or indirectly include this file:

Namespaces

- namespace vc

Classes

- class vc::LockableLever
- class vc::StandardSwitch
- class vc::StdSwitch2
- class vc::StdSwitch3
- class vc::RussianCB

Defines

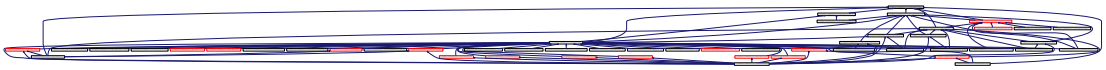
- #define AFX_STANDARDSWITCH_H__A87D233D_E5CF_4343_90D8_BF65F16C4DB4__INCLUDED_

8.138.1 Define Documentation

8.138.1.1 #define AFX_STANDARDSWITCH_H__A87D233D_E5CF_4343_90D8_-BF65F16C4DB4__INCLUDED_

8.139 G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/vc_defs.h File Reference

This graph shows which files directly or indirectly include this file:



Namespaces

- namespace vc

Classes

- struct vc::VCMOUSEEVENTPOINT

Defines

- #define ___SPACE_SHUTTLE_ULTRA_VC_DEFINITIONS_H_INCLUDED___

Typedefs

- typedef MDU * vc::PMDU

Variables

- const int vc::MDUID_CDR1 = 0
- const int vc::MDUID_CDR2 = 1
- const int vc::MDUID_PLT1 = 2
- const int vc::MDUID_PLT2 = 3
- const int vc::MDUID_CRT1 = 4
- const int vc::MDUID_CRT2 = 5
- const int vc::MDUID_CRT3 = 6
- const int vc::MDUID_CRT4 = 7
- const int vc::MDUID_MFD1 = 8
- const int vc::MDUID_MFD2 = 9
- const int vc::MDUID_AFD = 10

8.139.1 Define Documentation

8.139.1.1 #define ___SPACE_SHUTTLE_ULTRA_VC_DEFINITIONS_H_INCLUDED___

Index

~AP101S
dps::AP101S, 36
~ATVC
mps::ATVC, 83
~AccAssembly
gnc::AccAssembly, 30
~AirDataProbeSystem
AirDataProbeSystem, 32
~Atlantis
Atlantis, 53
~AtlantisAPUHYD
AtlantisAPUHYD, 72
~AtlantisSubsystem
AtlantisSubsystem, 78
~BIU
dps::BIU, 111
~BLOCK_II
mps::BLOCK_II, 113
~BasicExternalAirlock
eva_docking::BasicExternalAirlock, 86
~BasicLight
vc::BasicLight, 90
~BasicPanel
vc::BasicPanel, 92
~BasicSwitch
vc::BasicSwitch, 98
~BasicVCComponent
vc::BasicVCComponent, 104
~BasicValve
BasicValve, 101
~CRT
CRT, 123
~DiscInPort
discsignals::DiscInPort, 137
~DiscOutPort
discsignals::DiscOutPort, 138
~DiscreteBundle
discsignals::DiscreteBundle, 140
~DiscreteBundleManager
discsignals::DiscreteBundleManager, 143
~DiscretePort
discsignals::DiscretePort, 144
~EIU
mps::EIU, 147
~EventTimer
vc::EventTimer, 151
~ExtAirlock
eva_docking::ExtAirlock, 153
~HydActuator
hydraulics::HydActuator, 162
~IDP
dps::IDP, 166
~IDPSoftware
dps::IDPSoftware, 171
~Indicator7Seg
vc::Indicator7Seg, 173
~KuBandSystem
KuBandSystem, 181
~LockableLever
vc::LockableLever, 183
~MDM
dps::MDM, 191
~MDU
vc::MDU, 194
~MLP
MLP, 206
~MPS
mps::MPS, 209
~MasterTimingUnit
dps::MasterTimingUnit, 187
~MechActuator
MechActuator, 200
~Mission
mission::Mission, 204
~ODS
eva_docking::ODS, 214
~OMSSubsystem
OMSSubsystem, 221
~PanelA7A8ODS
vc::PanelA7A8ODS, 227
~PanelC2
PanelC2, 235
~PanelF6
vc::PanelF6, 241
~PanelF7
vc::PanelF7, 244
~PanelF8
vc::PanelF8, 248
~PanelGroup
vc::PanelGroup, 251

~RAM16
dps::RAM16, 270
~RussianCB
vc::RussianCB, 273
~SSME
mps::SSME, 277
~ShuttleBus
dps::ShuttleBus, 274
~StandardSwitch
vc::StandardSwitch, 288
~StdSwitch2
vc::StdSwitch2, 291
~StdSwitch3
vc::StdSwitch3, 293
~Stopwatch
Stopwatch, 294
~SubsystemDirector
SubsystemDirector, 295
~ValveTypeBool
ValveTypeBool, 297
_DCU
mps, 22
_ENGINE_STATUS
mps, 22
_MPS_VALVE
mps, 23
___SPACE_SHUTTLE_ULTRA_VC_-DEFINITIONS_H_INCLUDED___
vc_defs.h, 524
___iCurrentManifold
Atlantis, 57
_backdoor
BasicValve, 101
ValveTypeBool, 298

A
Atlantis, 66
a
Atlantis, 55
a0
Atlantis, 66
AC_resource.h
IDC_RADIO1, 301
IDC_RADIO2, 301
IDC_RADIO3, 301
IDC_RADIO4, 301
IDD_ACONFIG, 301
AccAssembly
gnc::AccAssembly, 29
ACPHASE
eps, 16
ActionString
Atlantis.cpp, 308
GearOp.cpp, 398
PanelA4.cpp, 467
PanelA8.cpp, 469
PanelC2.cpp, 472
PanelC3.cpp, 475
PanelF7.cpp, 478
PanelO3.cpp, 482
PanelR2.cpp, 485
PlBayOp.cpp, 488
ActivateSSS
MLP, 206
activeDCU
mps::SSME, 283
Add
vc::BasicPanel, 94
AddAIDToMouseEventList
vc::BasicPanel, 93
AddAIDToRedrawEventList
vc::BasicVCComponent, 105
AddBundle
discsignals::DiscreteBundleManager, 143
AddKUBandVisual
Atlantis, 57
AddMeshes
AtlantisSubsystem, 79
eva_docking::BasicExternalAirlock, 86
eva_docking::ExtAirlock, 153
eva_docking::ODS, 214
vc::BasicPanel, 94
vc::PanelA7A8ODS, 227
AddOrbiterVisual
Atlantis, 53
AddPanel
vc::PanelGroup, 251
AddPrimaryRCSExhaust
Atlantis, 53
AddSRBVisual
Atlantis, 53
AddSubsystem
SubsystemDirector, 295
AddTankVisual
Atlantis, 53
ADPS_DEPLOY
AirDataProbeSystem, 32
ADPS_DEPLOYED
AirDataProbeSystem, 32
ADPS_DEPLOYING
AirDataProbeSystem, 32
ADPS_STOW
AirDataProbeSystem, 32
ADPS_STOW_ENABLE
AirDataProbeSystem, 32
ADPS_STOW_INHIBIT
AirDataProbeSystem, 32
ADPS_STOWED

AirDataProbeSystem, 32
ADPS_STOWING
AirDataProbeSystem, 32
AftPitchThrusters
Atlantis.h, 325
AftRollThrusters
Atlantis.h, 325
AftRotThrusters
Atlantis.h, 325
AftYawThrusters
Atlantis.h, 325
AFX_AIRDATAPROBESYSTEM_H__-74A5598A_2DB5_47F6_AC02_-B106E9C86F81__INCLUDED_
AirDataProbeSystem.h, 303
AFX_AP101S_H__01BE9789_9E06_4F6B_-9FAD_F48EA99C5161__INCLUDED_
AP101S.h, 367
AFX_ATLANTISAPUHYD_H__76669820_-C41F_4F68_8755_0C199CFA6019__-INCLUDED_
AtlantisAPUHYD.h, 336
AFX_ATLANTISSUBSYSTEM_H__6A9F0F48_-D391_4E11_9536_F359182CA558__-INCLUDED_
AtlantisSubsystem.h, 340
AFX_BASICSWITCH_H__3790BCEE_A9B9_-492A_A7F4_B0DA5E454AFB__-INCLUDED_
BasicSwitch.h, 505
AFX_BASICVCCOMPONENT_H__B74A68D7_-2919_4C3F_AB0C_4B4FBA3AA2B7_-_INCLUDED_
BasicVCComponent.h, 507
AFX_DISCOUTPORT_H__B01F0E2B_BD2E_-4A08_B236_7DCF6C1507CD__-INCLUDED_
DiscOutPort.h, 358
AFX_DISCRETEBUNDLE_H__A928D10B_-004B_495D_87CD_3A0309B3EFDA__-INCLUDED_
DiscreteBundle.h, 360
AFX_DISCRETEPORT_H__18D00C82_9652_-493D_8A1D_D1CFF223A0B8__-INCLUDED_
DiscretePort.h, 364
AFX_ISUBSYSTEM_H__A4896D4F_46E0_-4D68_AEA4_F6B9CF259711__-INCLUDED_
ISubsystem.h, 407
AFX_MASTERTIMINGUNIT_H__FEF22983_-033D_412B_B3A9_876EE3A3EC04__-INCLUDED_
MasterTimingUnit.h, 376
AFX_MDM_H__BC1B13B9_53D9_4DC9_-A6D6_2E7EA9F155A6__INCLUDED_
MDM.h, 378
AFX_SHUTTLEBUS_H__5E15D9D6_08AF_-49BD_96FB_2835793F608E__-INCLUDED_
ShuttleBus.h, 381
AFX_STANDARDSWITCH_H__A87D233D_-E5CF_4343_90D8_BF65F16C4DB4__-INCLUDED_
StandardSwitch.h, 523
AFX_STDAFX_H__C669E737_B06A_482C_-B99B_41C360D81979__INCLUDED_
StdAfx.h, 405
ahHDP
Atlantis, 60
MLP, 207
AID_A4
Atlantis.h, 322
AID_A4_EVTTMR
Atlantis.h, 322
AID_A4_MAX
Atlantis.h, 322
AID_A4_METTMR1
Atlantis.h, 322
AID_A4_METTMR2
Atlantis.h, 322
AID_A4_MIN
Atlantis.h, 322
AID_A7A3
Atlantis.h, 325
AID_A8
Atlantis.h, 322
AID_A8_MAX
Atlantis.h, 322
AID_A8_MIN
Atlantis.h, 322
AID_A8_TKBK1
Atlantis.h, 322
AID_A8_TKBK10
Atlantis.h, 322
AID_A8_TKBK11
Atlantis.h, 322
AID_A8_TKBK12
Atlantis.h, 322
AID_A8_TKBK13
Atlantis.h, 322
AID_A8_TKBK14
Atlantis.h, 322
AID_A8_TKBK15
Atlantis.h, 322
AID_A8_TKBK16
Atlantis.h, 322
AID_A8_TKBK17

Atlantis.h, 322
AID_A8_TKBK2
Atlantis.h, 322
AID_A8_TKBK3
Atlantis.h, 322
AID_A8_TKBK4
Atlantis.h, 322
AID_A8_TKBK5
Atlantis.h, 322
AID_A8_TKBK6
Atlantis.h, 322
AID_A8_TKBK7
Atlantis.h, 322
AID_A8_TKBK8
Atlantis.h, 322
AID_A8_TKBK9
Atlantis.h, 322
AID_A8A3
Atlantis.h, 325
AID_AFD_BRT
Atlantis.h, 322
AID_AFD_BUTTONS
Atlantis.h, 322
AID_AFD_PWR
Atlantis.h, 322
AID_C2
Atlantis.h, 322
AID_C2_MAX
Atlantis.h, 322
AID_C2_MIN
Atlantis.h, 322
AID_C2_WND0
Atlantis.h, 322
AID_C2_WND1
Atlantis.h, 322
AID_C2_WND2
Atlantis.h, 322
AID_C2_WND3
Atlantis.h, 322
AID_C3
Atlantis.h, 322
AID_C3_MAX
Atlantis.h, 322
AID_C3_MIN
Atlantis.h, 322
AID_CDR1_BRT
Atlantis.h, 322
AID_CDR1_BUTTONS
Atlantis.h, 322
AID_CDR1_PWR
Atlantis.h, 322
AID_CDR2_BRT
Atlantis.h, 322
AID_CDR2_BUTTONS
Atlantis.h, 322
AID_CDR2_PWR
Atlantis.h, 322
AID_CRT1_BRT
Atlantis.h, 322
AID_CRT1_BUTTONS
Atlantis.h, 322
AID_CRT1_PWR
Atlantis.h, 322
AID_CRT2_BRT
Atlantis.h, 322
AID_CRT2_BUTTONS
Atlantis.h, 322
AID_CRT2_PWR
Atlantis.h, 322
AID_CRT3_BRT
Atlantis.h, 322
AID_CRT3_BUTTONS
Atlantis.h, 322
AID_CRT3_PWR
Atlantis.h, 322
AID_CRT4_BRT
Atlantis.h, 322
AID_CRT4_BUTTONS
Atlantis.h, 322
AID_CRT4_PWR
Atlantis.h, 322
AID_CUSTOM_PANELS_MAX
Atlantis.h, 325
AID_CUSTOM_PANELS_MIN
Atlantis.h, 325
AID_F6
Atlantis.h, 322
AID_F6_MAX
Atlantis.h, 322
AID_F6_MIN
Atlantis.h, 322
AID_F6_TKBK1
Atlantis.h, 322
AID_F6_TKBK2
Atlantis.h, 322
AID_F6_TKBK3
Atlantis.h, 322
AID_F7
Atlantis.h, 322
AID_F7_EVTTMR1
Atlantis.h, 322
AID_F7_EVTTMR2
Atlantis.h, 322
AID_F7_EVTTMR3
Atlantis.h, 322
AID_F7_EVTTMR4
Atlantis.h, 322
AID_F7_MAX

Atlantis.h, 322
AID_F7_MIN
Atlantis.h, 322
AID_F8
Atlantis.h, 322
AID_F8_MAX
Atlantis.h, 322
AID_F8_MIN
Atlantis.h, 322
AID_KYBD_CDR
Atlantis.h, 322
AID_KYBD_PLT
Atlantis.h, 322
AID_MDU_AFD1
Atlantis.h, 322
AID_MDU_CDR1
Atlantis.h, 322
AID_MDU_CDR2
Atlantis.h, 322
AID_MDU_CRT1
Atlantis.h, 322
AID_MDU_CRT2
Atlantis.h, 322
AID_MDU_CRT3
Atlantis.h, 322
AID_MDU_CRT4
Atlantis.h, 322
AID_MDU_MFD1
Atlantis.h, 322
AID_MDU_MFD2
Atlantis.h, 322
AID_MDU_PLT1
Atlantis.h, 322
AID_MDU_PLT2
Atlantis.h, 322
AID_MFD1_BRT
Atlantis.h, 322
AID_MFD1_BUTTONS
Atlantis.h, 322
AID_MFD1_PWR
Atlantis.h, 322
AID_MFD2_BRT
Atlantis.h, 322
AID_MFD2_BUTTONS
Atlantis.h, 322
AID_MFD2_PWR
Atlantis.h, 322
AID_O3
Atlantis.h, 322
AID_O3_MAX
Atlantis.h, 322
AID_O3_METTMR1
Atlantis.h, 322
AID_O3_METTMR2
Atlantis.h, 322
AID_O3_MIN
Atlantis.h, 322
AID_O3_RCS
Atlantis.h, 322
AID_PLT1_BRT
Atlantis.h, 322
AID_PLT1_BUTTONS
Atlantis.h, 322
AID_PLT1_PWR
Atlantis.h, 322
AID_PLT2_BRT
Atlantis.h, 322
AID_PLT2_BUTTONS
Atlantis.h, 322
AID_PLT2_PWR
Atlantis.h, 322
AID_R13L
Atlantis.h, 322
AID_R13L_MAX
Atlantis.h, 322
AID_R13L_MIN
Atlantis.h, 322
AID_R13L_TKBK1
Atlantis.h, 322
AID_R13L_TKBK2
Atlantis.h, 322
AID_R13L_TKBK3
Atlantis.h, 322
AID_R13L_TKBK4
Atlantis.h, 322
AID_R13L_TKBK5
Atlantis.h, 322
AID_R13L_TKBK6
Atlantis.h, 322
AID_R2
Atlantis.h, 322
AID_R2_MAX
Atlantis.h, 322
AID_R2_MIN
Atlantis.h, 322
AID_R2_TKBK1
Atlantis.h, 322
AID_R2_TKBK2
Atlantis.h, 322
AID_R2_TKBK3
Atlantis.h, 322
AID_R2_TKBK4
Atlantis.h, 322
AID_R2_TKBK5
Atlantis.h, 322
AID_R2_TKBK6
Atlantis.h, 322
AID_R2_TKBK7

Atlantis.h, 322
AID_R2_TKBK8
Atlantis.h, 322
Air_Data_Probe
PanelC3, 239
AIR_DATA_PROBE_SPEED
Atlantis.h, 325
Air_Data_Stow
PanelC3, 239
AirDataProbe
PanelC3, 239
AirDataProbeControl
PanelC3, 239
AirDataProbeSystem, 31
~AirDataProbeSystem, 32
ADPS_DEPLOY, 32
ADPS_DEPLOYED, 32
ADPS_DEPLOYING, 32
ADPS_STOW, 32
ADPS_STOW_ENABLE, 32
ADPS_STOW_INHIBIT, 32
ADPS_STOWED, 32
ADPS_STOWING, 32
AirDataProbeSystem, 32
DEPLOY_MODE, 32
DEPLOY_STATE, 32
GetDynamicPressure, 33
GetStaticPressure, 33
GetTotalTemperature, 33
IsDeployed, 33
left_deploy, 34
left_mode, 34
left_state, 34
left_stow_state, 34
OnPostStep, 33
OnPreStep, 33
OnPropagate, 32
right_deploy, 34
right_mode, 34
right_state, 34
right_stow_state, 34
SetDeployMode, 33
STOW_ENABLE_STATE, 32
AirDataProbeSystem.h
AFX_AIRDATAPROBESYSTEM_H_-
_74A5598A_2DB5_47F6_AC02_-
B106E9C86F81__INCLUDED_, 303
alpha_angle
KuBandSystem, 181
ALT_DELAY
DAPConfig, 134
ALT_JET_OPT
DAPConfig, 134
ALT_JETS
DAPConfig, 134
ALT_ON_TIME
DAPConfig, 134
ALT_RATE_DB
DAPConfig, 134
angle
Atlantis.cpp, 306
AngularVelocity
Atlantis, 66
CRT, 130
anim_adpl
Atlantis, 62
anim_adpr
Atlantis, 62
anim_arm_ee
Atlantis, 66
anim_arm_ep
Atlantis, 66
anim_arm_sp
Atlantis, 66
anim_arm_sy
Atlantis, 66
anim_arm_wp
Atlantis, 66
anim_arm_wr
Atlantis, 66
anim_arm_wy
Atlantis, 66
anim_bf
Atlantis, 62
anim_camBLpitch
Atlantis, 62
anim_camBLyaw
Atlantis, 62
anim_camBRpitch
Atlantis, 62
anim_camBRyaw
Atlantis, 62
anim_camFLpitch
Atlantis, 62
anim_camFLyaw
Atlantis, 62
anim_camFRpitch
Atlantis, 62
anim_camFRyaw
Atlantis, 62
anim_clatch
Atlantis, 62
anim_door
Atlantis, 62
anim_dummy
Atlantis, 62
anim_elev
Atlantis, 62

anim_gear
GearOp, 160
anim_GVA
SSU_FSS_RSS, 285
anim_kualpha
Atlantis, 62
anim_kubd
Atlantis, 62
anim_kubeta
Atlantis, 62
anim_laileron
Atlantis, 62
anim_letumbdoor
Atlantis, 62
anim_OAA
SSU_FSS_RSS, 285
anim_portTS
Atlantis, 62
anim_rad
Atlantis, 62
anim_raileron
Atlantis, 62
anim_retumbdoor
Atlantis, 62
anim_rollout
Atlantis, 66
anim_RSS
SSU_FSS_RSS, 285
anim_rudder
Atlantis, 62
anim_spdb
Atlantis, 62
anim_ssmeLpitch
Atlantis, 62
anim_ssmeLyaw
Atlantis, 62
anim_ssmeRpitch
Atlantis, 62
anim_ssmeRyaw
Atlantis, 62
anim_ssmeTpitch
Atlantis, 62
anim_ssmeTyaw
Atlantis, 62
anim_switch
vc::LockableLever, 183
vc::StandardSwitch, 289
anim_VC_A4
PanelA4, 224
anim_VC_A8
PanelA8, 233
anim_VC_C2
PanelC2, 236
anim_VC_C2Wheel
PanelC2, 236
anim_VC_C3
PanelC3, 239
anim_VC_F6
GearOp, 160
anim_VC_O3
PanelO3, 254
anim_VC_R13L
PayloadBayOp, 269
anim_VC_R2
PanelR2, 260
aoa1
Atlantis.cpp, 308
AP101S
dps::AP101S, 36
AP101S.h
AFX_AP101S_H__01BE9789_9E06_4F6B_-9FAD_F48EA99C5161__INCLUDED_, 367
Apply
AtlantisConfig, 75
APU_1
AtlantisAPUHYD.h, 336
APU_2
AtlantisAPUHYD.h, 336
APU_3
AtlantisAPUHYD.h, 336
APU_AUTO_SHUT_DOWN
PanelR2, 260
APU_CNTLR_PWR
PanelR2, 260
APU_CONTROL
PanelR2, 260
APU_FUEL
PanelR2, 260
APU_Fuel_Last
CRT, 130
APU_Fuel_Rate
CRT, 130
APU_FUEL_TANK_FLOWRATE
Atlantis.h, 325
APU_FUEL_TANK_MASS
Atlantis.h, 325
APU_FUEL_TK_VLV
PanelR2, 260
APU_INDEX
AtlantisAPUHYD.h, 336
APU_READY
PanelR2, 260
APU_Speed
PanelR2, 260
APU_SPEED_SELECT
PanelR2, 260
APU_STATE

PanelR2, 260
apu_tank
Atlantis, 66
APUHYD
CRT, 126
ARM_DEPLOY_SPEED
Atlantis.h, 325
arm_ee_dir
Atlantis, 66
arm_ee_pos
Atlantis, 66
arm_ee_rot
Atlantis, 66
arm_ep
Atlantis, 66
ARM_EXTEND_SPEED
Atlantis.h, 325
ARM_GRAPPLE_SPEED
Atlantis.h, 325
arm_moved
Atlantis, 60
ARM_OPERATING_SPEED
Atlantis.h, 325
ARM_RIGID_SPEED
Atlantis.h, 325
arm_sp
Atlantis, 66
arm_sy
Atlantis, 66
arm_tgt_dir
Atlantis, 66
arm_tgt_pos
Atlantis, 66
arm_tip
Atlantis, 60
ARM_TRANSLATE_SPEED
Atlantis.h, 325
arm_wp
Atlantis, 66
arm_wr
Atlantis, 66
ARM_WRIST_CAM_OFFSET
Atlantis.h, 325
arm_wrist_pos
Atlantis, 66
arm_wy
Atlantis, 66
ArmCradled
Atlantis, 53
ArmGear
GearOp, 159
Atlantis, 37
~Atlantis, 53
___iCurrentManifold, 57
A, 66
a, 55
a0, 66
AddKUBandVisual, 57
AddOrbiterVisual, 53
AddPrimaryRCSExhaust, 53
AddSRBVisual, 53
AddTankVisual, 53
ahHDP, 60
AngularVelocity, 66
anim_adpl, 62
anim_adpr, 62
anim_arm_ee, 66
anim_arm_ep, 66
anim_arm_sp, 66
anim_arm_sy, 66
anim_arm_wp, 66
anim_arm_wr, 66
anim_arm_wy, 66
anim_bf, 62
anim_camBLpitch, 62
anim_camBLyaw, 62
anim_camBRpitch, 62
anim_camBRyaw, 62
anim_camFLpitch, 62
anim_camFLyaw, 62
anim_camFRpitch, 62
anim_camFRyaw, 62
anim_clatch, 62
anim_door, 62
anim_dummy, 62
anim_elev, 62
anim_kualpha, 62
anim_kubd, 62
anim_kubeta, 62
anim_laileron, 62
anim_letumbdoor, 62
anim_portTS, 62
anim_rad, 62
anim_raileron, 62
anim_retumbdoor, 62
anim_rollout, 66
anim_rudder, 62
anim_spdb, 62
anim_ssmeLpitch, 62
anim_ssmeLyaw, 62
anim_ssmeRpitch, 62
anim_ssmeRyaw, 62
anim_ssmeTpitch, 62
anim_ssmeTyaw, 62
apu_tank, 66
arm_ee_dir, 66
arm_ee_pos, 66
arm_ee_rot, 66

arm_ep, 66
arm_moved, 60
arm_sp, 66
arm_sy, 66
arm_tgt_dir, 66
arm_tgt_pos, 66
arm_tip, 60
arm_wp, 66
arm_wr, 66
arm_wrist_pos, 66
arm_wy, 66
ArmCradled, 53
Atlantis, 53
AttControl, 55
AttDeadband, 66
AutoGrappleSequence, 53
AutoMainGimbal, 55
AutoReleaseSequence, 53
B, 66
b0, 55
bAutopilot, 66
bCommMode, 62
bEngineFail, 66
bGrappleInProgress, 66
bHasKUBand, 60
bHasODS, 60
bLiftOff, 60
bManualSeparate, 58
bMECO, 66
bMidDeckVisible, 60
bn, 55
BODY_VECT, 66
bReleaseInProgress, 66
bSRBCutoffFlag, 60
bSSMEGOXVent, 66
bThrottle, 66
BundleManager, 53
bundleManager, 66
BurnAtt, 66
BurnCompleted, 66
BurnInProg, 66
BurnTime, 66
bUseRealRCS, 62
bZThrust, 66
C, 66
c0, 56
C1, 66
C2, 66
c3po, 60
CalcAnimationFKArm, 53
CalcLVLHAttitude, 55
CalcManeuverTargets, 55
CalcNetThrust, 55
CalcPitchYawRollAngles, 55
CalcRelLVLHAttitude, 55
CalcRequiredRates, 55
CalcThrustAngles, 55
CalculateAzimuth, 55
camBLpitch, 62
camBLyaw, 62
camBRpitch, 62
camBRyaw, 62
CameraBLPitch, 62
CameraBLYaw, 62
CameraBRPitch, 62
CameraBRYaw, 62
cameraControl, 62
CameraFLPitch, 62
CameraFLYaw, 62
CameraFRPitch, 62
CameraFRYaw, 62
cameraLowRate, 62
cameraMoved, 62
camFLpitch, 62
camFLyaw, 62
camFRpitch, 62
camFRyaw, 62
CanArrest, 55
cargo_static_mesh_name, 60
cargo_static_ofs, 60
CDRKeyboard, 60
center_arm, 60
center_arm_t, 60
clbkAnimate, 53
clbkConsumeBufferedKey, 53
clbkDrawHUD, 53
clbkFocusChanged, 53
clbkLoadGenericCockpit, 53
clbkLoadStateEx, 53
clbkLoadVC, 53
clbkMFDMode, 53
clbkPlaybackEvent, 53
clbkPostStep, 53
clbkPreStep, 53
clbkSaveState, 53
clbkSetClassCaps, 53
clbkSetStateEx, 53
clbkVCMouseEvent, 53
clbkVCRedrawEvent, 53
clbkVisualCreated, 53
clbkVisualDestroyed, 53
ClearMeshes, 53
CmdPDot, 66
CmdRDot, 66
CmdYDot, 66
cn, 56
ControlSurfacesEnabled, 66
ConvertAnglesBetweenM50AndOrbiter, 55

ConvertLVLHAnglesToM50, 55
ConvertOrbiterAnglesToLocal, 55
ConvertVectorBetweenOrbiterAndM50, 55
CreateAttControls_RCS, 53
CreateFRCS, 53
CreateLeftARCS, 53
CreateMPSGOXVents, 53
CreateOrbiterTanks, 53
CreateRightARCS, 53
CRT, 57, 130
CRT_SEL, 66
CurrentAttitude, 66
d1, 66
d2, 66
d3, 66
d4, 66
DAP, 66
DAPMode, 66
DataInput, 66
DefineAnimations, 53
DefineKUBandAnimations, 53
DefineSSMEExhaust, 53
deltah, 66
deltatheta, 66
DeltaV, 66
deltav, 66
DeltaVTot, 66
DeltaVTotms, 66
DisableAllRCS, 53
DisableControlSurfaces, 53
DisableThrusters, 55
Display, 66
DisplayCameraLabel, 57
DisplayJointAngles, 66
do_cargostatic, 60
do_eva, 60
do_plat, 60
dV, 66
eCurrent, 66
edit, 66
EEGrappleMode, 66
el, 66
elbow_max, 66
elbow_min, 66
elbow_neutral, 66
elbow_range, 66
EnableAllRCS, 53
EnableControlSurfaces, 53
EnableThrusters, 55
EngineFail, 66
EngineFailTime, 66
EngineNullPosition, 66
ep_angle, 66
ep_null, 66
Eq, 55
Estimate, 55
ex_main, 64
ex_retro, 66
Extend, 66
F, 66
FailEngine, 55
fdotdottheta, 66
fdotr, 66
fdottheta, 66
fh, 66
fhdotrh, 66
FireAllNextManifold, 53
fr, 66
ftheta, 66
fTimeCameraLabel, 66
g, 66
GearOp, 57, 160
GetETPropellant, 53
GetGPCLVLHVel, 53
GetGPCMET, 53
GetGPCRefHDot, 53
GetIDP, 53
GetLastCreatedMFD, 53
GetLiftOffFlag, 53
GetMDU, 53
GetOMSPressure, 53
GetOrbiterCoGOffset, 53
GetPYR, 55
GetPYR2, 55
GetRotMatrixX, 55
GetRotMatrixY, 55
GetRotMatrixZ, 55
GetSRBChamberPressure, 53
GimbalOMS, 55
gop, 60
GPC, 55
Grapple, 66
Guide, 55
GVesselPos, 66
GVesselVel, 66
h, 66
hAC_arm, 66
hAC_sat, 66
hAC_satref, 66
HasExternalAirlock, 53
hbodyflap, 66
helevator, 66
hh, 66
HideMidDeck, 53
HideODS, 53
hKUBandMesh, 60
hlaileron, 66
hMidDeckMesh, 60

hMMU, 66
hODSMesh, 60
hOrbiterCockpitMesh, 60
hOrbiterMesh, 60
hOrbiterRMSMesh, 60
hOrbiterVCMesh, 60
hraileron, 66
hrudder, 66
hSAT, 66
hSRBMesh, 60
HT, 66
hT, 66
hTankMesh, 60
huds, 66
hv, 66
Hydraulic_Press, 66
IgnitionTime, 66
InertialOrientationRad, 66
InitializeAutopilot, 55
Input, 55
Isp, 66
IsValidSPEC, 53
item, 66
JetsEnabled, 66
Jettison, 53
jettison_time, 66
Keyboard, 57, 178
last_mfd, 66
last_target_pitch, 66
last_target_roll, 66
last_target_yaw, 66
Launch_time, 58
LaunchClamps, 53
ldoor_drag, 60
ll, 66
LoadManeuver, 55
lu, 66
LVLH_X, 66
LVLH_Y, 66
LVLH_Z, 66
LVLHError, 66
LVLHOrientationReqd, 66
LVLHRateVector, 66
m, 66
MajorCycle, 55
Maneuver, 55
ManeuverinProg, 66
ManeuverStatus, 66
Mass, 66
MaxThrust, 66
mdus, 66
mesh_cargo_static, 58
mesh_cockpit, 58
mesh_kuband, 58
mesh_middeck, 58
mesh_ods, 58
mesh_orbiter, 58
mesh_rms, 58
mesh_srb, 58
mesh_tank, 58
mesh_vc, 58
MET, 58
met, 58
MET_Add, 58
metCutoff, 66
mfdbright, 66
mfds, 66
MNVR, 66
MNVR_COMPLETE, 51
MNVR_IN_PROGRESS, 51
MNVR_OFF, 51
MNVR_STARTING, 51
MNVR_OPTION, 66
MNVR_TIME, 66
MNVRLOAD, 66
MPM_Microswitches, 66
MPS_He_Reg_Press, 66
MPS_He_Tank_Press, 66
mps_helium_tank, 66
MPS_Manifold_Press, 66
MRL, 66
MRL_AftMicroswitches, 66
MRL_FwdMicroswitches, 66
MRL_MidMicroswitches, 66
mu, 66
MultiplyByMatrix, 55
Navigate, 55
newmfd, 66
nhCameraLabel, 66
Normalize, 55
NormZ, 55
NoseThrusters, 66
NullRates, 66
NullStartAngle, 55
obss_attach, 60
ofs_sts_sat, 60
OM, 66
omega, 66
OMS, 66
OMS_Assist, 66
oms_helium_tank, 66
OMSGimbal, 66
oparam, 66
OperateSpeedbrake, 53
ops, 66
orbiter_ofs, 60
P, 66
p, 66

pA7A8Panel, 58
pADPS, 60
PaintMarkings, 53
PanelA4, 57, 224
panela4, 60
PanelA8, 57, 233
panela8, 60
PanelC2, 57, 236
panelc2, 60
PanelC3, 57, 239
PanelF7, 246
PanelO3, 57, 254
panelo3, 60
PanelR2, 57, 260
PAYCAM_DlgProc, 57
PayloadBayOp, 57, 269
pCommModeHandler, 60
PEG7, 66
pExtAirlock, 60
pgAftMSS, 60
pgAftOOS, 60
pgAftPSS, 60
pgCenter, 60
pgForward, 60
pgLeft, 60
pgOverhead, 60
pgRight, 60
ph_frcs, 62
ph_lrcs, 62
ph_oms, 62
ph_rrcs, 62
ph_srb, 62
ph_tank, 62
phi, 66
pIDP, 60
Pitch, 66
pitch, 66
PitchActive, 60
PitchYawRoll, 66
pl_mass, 66
plop, 60
PLTKeyboard, 60
PMI, 66
pMTU, 60
pOMS, 60
posMoon, 66
Projection, 55
pshSlag1, 60
pshSlag2, 60
pshSlag3, 60
pSSME, 60
psubsystems, 60
pszCameraLabelBuffer, 66
r, 66
r2d2, 60
Radius, 66
range, 55
RateCommand, 55
RateDeadband, 66
rbar, 66
RCSEnabled, 66
rdoor_drag, 60
Re, 66
RegisterMDU, 53
render_cockpit, 66
REQD_ATT, 66
ReqdRates, 66
reset_mmu, 66
reset_sat, 66
RevertSpeedbrake, 53
rh, 66
rh0, 66
Rigidize, 66
rmh, 66
RMS, 66
rms_anim, 66
rms_attach, 60
RMS_DlgProc, 57
rms_rollout_anim, 66
RMSRollout, 66
Roll, 66
roll, 66
RollActive, 60
RollToHeadsUp, 66
ROT, 66
RotateVector, 55
RotateVectorPYR, 55
RotateVectorX, 55
RotateVectorY, 55
RotateVectorZ, 55
RotRate, 66
rv, 66
sat_anim, 66
sat_attach, 60
sat_ref, 66
SatGrappled, 55
SatStowed, 55
SeparateBoosters, 53
SeparateMMU, 53
SeparateTank, 53
SetAirDataProbeDeployment, 53
SetAnimationArm, 53
SetAnimationCameras, 56
SetAnimationIKArm, 53
SetBayDoorLatchPosition, 53
SetBayDoorPosition, 53
SetETUmbDoorPosition, 53
SetILoads, 55

SetKuAntennaPosition, 53
SetKuGimbalAngles, 53
SetLastCreatedMFD, 53
SetLaunchConfiguration, 53
SetOrbiterConfiguration, 53
SetOrbiterTankConfiguration, 53
SetPostLaunchConfiguration, 53
SetRadiatorPosition, 53
SetRadLatchPosition, 53
SetRates, 55
SetSpeedbrake, 53
shoulder_brace, 66
shoulder_max, 66
shoulder_min, 66
shoulder_neutral, 66
shoulder_pos, 66
shoulder_range, 66
ShowMidDeck, 53
ShowODS, 53
SidDay, 66
SignalGSEBreakHDP, 53
SignalGSEStart, 53
slag1, 60
slag2, 60
slag3, 60
SMOps, 66
sp_angle, 66
sp_null, 66
spdb_proc, 58
spdb_status, 57
spdb_tgt, 60
srb_id1, 58
srb_id2, 58
SSMEL_GOX_REF1, 66
SSMER_GOX_REF1, 66
SSMET_GOX_REF1, 66
stage1guidance, 66
stage1guidance_size, 66
START_TIME, 66
Status, 66
status, 57
SteerGimbal, 55
StopAllManifolds, 53
sy_angle, 66
T, 66
t0, 58
TailThrusters, 66
TAp, 66
target_pitch, 66
target_roll, 66
target_yaw, 66
TargetAttM50, 66
TargetAttOrbiter, 66
tau, 66
TEcc, 66
tex_rcs, 62
TGT_ID, 66
TgtAlt, 66
TgtFPA, 66
TgtInc, 66
TgtLAN, 66
TgtRad, 66
TgtSpd, 66
th_att_lin, 62
th_att_rcs, 62
th_main, 62
th_oms, 62
th_srb, 62
th_ssme_gox, 62
th_ssme_loxdump, 62
THeading, 66
theta, 66
thetah, 66
ThetaT, 66
thetaT, 66
thg_main, 66
thg_retro, 66
thg_srb, 66
ThisVessel, 60
thManFRCS1, 63
thManFRCS2, 63
thManFRCS3, 63
thManFRCS4, 63
thManFRCS5, 63
thManLRCS1, 63
thManLRCS2, 63
thManLRCS3, 63
thManLRCS4, 63
thManLRCS5, 63
thManRRCS1, 63
thManRRCS2, 64
thManRRCS3, 64
thManRRCS4, 64
thManRRCS5, 64
ThrAngleP, 66
ThrAngleY, 66
Throttle, 55
Throttle_Bucket, 66
Thrust, 66
Thrusters, 66
TIG, 66
tig, 66
TLastAB, 66
TLastMajorCycle, 66
tMECO, 66
ToDeg, 55
ToggleArrest, 53
ToggleGrapple, 53

ToggleVCMode, 53
ToRad, 55
Torque, 66
TPe, 66
TrackRates, 66
TriggerLiftOff, 57
Trim, 66
TRK, 66
TrkRate, 66
TRKROT_OPTION, 66
tSRBSep, 66
TTrA, 66
TV_ROLL, 66
UpdateDAP, 55
UpdateMesh, 53
UpdateMPMMicroswitches, 55
UpdateMRLMicroswitches, 55
UpdateRMSAngles, 53
UpdateRMSPositions, 53
UpdateSSMEGimbalAnimations, 53
usCurrentPlayerChar, 62
usLastMDUID, 66
v, 66
vc::MDU, 57
vcDeckMode, 58
VCM_AIRLOCK, 51
VCM_FLIGHTDECK, 51
VCM_MIDDECK, 51
VCMode, 57
velMoon, 66
VGO, 66
vgoTot, 66
vh, 66
vis, 60
vr, 66
vtheta, 66
vv, 66
WingName, 57
wp_angle, 66
wp_wy, 66
wr_angle, 66
wrist_max, 66
wrist_min, 66
wrist_neutral, 66
wrist_range, 66
wrist_roll_max, 66
wrist_roll_min, 66
wrist_roll_neutral, 66
wrist_roll_range, 66
wrist_yaw_max, 66
wrist_yaw_min, 66
wrist_yaw_neutral, 66
wrist_yaw_range, 66
WT, 66
wy_angle, 66
wy_ee, 66
Y, 66
Yaw, 66
yaw, 66
YawActive, 60
Atlantis.cpp
ActionString, 308
angle, 306
aoa1, 308
Atlantis_DlgProc, 306
cdBase, 308
clBase, 308
cmBase, 309
cmTrim, 309
ExitModule, 306
FlatPlateCoeff, 307
g_hc, 309
g_Param, 309
GetSRB_State, 307
gpcReadValue, 307
gpcSetValue, 307
GRP_Endeffector, 306
HLiftCoeff, 307
InitModule, 307
linterp, 307
listerp, 307
LOADBMP, 306
mach, 310
n_aoa1, 310
n_mach, 310
n_trimExt, 310
ORBITER_MODULE, 306
OrbiterA, 310
Orbiterb, 310
Orbiterc, 310
OrbiterS, 310
ovcExit, 307
ovcInit, 307
PAYCAM_DlgProc, 307
PitchOff, 310
RMS_DlgProc, 307
RollOff, 310
STRICT, 306
tableterp, 307
tpir, 307
trimExt, 310
VLiftCoeff, 308
YawOff, 310
Atlantis.h
AftPitchThrusters, 325
AftRollThrusters, 325
AftRotThrusters, 325
AftYawThrusters, 325

AID_A4, 322
AID_A4_EVTTMR, 322
AID_A4_MAX, 322
AID_A4_METTMR1, 322
AID_A4_METTMR2, 322
AID_A4_MIN, 322
AID_A7A3, 325
AID_A8, 322
AID_A8_MAX, 322
AID_A8_MIN, 322
AID_A8_TKBK1, 322
AID_A8_TKBK10, 322
AID_A8_TKBK11, 322
AID_A8_TKBK12, 322
AID_A8_TKBK13, 322
AID_A8_TKBK14, 322
AID_A8_TKBK15, 322
AID_A8_TKBK16, 322
AID_A8_TKBK17, 322
AID_A8_TKBK2, 322
AID_A8_TKBK3, 322
AID_A8_TKBK4, 322
AID_A8_TKBK5, 322
AID_A8_TKBK6, 322
AID_A8_TKBK7, 322
AID_A8_TKBK8, 322
AID_A8_TKBK9, 322
AID_A8A3, 325
AID_AFD_BRT, 322
AID_AFD_BUTTONS, 322
AID_AFD_PWR, 322
AID_C2, 322
AID_C2_MAX, 322
AID_C2_MIN, 322
AID_C2_WND0, 322
AID_C2_WND1, 322
AID_C2_WND2, 322
AID_C2_WND3, 322
AID_C3, 322
AID_C3_MAX, 322
AID_C3_MIN, 322
AID_CDR1_BRT, 322
AID_CDR1_BUTTONS, 322
AID_CDR1_PWR, 322
AID_CDR2_BRT, 322
AID_CDR2_BUTTONS, 322
AID_CDR2_PWR, 322
AID_CRT1_BRT, 322
AID_CRT1_BUTTONS, 322
AID_CRT1_PWR, 322
AID_CRT2_BRT, 322
AID_CRT2_BUTTONS, 322
AID_CRT2_PWR, 322
AID_CRT3_BRT, 322
AID_CRT3_BUTTONS, 322
AID_CRT3_PWR, 322
AID_CRT4_BRT, 322
AID_CRT4_BUTTONS, 322
AID_CRT4_PWR, 322
AID_CUSTOM_PANELS_MAX, 325
AID_CUSTOM_PANELS_MIN, 325
AID_F6, 322
AID_F6_MAX, 322
AID_F6_MIN, 322
AID_F6_TKBK1, 322
AID_F6_TKBK2, 322
AID_F6_TKBK3, 322
AID_F7, 322
AID_F7_EVTTMR1, 322
AID_F7_EVTTMR2, 322
AID_F7_EVTTMR3, 322
AID_F7_EVTTMR4, 322
AID_F7_MAX, 322
AID_F7_MIN, 322
AID_F8, 322
AID_F8_MAX, 322
AID_F8_MIN, 322
AID_KYBD_CDR, 322
AID_KYBD_PLT, 322
AID_MDU_AFD1, 322
AID_MDU_CDR1, 322
AID_MDU_CDR2, 322
AID_MDU_CRT1, 322
AID_MDU_CRT2, 322
AID_MDU_CRT3, 322
AID_MDU_CRT4, 322
AID_MDU_MFD1, 322
AID_MDU_MFD2, 322
AID_MDU_PLT1, 322
AID_MDU_PLT2, 322
AID_MFD1_BRT, 322
AID_MFD1_BUTTONS, 322
AID_MFD1_PWR, 322
AID_MFD2_BRT, 322
AID_MFD2_BUTTONS, 322
AID_MFD2_PWR, 322
AID_O3, 322
AID_O3_MAX, 322
AID_O3_METTMR1, 322
AID_O3_METTMR2, 322
AID_O3_MIN, 322
AID_O3_RCS, 322
AID_PLT1_BRT, 322
AID_PLT1_BUTTONS, 322
AID_PLT1_PWR, 322
AID_PLT2_BRT, 322
AID_PLT2_BUTTONS, 322
AID_PLT2_PWR, 322

AID_R13L, 322
AID_R13L_MAX, 322
AID_R13L_MIN, 322
AID_R13L_TKBK1, 322
AID_R13L_TKBK2, 322
AID_R13L_TKBK3, 322
AID_R13L_TKBK4, 322
AID_R13L_TKBK5, 322
AID_R13L_TKBK6, 322
AID_R2, 322
AID_R2_MAX, 322
AID_R2_MIN, 322
AID_R2_TKBK1, 322
AID_R2_TKBK2, 322
AID_R2_TKBK3, 322
AID_R2_TKBK4, 322
AID_R2_TKBK5, 322
AID_R2_TKBK6, 322
AID_R2_TKBK7, 322
AID_R2_TKBK8, 322
AIR_DATA_PROBE_SPEED, 325
APU_FUEL_TANK_FLOWRATE, 325
APU_FUEL_TANK_MASS, 325
ARM_DEPLOY_SPEED, 325
ARM_EXTEND_SPEED, 325
ARM_GRAPPLE_SPEED, 325
ARM_OPERATING_SPEED, 325
ARM_RIGID_SPEED, 325
ARM_TRANSLATE_SPEED, 325
ARM_WRIST_CAM_OFFSET, 325
AXIS, 322
AXIS_TILT, 325
BSM_ISP0, 325
BSM_MAX_PROPELLANT_MASS, 325
BSM_THRUST0, 325
CAM_HIGHRATE_SPEED, 325
CAM_LOWRATE_SPEED, 325
DEFAULT_MESHNAME_COCKPIT, 325
DEFAULT_MESHNAME_ET, 325
DEFAULT_MESHNAME_KU, 325
DEFAULT_MESHNAME_LSRB, 325
DEFAULT_MESHNAME_MIDDECK, 325
DEFAULT_MESHNAME_ODS, 325
DEFAULT_MESHNAME_ORBITER, 325
DEFAULT_MESHNAME_RMS, 325
DEFAULT_MESHNAME_RSRB, 325
DEFAULT_MESHNAME_VC, 325
defaultStage1Guidance, 325
DOOR_OPERATING_SPEED, 326
DOORLATCH_OPERATING_SPEED, 326
DYNP_MAX, 326
ET_SEP_RATE, 326
ET_SEP_TIME, 326
ET_UMB_DOOR_OPERATING_SPEED, 326
fps_to_ms, 326
GEAR_MAX_DEPLOY_SPEED, 326
GEAR_OPERATING_SPEED, 326
kg_to_pounds, 326
KU_OPERATING_SPEED, 326
LATCH_OPERATING_SPEED, 326
LATCH_STATE, 323
LAUNCH_SITE, 326
LBM, 326
LT_CLOSING, 323
LT_LATCHED, 323
LT_OPENING, 323
LT_RELEASED, 323
MAX_ATT_LAUNCH, 326
MAX_GRAPPLING_ANGLE, 326
MAX_GRAPPLING_DIST, 326
MAX_ROLL_SRB, 326
MESH_UNDEFINED, 326
MFDGROUPS, 326
MIDX_COCKPIT, 326
MIDX_ET, 328
MIDX_KU_ANTENNA, 328
MIDX_LSRB, 328
MIDX_MIDDECK, 328
MIDX_ODS, 328
MIDX_ODS_VC, 328
MIDX_ORBITER, 328
MIDX_PLPANEL_A7, 328
MIDX_PLPANEL_A8, 328
MIDX_PLPANEL_A9, 328
MIDX_RMS, 328
MIDX_RSRB, 328
MIDX_VC, 328
MPS2FPS, 328
NO_SLAG_TIME, 328
NosePitchThrusters, 328
NoseRollThrusters, 328
NoseRotThrusters, 328
NoseYawThrusters, 328
ODS_POS, 328
ODSPANEL_MESHNAME, 328
OFS_LAUNCH_LEFTSRB, 328
OFS_LAUNCH_ORBITER, 328
OFS_LAUNCH_RIGHTSRB, 328
OFS_LAUNCH_TANK, 328
OFS_MMU, 328
OFS_WITHTANK_ORBITER, 328
OFS_WITHTANK_TANK, 328
OFS_ZERO, 328
OMS_KIT, 323
OMS_LEFT, 323
OMS_RIGHT, 323

OMS_HELIUM_TANK_MASS, 328
OMS_REF, 323
ORBITER_CS, 328
ORBITER_CS_GEAR, 328
ORBITER_CW, 328
ORBITER_CW_GEAR, 328
ORBITER_DOCKPOS, 328
ORBITER_EMPTY_MASS, 328
ORBITER_FRCS_PROPELLANT_MASS, 328
ORBITER_MAIN_ISP0, 328
ORBITER_MAIN_ISP1, 328
ORBITER_MAIN_THRUST, 328
ORBITER_MAX_PROPELLANT_MASS, 328
ORBITER_MAX_THRUST, 328
ORBITER_OMS_ISP0, 328
ORBITER_OMS_ISP1, 328
ORBITER_OMS_PITCH, 328
ORBITER_OMS_THRUST, 328
ORBITER_OMS_YAW, 328
ORBITER_PITCH_TORQUE, 328
ORBITER_RCS_ISP0, 328
ORBITER_RCS_ISP1, 328
ORBITER_RCS_THRUST, 328
ORBITER_ROLL_TORQUE, 328
ORBITER_YAW_TORQUE, 328
PITCH, 322
POS_HDP, 328
PROBE_AXIS, 328
PROBEL_REF, 328
PROBER_REF, 328
RAD_OPERATING_SPEED, 328
RADLATCH_OPERATING_SPEED, 328
RCS_F1D, 328
RCS_F1D_DIR, 328
RCS_F1D_OFS, 328
RCS_F1F, 328
RCS_F1F_DIR, 328
RCS_F1F_OFS, 328
RCS_F1L, 328
RCS_F1L_DIR, 328
RCS_F1L_OFS, 328
RCS_F1U, 328
RCS_F1U_DIR, 328
RCS_F1U_OFS, 328
RCS_F2D, 328
RCS_F2D_DIR, 328
RCS_F2D_OFS, 328
RCS_F2F, 328
RCS_F2F_DIR, 328
RCS_F2F_OFS, 328
RCS_F2R, 328
RCS_F2R_DIR, 328
RCS_F2R_OFS, 328
RCS_F2U, 328
RCS_F2U_DIR, 328
RCS_F2U_OFS, 328
RCS_F3D, 328
RCS_F3D_DIR, 328
RCS_F3D_OFS, 328
RCS_F3F, 328
RCS_F3F_DIR, 328
RCS_F3F_OFS, 328
RCS_F3L, 328
RCS_F3L_DIR, 328
RCS_F3L_OFS, 328
RCS_F3U, 328
RCS_F3U_DIR, 328
RCS_F3U_OFS, 328
RCS_F4D, 328
RCS_F4D_DIR, 328
RCS_F4D_OFS, 328
RCS_F4R, 328
RCS_F4R_DIR, 328
RCS_F4R_OFS, 328
RCS_F5L, 328
RCS_F5R, 328
RCS_L1A, 328
RCS_L1L, 328
RCS_L1U, 328
RCS_L2D, 328
RCS_L2L, 328
RCS_L2U, 328
RCS_L3A, 328
RCS_L3D, 328
RCS_L3L, 328
RCS_L4D, 328
RCS_L4L, 328
RCS_L4U, 328
RCS_L5D, 328
RCS_L5R, 328
RCS_R1A, 328
RCS_R1R, 328
RCS_R1U, 328
RCS_R2D, 328
RCS_R2R, 328
RCS_R2U, 328
RCS_R3A, 328
RCS_R3D, 328
RCS_R3R, 328
RCS_R4D, 328
RCS_R4R, 328
RCS_R4U, 328
RCS_R5D, 328
RCS_R5R, 328
ROLL, 323
SHOULDER_BRACE_SPEED, 328

SLAG1_RATE, 328
SLAG2_RATE, 328
SLAG3_RATE, 328
SLAG_DIR, 328
SLAG_SPREAD, 328
SPEEDBRAKE_OPERATING_SPEED, 328
SRB_CUTOUT_TIME, 328
SRB_EMPTY_MASS, 328
SRB_ISP0, 328
SRB_ISP1, 328
SRB_MAX_PROPELLANT_MASS, 328
SRB_nt, 328
SRB_SEPARATION_TIME, 328
SRB_STABILISATION_TIME, 328
SRB_THRUST, 328
SSME_RATED_THRUST, 328
SSMEL_GOX_REF, 328
SSMEL_REF, 328
SSMER_GOX_REF, 328
SSMER_REF, 328
SSMET_GOX_REF, 328
SSMET_REF, 328
STATE_DESTROYED, 328
STATE_ORBITER, 328
STATE_PRELAUNCH, 328
STATE_STAGE1, 328
STATE_STAGE2, 328
SWITCH1, 322
SWITCH10, 322
SWITCH11, 322
SWITCH12, 322
SWITCH13, 322
SWITCH14, 322
SWITCH15, 322
SWITCH16, 322
SWITCH17, 322
SWITCH18, 322
SWITCH19, 322
SWITCH2, 322
SWITCH20, 322
SWITCH3, 322
SWITCH4, 322
SWITCH5, 322
SWITCH6, 322
SWITCH7, 322
SWITCH8, 322
SWITCH9, 322
SWITCH_HOLD_TIME, 328
TANK_EMPTY_MASS, 328
TANK_MAX_PROPELLANT_MASS, 328
TMajorCycle, 328
TPEGStop, 328
UMBDOOR_AXIS, 328
UMBDOORL_REF, 328
UMBDOORR_REF, 328
VARSTATE_HIGH, 328
VARSTATE_LOW, 328
VARSTATE_MISSING, 328
VARSTATE_OFFSCALE_HIGH, 328
VARSTATE_OFFSCALE_LOW, 328
VARSTATE_OK, 328
VC_AFTPILOT, 328
VC_AFTWORKSTATION, 328
VC_CDR, 328
VC_DIR_AFTPILOT, 328
VC_DIR_AFTWORKSTATION, 330
VC_DIR_COAS, 330
VC_DIR_EXT_AL, 330
VC_DIR_MIDDECK, 330
VC_DIR_MS1, 330
VC_DIR_MS2, 330
VC_DIR_PORTSTATION, 330
VC_DIR_RMSSTATION, 330
VC_DIR_STBDSTATION, 330
VC_DOCKCAM, 330
VC_EXT_AL, 330
VC_LBL_AFTPILOT, 330
VC_LBL_AFTWORKSTATION, 330
VC_LBL_CDR, 330
VC_LBL_COAS, 330
VC_LBL_DOCKCAM, 330
VC_LBL_EXT_AL, 330
VC_LBL_LEECAM, 330
VC_LBL_MIDDECK, 330
VC_LBL_MS1, 330
VC_LBL_MS2, 330
VC_LBL_PLBCAMBL, 330
VC_LBL_PLBCAMBR, 330
VC_LBL_PLBCAMFL, 330
VC_LBL_PLBCAMFR, 330
VC_LBL_PLT, 330
VC_LBL_PORTSTATION, 330
VC_LBL_RMSSTATION, 330
VC_LBL_STBDSTATION, 330
VC_LEECAM, 330
VC_MIDDECK, 330
VC_MS1, 330
VC_MS2, 330
VC_MS3, 330
VC_MS4, 330
VC_MS5, 330
VC_OFSFWD_AFTPILOT, 330
VC_PLBCAMBL, 330
VC_PLBCAMBR, 330
VC_PLBCAMFL, 330
VC_PLBCAMFR, 330
VC_PLT, 330
VC_PORTSTATION, 330

VC_POS_AFTPILOT, 330
VC_POS_AFTWORKSTATION, 330
VC_POS_CDR, 330
VC_POS_COAS, 330
VC_POS_EXT_AL, 330
VC_POS_MIDDECK, 330
VC_POS_MS1, 330
VC_POS_MS2, 330
VC_POS_PLT, 330
VC_POS_PORTSTATION, 330
VC_POS_RMSSTATION, 330
VC_POS_STBDSTATION, 330
VC_RMSSTATION, 330
VC_STBDSTATION, 330
YAW, 322
Atlantis_DlgProc
Atlantis.cpp, 306
Atlantis_SRB, 68
Atlantis_SRB, 68
Atlantis_SRB, 68
bMainEngine, 69
bSeparationEngine, 69
clbkPostCreation, 68
clbkPostStep, 68
clbkSetClassCaps, 68
hSRBMesh, 69
hSRBMesh_Left, 69
hSRBMesh_Right, 69
ph_main, 69
phBSM, 69
SetRefTime, 68
slag1, 69
slag2, 69
slag3, 69
srb_separation_time, 69
t0, 69
th_bolt, 69
th_main, 69
thBSM, 69
Atlantis_SRB.cpp
ORBITER_MODULE, 331
ovcExit, 331
ovcInit, 331
Atlantis_Tank, 70
Atlantis_Tank, 70
Atlantis_Tank, 70
clbkPostStep, 70
clbkSetClassCaps, 70
hTankMesh, 70
Atlantis_Tank.cpp
ORBITER_MODULE, 332
ovcExit, 332
ovcInit, 332
ovcSetState, 332
AtlantisAPUHYD, 71
~AtlantisAPUHYD, 72
AtlantisAPUHYD, 72
CalculateAPU, 72
clbkPostStep, 73
clbkPreStep, 72
fAPUControllerPower, 73
fAPUStartUpDelay, 73
fGGBedTemp, 73
fGGTemperature, 73
fLubeOilTemp, 73
fTurbineRPM, 73
fValve1Fail, 73
fValve1Pulse, 73
fValve2Fail, 73
fValve2Pulse, 73
GetAPUTurbineRPM, 73
GetGasGeneratorTemperature, 72
AtlantisAPUHYD.h
AFX_ATLANTISAPUHYD_H__76669820_-C41F_4F68_8755_0C199CFA6019__-INCLUDED_, 336
APU_1, 336
APU_2, 336
APU_3, 336
APU_INDEX, 336
AtlantisConfig, 74
Apply, 75
AtlantisConfig, 74
clbkOpen, 74
Description, 74
DlgProc, 75
InitDialog, 75
MshEnableHires, 75
MshHiresEnabled, 75
Name, 74
TexEnableHires, 75
TexHiresEnabled, 74
AtlantisConfig.cpp
ExitModule, 338
gParams, 338
hInst, 338
InitModule, 338
item, 338
msh_hires_bkup, 338
msh_lores_bkup, 338
ORBITER_MODULE, 338
STRICT, 338
tex_hires_disabled, 338
tex_hires_enabled, 338
vcmsh_fname, 338
AtlantisSubsystem, 77
~AtlantisSubsystem, 78
AddMeshes, 79

AtlantisSubsystem, 78
CountComponents, 80
DelComponent, 79
director, 82
GetIdentifier, 82
GetQualifiedIdentifier, 81
GetSubsystemEmptyMass, 82
ident, 82
OnDumpToLog, 81
OnFail, 82
OnParseLine, 81
OnPlaybackEvent, 81
OnPostStep, 80
OnPreStep, 80
OnPropagate, 80
OnSaveState, 81
Realize, 79
RegisterComponent, 79
STS, 78
AtlantisSubsystem.h
AFX_ATLANTISSUBSYSTEM_H_-_6A9F0F48_D391_4E11_9536_-F359182CA558__INCLUDED_, 340
AttControl
Atlantis, 55
AttDeadband
Atlantis, 66
ATVC
mps::ATVC, 83
AutoDoorSequenceControl
PayloadBayOp, 266
AutoGrappleSequence
Atlantis, 53
AutoMainGimbal
Atlantis, 55
AutoReleaseSequence
Atlantis, 53
availableForMouse
vc::BasicPanel, 96
availableForRendering
vc::BasicVCComponent, 109
AXIS
Atlantis.h, 322
AXIS_TILT
Atlantis.h, 325

B
Atlantis, 66
b0
Atlantis, 55
BasicExternalAirlock
eva_docking::BasicExternalAirlock, 86
BasicLight
vc::BasicLight, 90
BasicPanel
vc::BasicPanel, 92
BasicSwitch
vc::BasicSwitch, 98
BasicSwitch.h
AFX_BASICSWITCH_H__3790BCEE_-A9B9_492A_A7F4_B0DA5E454AFB_-_INCLUDED_, 505
BasicValve, 100
~BasicValve, 101
_backdoor, 101
BasicValve, 101
Close, 101
GetPos, 101
maxrate, 101
Move, 101
mpos, 101
mrate, 101
Open, 101
pos, 101
tmestp, 101
BasicVCComponent
vc::BasicVCComponent, 104
BasicVCComponent.h
AFX_BASICVCCOMPONENT_H__-B74A68D7_2919_4C3F_AB0C_-4B4FBA3AA2B7__INCLUDED_, 507
bAutopilot
Atlantis, 66
BayDoor
PayloadBayOp, 269
BayDoorOp
PayloadBayOp, 269
BayDoorStatus
PayloadBayOp, 269
bBrokenLine
discsignals::DiscreteBundle, 141
bCommMode
Atlantis, 62
CommModeHandler, 119
bCoordinateDisplayMode
vc::BasicPanel, 96
BD_DISABLE
PayloadBayOp, 263
BD_ENABLE
PayloadBayOp, 263
BDO_CLOSE
PayloadBayOp, 263
BDO_OPEN
PayloadBayOp, 263
BDO_STOP
PayloadBayOp, 263
bEngineFail

Atlantis, 66
beta_angle
KuBandSystem, 181
bfs_ops
dps::IDP, 168
bfs_specdisp
dps::IDP, 168
bGrappleInProgress
Atlantis, 66
bHasAnimations
vc::BasicVCComponent, 109
bHasDirection
vc::BasicVCComponent, 109
bHasKUBand
Atlantis, 60
bHasODS
Atlantis, 60
bHasOwnVCMesh
vc::BasicPanel, 96
bHasReference
vc::BasicVCComponent, 109
bHeatersOn
KuBandSystem, 181
bHydraulicPressure
PanelR2, 260
bInverseX
vc::MDU, 198
bIsBusMaster
dps::BIU, 111
bIsConnectedToCRTMFD
vc::MDU, 198
bIsPulled
vc::LockableLever, 183
BitmapLocation
vc, 25
BIU
dps::BIU, 111
BLACK
CRT.cpp, 350
BlankReceiveAnnotation
CommModeHandler, 118
BlankSendAnnotations
CommModeHandler, 118
bLiftOff
Atlantis, 60
BLOCK_II
mps::BLOCK_II, 113
BLOCK_II_FPL_THRUST
mps, 23
BLOCK_II_ISP0
mps, 23
BLOCK_II_ISP1
mps, 23
BLOCK_II_RPL_THRUST
mps, 23
BLUE
CRT.cpp, 350
bMainEngine
Atlantis_SRB, 69
bManualSeparate
Atlantis, 58
bMECO
Atlantis, 66
bMETCounting
dps::MasterTimingUnit, 189
bMidDeckVisible
Atlantis, 60
bn
Atlantis, 55
BODY_VECT
Atlantis, 66
BOILER_CNTLR
PanelR2, 260
BOILER_CNTLRHTR
PanelR2, 260
BOILER_N2_SUPPLY
PanelR2, 260
bOrientation
vc::BasicSwitch, 99
bPower
vc::MDU, 198
bReleaseInProgress
Atlantis, 66
BRIGHTERGRAY
CRT.cpp, 350
bSeparationEngine
Atlantis_SRB, 69
bsm_contrail
Common.cpp, 345
bsm_exhaust
Common.cpp, 346
BSM_ISP0
Atlantis.h, 325
BSM_MAX_PROPELLANT_MASS
Atlantis.h, 325
bsm_residual
Common.cpp, 346
BSM_THRUST0
Atlantis.h, 325
bSRBCutoffFlag
Atlantis, 60
bSSMEGOXVent
Atlantis, 66
bSSS_Active
MLP, 207
bStartSequence
MLP, 207
bThrottle

Atlantis, 66
bTIMER
CRT, 130
BUFFER_SSME_DATA_TABLE
mps::EIU, 149
BundleManager
Atlantis, 53
SubsystemDirector, 296
bundleManager
Atlantis, 66
bundles
discsignals::DiscreteBundleManager, 143
BurnAtt
Atlantis, 66
BurnCompleted
Atlantis, 66
BurnInProg
Atlantis, 66
BurnTime
Atlantis, 66
bUseCRTMFD
vc::MDU, 198
bUseKeyboardA
dps::IDP, 168
bUseRealRCS
Atlantis, 62
bUseRMS
mission::Mission, 204
bUseSecondaryPort
vc::MDU, 198
ButtonLabel
CRT, 123
ButtonMenu
CRT, 123
bValid
CRT::SavePrm, 132
bZThrust
Atlantis, 66

C
Atlantis, 66
c0
Atlantis, 56
C1
Atlantis, 66
C2
Atlantis, 66
c3po
Atlantis, 60
cache_lastx
vc::BasicVCComponent, 109
cache_lasty
vc::BasicVCComponent, 109
cache_mx
vc::BasicVCComponent, 109
cache_my
vc::BasicVCComponent, 109
CalcAnimationFKArm
Atlantis, 53
CalcLVLHAttitude
Atlantis, 55
CalcManeuverTargets
Atlantis, 55
CalcNetThrust
Atlantis, 55
CalcPitchYawRollAngles
Atlantis, 55
CalcRelLVLHAttitude
Atlantis, 55
CalcRequiredRates
Atlantis, 55
CalcThrustAngles
Atlantis, 55
CalculateAPU
AtlantisAPUHYD, 72
CalculateAzimuth
Atlantis, 55
CalculateN2System
OMSPod, 218
CalculateSteamProduction
MLP, 206
CAM_HIGHRATE_SPEED
Atlantis.h, 325
CAM_LOWRATE_SPEED
Atlantis.h, 325
camBLpitch
Atlantis, 62
camBLyaw
Atlantis, 62
camBRpitch
Atlantis, 62
camBRyaw
Atlantis, 62
CameraBLPitch
Atlantis, 62
CameraBLYaw
Atlantis, 62
CameraBRPitch
Atlantis, 62
CameraBRYaw
Atlantis, 62
cameraControl
Atlantis, 62
CameraFLPitch
Atlantis, 62
CameraFLYaw
Atlantis, 62
CameraFRPitch

Atlantis, 62
CameraFRYaw
Atlantis, 62
cameraLowRate
Atlantis, 62
cameraMoved
Atlantis, 62
camFLpitch
Atlantis, 62
camFLyaw
Atlantis, 62
camFRpitch
Atlantis, 62
camFRyaw
Atlantis, 62
CanArrest
Atlantis, 55
cargo_static_mesh_name
Atlantis, 60
cargo_static_ofs
Atlantis, 60
cAttr
dps::DEUCHAR, 135
cbChangeOps
CRT.cpp, 350
cbItem
CRT.cpp, 350
cbItemData
CRT.cpp, 351
cbSpec
CRT.cpp, 351
cbSpecData
CRT.cpp, 351
cdBase
Atlantis.cpp, 308
cDispTitle
CRT, 130
CDRKeyboard
Atlantis, 60
center_arm
Atlantis, 60
center_arm_t
Atlantis, 60
CENTERLINE_LATCH
PanelR2, 260
CenterlineLatches
PanelR2, 260
CheckAPUReadytoStart
PanelR2, 258
CheckAPUShutdown
PanelR2, 258
CheckMPSArmed
PanelR2, 258
CheckOMSArm
PanelC3, 238
CHECKOUT_COMPONENTCHECKOUT
mps, 22
CHECKOUT_STANDBY
mps, 22
CheckProbesDeployed
PanelC3, 238
CheckValidity
mission::Mission, 204
clBase
Atlantis.cpp, 308
CLBayDoorLatch
PayloadBayOp, 269
clbkAnimate
Atlantis, 53
clbkConsumeBufferedKey
Atlantis, 53
clbkDrawHUD
Atlantis, 53
clbkFocusChanged
Atlantis, 53
clbkLoadGenericCockpit
Atlantis, 53
clbkLoadStateEx
Atlantis, 53
MLP, 206
clbkLoadVC
Atlantis, 53
MLP, 206
clbkMFDMode
Atlantis, 53
clbkOpen
AtlantisConfig, 74
clbkPlaybackEvent
Atlantis, 53
clbkPostCreation
Atlantis_SRB, 68
clbkPostStep
Atlantis, 53
Atlantis_SRB, 68
Atlantis_Tank, 70
AtlantisAPUHYD, 73
clbkPreStep
Atlantis, 53
AtlantisAPUHYD, 72
MLP, 206
clbkSaveState
Atlantis, 53
MLP, 206
clbkSetClassCaps
Atlantis, 53
Atlantis_SRB, 68
Atlantis_Tank, 70
MLP, 206

clbkSetStateEx
Atlantis, 53
clbkVCMouseEvent
Atlantis, 53
clbkVCRedrawEvent
Atlantis, 53
clbkVisualCreated
Atlantis, 53
clbkVisualDestroyed
Atlantis, 53
ClearInput
Keyboard, 178
ClearMeshes
Atlantis, 53
ClearRecAnnotations
CommModeHandler, 117
ClearSendAnnotations
CommModeHandler, 117
clock_digits
GDIParams, 155
CloneNewBranch
mission::Mission, 204
CloneRevision
mission::Mission, 204
Close
BasicValve, 101
ValveTypeBool, 297
cmBase
Atlantis.cpp, 309
cmdChannelReset
mps::SSME, 281
cmdCheckoutStandby
mps::SSME, 281
cmdControllerReset
mps::SSME, 281
CmdDriveFwd
MechActuator, 201
CmdDriveRwd
MechActuator, 201
cmdExitPROM
mps::SSME, 281
cmdFRT1
mps::SSME, 281
cmdOxidizerDump
mps::SSME, 281
cmdPC
mps::_DATA_TABLE, 28
CmdPDot
Atlantis, 66
cmdPowerOff
mps::SSME, 281
cmdPowerOn
mps::SSME, 281
cmdPurgeSequence1
mps::SSME, 280
cmdPurgeSequence2
mps::SSME, 281
cmdPurgeSequence3
mps::SSME, 281
cmdPurgeSequence4
mps::SSME, 281
CmdRDot
Atlantis, 66
cmdShutdown
mps::SSME, 281
cmdShutdownEnable
mps::SSME, 281
cmdStart
mps::SSME, 281
cmdStartEnable
mps::SSME, 281
cmdTerminateSequence
mps::SSME, 281
cmdThrottle
mps::SSME, 281
CmdYDot
Atlantis, 66
cmTrim
Atlantis.cpp, 309
cn
Atlantis, 56
COMM_ROLE
CommModeHandler.h, 344
CommandN2PressIsolValve
OMSPod, 218
CommModeHandler, 116
bCommMode, 119
BlankReceiveAnnotation, 118
BlankSendAnnotations, 118
ClearRecAnnotations, 117
ClearSendAnnotations, 117
CommModeHandler, 117
ConsumeBufferedKey, 117
DefineAnnotations, 117
EnterCommMode, 118
fRecDisplayT0, 119
fSendDisplayT0, 119
iMenu, 119
IsInCommMode, 118
LeaveCommMode, 118
nhLastReceived, 119
nhLastSend, 119
nhOptionPos, 119
OnLooseFocus, 118
OnReceiveFocus, 117
PostStep, 117
pSTS, 119
strCompA2G, 119

strDecompA2G, 119
strDecompG2A, 119
strNegAnswer_Comp, 119
strNegAnswer_Decomp, 119
strPosAnswer_Comp, 119
strPosAnswer_Decomp, 119
CommModeHandler.h
COMM_ROLE, 344
COMMROLE_CDR_ASCENT, 344
COMMROLE_CDR_ORBIT, 344
COMMROLE_CDR_PRELAUNCH, 344
Common.cpp
bsm_contrail, 345
bsm_exhaust, 346
bsm_residual, 346
GetSRB_State, 345
srb_contrail, 346
srb_exhaust, 346
SRB_Prop, 346
SRB_Seq, 346
srb_slag1, 347
srb_slag2, 347
srb_slag3, 347
SRB_Thrust, 347
COMMROLE_CDR_ASCENT
CommModeHandler.h, 344
COMMROLE_CDR_ORBIT
CommModeHandler.h, 344
COMMROLE_CDR_PRELAUNCH
CommModeHandler.h, 344
components
vc::BasicPanel, 96
Connect
discsignals::DiscretePort, 145
vc::BasicLight, 90
ConnectPort
vc::StandardSwitch, 289
vc::StdSwitch2, 291
vc::StdSwitch3, 293
ConnectToBus
dps::BIU, 111
ConnectToCRTMFD
vc::MDU, 197
ConnectToKeyboard
dps::IDP, 168
ConnectToMDU
dps::IDP, 166
ConsumeBufferedKey
CommModeHandler, 117
ConsumeButton
CRT, 123
ConsumeKeyBuffered
CRT, 123
ControlSurfacesEnabled
Atlantis, 66
convert
CRT.h, 353
ConvertAnglesBetweenM50AndOrbiter
Atlantis, 55
ConvertLVLHAnglesToM50
Atlantis, 55
ConvertOrbiterAnglesToLocal
Atlantis, 55
ConvertVectorBetweenOrbiterAndM50
Atlantis, 55
COtimecoef
mps::SSME, 283
COUNT_DOWN
dps, 15
COUNT_STARTED
dps, 14
COUNT_STOPPED
dps, 14
COUNT_TEST
dps, 15
COUNT_UP
dps, 15
CountBundles
discsignals::DiscreteBundleManager, 143
CountComponents
AtlantisSubsystem, 80
ISubsystem, 175
vc::BasicVCComponent, 107
counting
vc::MDU, 198
CreateAttControls_RCS
Atlantis, 53
CreateBundle
discsignals::DiscreteBundleManager, 143
CreateFRCS
Atlantis, 53
CreateLeftARCS
Atlantis, 53
CreateMPSGOXVents
Atlantis, 53
CreateOrbiterTanks
Atlantis, 53
CreateRightARCS
Atlantis, 53
CreateSwitch2
vc::BasicPanel, 93
CreateSwitch3
vc::BasicPanel, 93
CreateSwitch3H
vc::BasicPanel, 93
CreateVisual
PanelC2, 235
CRT, 120

~CRT, 123
AngularVelocity, 130
APU_Fuel_Last, 130
APU_Fuel_Rate, 130
APUHYD, 126
Atlantis, 57, 130
bTIMER, 130
ButtonLabel, 123
ButtonMenu, 123
cDispTitle, 130
ConsumeButton, 123
ConsumeKeyBuffered, 123
CRT, 123
CRTLine, 128
CRTTextOut, 128
DAP_CONFIG, 126
Data, 125
data, 130
DisableThrusters, 127
display, 130
DisplayScratchPad, 127
DrawCommonHeader, 128
DrawDelta, 127
dT, 130
edit, 130
EnableThrusters, 127
Eq, 126
GNCSYSSUMM1, 128
GPCMEMORY, 126
GVesselPos, 130
GVesselVel, 130
hCRTFont, 130
height, 130
Hydraulic_Press, 130
id, 130
Input, 125
item, 130
Keyboard, 130, 178
landed, 130
Launch_time, 130
LoadManeuver, 127
MET, 130
MET_Add, 130
MNVR, 126
mode, 130
MPS_He_Reg_Press, 130
MPS_He_Tank_Press, 130
MPS_Manifold_Press, 130
MsgProc, 124
OMSGimbal, 127
OMSMPS, 126
ops, 130
PanelA8, 233
PanelC2, 236
PanelR2, 260
PASSTRAJ, 128
PDRS, 127
PMI, 130
ReadStatus, 126
RecallCriticalFormat, 128
RecallStatus, 126
RMS_SEL, 130
saveprm, 130
SetDisplayTitle, 126
Simtime, 130
Simtime_last, 130
spec, 130
SPI, 126
Status, 130
StoreStatus, 126
sts, 130
TGO, 130
ToDeg, 126
ToRad, 126
UNIVPTG, 126
Update, 124
UpdateDAP, 127
UpdateStatus, 130
usDisp, 130
usGPCDriver, 130
usMDU, 130
usPageNumber, 130
vessel, 130
width, 130
WriteStatus, 125
CRT.cpp
BLACK, 350
BLUE, 350
BRIGHTERGRAY, 350
cbChangeOps, 350
cbItem, 350
cbItemData, 351
cbSpec, 351
cbSpecData, 351
DARK_GREEN, 350
DARK_YELLOW, 350
ExitModule, 352
g_MFDmode, 352
GRAY, 350
GREEN, 350
InitModule, 352
mfd, 352
mfd_gparam, 352
ORBITER_MODULE, 350
RED, 350
STRICT, 350
WHITE, 350
YELLOW, 350

CRT.h
convert, 353
STD_CLASS_NAME, 353
CRT::SavePrm, 132
bValid, 132
display, 132
mode, 132
spec, 132
CRT_SEL
Atlantis, 66
CRTLine
CRT, 128
CRTTextOut
CRT, 128
cScratchPadLine
dps::IDP, 168
cSymbol
dps::DEUCHAR, 135
CurrentAttitude
Atlantis, 66

d1
Atlantis, 66
d2
Atlantis, 66
d3
Atlantis, 66
d4
Atlantis, 66
DamageGear
GearOp, 159
DAP
Atlantis, 66
DAP_CONFIG
CRT, 126
DAPConfig, 133
ALT_DELAY, 134
ALT_JET_OPT, 134
ALT_JETS, 134
ALT_ON_TIME, 134
ALT_RATE_DB, 134
PRI_ATT_DB, 134
PRI_COMP, 134
PRI_P_OPTION, 134
PRI_RATE_DB, 134
PRI_ROT_PLS, 134
PRI_ROT_RATE, 134
PRI_TRAN_PLS, 134
PRI_Y_OPTION, 134
VERN_ATT_DB, 134
VERN_CNTL_ACC, 134
VERN_COMP, 134
VERN_RATE_DB, 134
VERN_ROT_PLS, 134
VERN_ROT_RATE, 134
DAPMode
Atlantis, 66
DARK_GREEN
CRT.cpp, 350
DARK_YELLOW
CRT.cpp, 350
Data
CRT, 125
data
CRT, 130
DATA_TABLE
mps, 22
dataDataTable
mps::SSME, 281
DataInput
Atlantis, 66
dcCCV_CO
mps::BLOCK_II, 114
mps::SSME, 279
dcCCV_ESC
mps::BLOCK_II, 113
mps::SSME, 279
dcCCV_MS
mps::BLOCK_II, 114
mps::SSME, 280
dcFPOV_CO
mps::BLOCK_II, 114
mps::SSME, 279
dcFPOV_ESC
mps::BLOCK_II, 113
mps::SSME, 279
dcFPOV_MS
mps::BLOCK_II, 114
mps::SSME, 280
dcMFV_CO
mps::BLOCK_II, 114
mps::SSME, 279
dcMFV_ESC
mps::BLOCK_II, 113
mps::SSME, 279
dcMOV_CO
mps::BLOCK_II, 113
mps::SSME, 279
dcMOV_ESC
mps::BLOCK_II, 113
mps::SSME, 278
dcOPOV_CO
mps::BLOCK_II, 114
mps::SSME, 279
dcOPOV_ESC
mps::BLOCK_II, 113
mps::SSME, 279
dcOPOV_MS

mps::BLOCK_II, 114
mps::SSME, 280
dcPC_CO
mps::BLOCK_II, 113
mps::SSME, 279
dcPC_ESC
mps::BLOCK_II, 113
mps::SSME, 278
dcPC_MS
mps::BLOCK_II, 114
mps::SSME, 279
DCU
mps, 22
DCU_A
mps, 22
DCU_B
mps, 22
DEFAULT_MESHNAME_COCKPIT
Atlantis.h, 325
DEFAULT_MESHNAME_ET
Atlantis.h, 325
DEFAULT_MESHNAME_KU
Atlantis.h, 325
DEFAULT_MESHNAME_LSRB
Atlantis.h, 325
DEFAULT_MESHNAME_MIDDECK
Atlantis.h, 325
DEFAULT_MESHNAME_ODS
Atlantis.h, 325
DEFAULT_MESHNAME_ORBITER
Atlantis.h, 325
DEFAULT_MESHNAME_RMS
Atlantis.h, 325
DEFAULT_MESHNAME_RSRB
Atlantis.h, 325
DEFAULT_MESHNAME_VC
Atlantis.h, 325
defaultStage1Guidance
Atlantis.h, 325
DefineAnimations
Atlantis, 53
GearOp, 158
PanelA4, 223
PanelC2, 235
PanelC3, 238
PanelF7, 246
PanelO3, 253
PanelR2, 257
PayloadBayOp, 265
SSU_FSS_RSS, 285
DefineAnnotations
CommModeHandler, 117
DefineKUBandAnimations
Atlantis, 53
DefinePanelMouseArea
vc::BasicPanel, 93
DefineRegionAID
vc::MDU, 196
DefineSSMEExhaust
Atlantis, 53
DefineSwitches
vc::BasicPanel, 93
DefineSwitchGroup
vc::StandardSwitch, 288
DefineVC
vc::BasicPanel, 94
vc::PanelA7A8ODS, 227
vc::PanelF6, 241
vc::PanelF7, 244
vc::PanelF8, 248
vc::PanelGroup, 251
DefineVCAnimations
GearOp, 158
PanelA4, 223
PanelA8, 232
PanelC2, 235
PanelC3, 238
PanelF7, 246
PanelO3, 253
PanelR2, 257
vc::BasicPanel, 93
vc::BasicSwitch, 98
vc::BasicVCComponent, 105
vc::MDU, 195
vc::PanelA7A8ODS, 227
vc::PanelGroup, 251
vc::RussianCB, 273
vc::StandardSwitch, 288
DefineVCGroup
vc::MDU, 196
DefineVCTexture
vc::MDU, 196
DelComponent
AtlantisSubsystem, 79
ISubsystem, 175
vc::BasicVCComponent, 106
DeleteAllComponents
vc::BasicPanel, 93
deltah
Atlantis, 66
deltatheta
Atlantis, 66
DeltaV
Atlantis, 66
deltav
Atlantis, 66
DeltaVTot
Atlantis, 66

DeltaVTotms
Atlantis, 66
DEPLOY_MODE
AirDataProbeSystem, 32
DEPLOY_STATE
AirDataProbeSystem, 32
DeployState
KuBandSystem, 181
Description
AtlantisConfig, 74
deu_characters
GDIParams, 155
DEU_KEY_RESUME
Keyboard.h, 409
DEUATT_FLASHING
dps, 15
DEUATT_NORMAL
dps, 15
DEUATT_OVERBRIGHT
dps, 15
DEUATT_UNDERLINED
dps, 15
DeuCharBitmapDC
GDIParams, 155
digit_texture
PanelC2, 236
digits_7seg
GDIParams, 155
dilModeA
vc::EventTimer, 151
dilModeB
vc::EventTimer, 151
dilReset
vc::EventTimer, 151
dilSet
vc::EventTimer, 151
dilStart
vc::EventTimer, 151
dilTest
vc::EventTimer, 151
dir
mps::SSME, 283
vc::BasicVCComponent, 109
DIR_MPS_SMOKE
MLP.h, 449
DIR_SRB_SMOKE
MLP.h, 449
director
AtlantisSubsystem, 82
OMSPod, 218
DisableAllRCS
Atlantis, 53
DisableControlSurfaces
Atlantis, 53
DisableCoordinateDisplayMode
vc::BasicPanel, 96
vc::PanelGroup, 251
DisableThrusters
Atlantis, 55
CRT, 127
DiscInPort
discsignals::DiscInPort, 137
Disconnect
discsignals::DiscretePort, 145
DiscOutPort
discsignals::DiscOutPort, 138
DiscOutPort.h
AFX_DISCOUTPORT_H__B01F0E2B_-BD2E_4A08_B236_7DCF6C1507CD__-INCLUDED_, 358
DiscreteBundle
discsignals::DiscreteBundle, 140
DiscreteBundle.h
AFX_DISCRETEBUNDLE_H__-A928D10B_004B_495D_87CD_-3A0309B3EFDA__INCLUDED_, 360
DiscreteBundleManager
discsignals::DiscreteBundleManager, 143
DiscretePort
discsignals::DiscretePort, 144
DiscretePort.h
AFX_DISCRETEPORT_H__18D00C82_-9652_493D_8A1D_D1CFF223A0B8__-INCLUDED_, 364
discsignals, 13
discsignals::DiscInPort, 136
~DiscInPort, 137
DiscInPort, 137
GetVoltage, 137
IsSet, 137
discsignals::DiscOutPort, 138
~DiscOutPort, 138
DiscOutPort, 138
ResetLine, 139
SetLine, 138
discsignals::DiscreteBundle, 140
~DiscreteBundle, 140
bBrokenLine, 141
DiscreteBundle, 140
fLines, 141
fState, 141
GetIdentity, 141
GetVoltage, 140
ident, 141
SetDiscrete, 141
SetLogMode, 140
snLines, 141

discsignals::DiscreteBundleManager, 142
~DiscreteBundleManager, 143
AddBundle, 143
bundles, 143
CountBundles, 143
CreateBundle, 143
DiscreteBundleManager, 143
DumpToLog, 143
ExistsBundle, 143
FindBundle, 143
GetBundleByIndex, 143
RemoveBundle, 143
discsignals::DiscretePort, 144
~DiscretePort, 144
Connect, 145
Disconnect, 145
DiscretePort, 144
pBundle, 145
usLine, 145
Display
Atlantis, 66
display
CRT, 130
CRT::SavePrm, 132
DisplayCameraLabel
Atlantis, 57
DisplayJointAngles
Atlantis, 66
DisplayScratchPad
CRT, 127
DistributeMouseEvent
vc::BasicPanel, 93
DlgProc
AtlantisConfig, 75
PayloadBayOp, 268
do_cargostatic
Atlantis, 60
do_eva
Atlantis, 60
do_plat
Atlantis, 60
DOOR_CLOSED
SSU_FSS_RSS, 284
DOOR_CLOSING
SSU_FSS_RSS, 284
DOOR_OPEN
SSU_FSS_RSS, 284
DOOR_OPENING
SSU_FSS_RSS, 284
DOOR_OPERATING_SPEED
Atlantis.h, 326
DOORLATCH_OPERATING_SPEED
Atlantis.h, 326
DoorStatus
SSU_FSS_RSS, 284
dps, 14
COUNT_DOWN, 15
COUNT_STARTED, 14
COUNT_STOPPED, 14
COUNT_TEST, 15
COUNT_UP, 15
DEUATT_FLASHING, 15
DEUATT_NORMAL, 15
DEUATT_OVERBRIGHT, 15
DEUATT_UNDERLINED, 15
EVENT_TIMER_CONTROL, 14
EVENT_TIMER_MODE, 14
GNC, 15
MAJORFUNCTION, 15
MTU_ACCU_INDEX, 14
MTU_EVTTMR_INDEX, 14
PL, 15
SM, 15
TIMER_AFT, 15
TIMER_FORWARD, 15
dps::AP101S, 35
~AP101S, 36
AP101S, 36
lMemory, 36
LoadMemoryConfiguration, 36
LoadSoftware, 36
dps::BIU, 110
~BIU, 111
bIsBusMaster, 111
BIU, 111
ConnectToBus, 111
Listen, 111
MakeBusMaster, 111
OnNewMessage, 111
pBus, 111
pRAMBuffer, 111
dps::DEUCHAR, 135
cAttr, 135
cSymbol, 135
dps::IDP, 165
~IDP, 166
bfs_ops, 168
bfs_specdisp, 168
bUseKeyboardA, 168
ConnectToKeyboard, 168
ConnectToMDU, 166
cScratchPadLine, 168
GetIDPID, 168
gnc_ops, 168
gnc_specdisp, 168
IDP, 166
IPL, 166
ipl_software, 168

KeybSelectA, 168
KeybSelectB, 168
majfunc, 168
MajorFuncGNC, 168
MajorFuncPL, 168
mdu_list, 168
pKeyboardA, 168
pKeyboardB, 168
pl_ops, 168
pl_specdisp, 168
sm_ops, 168
sm_specdisp, 168
sScratchPadLength, 168
usDISP, 168
usGPCDriver, 168
usIDPID, 168
usOPS, 168
usSelectedFC, 168
usSPEC, 168
dps::IDPSoftware, 170
~IDPSoftware, 171
IDPSoftware, 171
pIDP, 171
Run, 171
dps::MasterTimingUnit, 184
~MasterTimingUnit, 187
bMETCounting, 189
event_control, 189
event_mode, 189
fEvent, 189
fGMT, 189
fMET, 189
GetEventTimerMin, 187
GetEventTimerSec, 187
GetGMTDay, 187
GetGMTHour, 187
GetGMTMilli, 187
GetGMTMin, 187
GetGMTSec, 187
GetMETDay, 187
GetMETHour, 187
GetMETMilli, 187
GetMETMin, 187
GetMETSec, 187
MasterTimingUnit, 187
OnParseLine, 188
OnPreStep, 187
OnPropagate, 188
OnSaveState, 188
ResetEventTimer, 187
ResetMET, 187
SetEventTimer, 187
SetEventTimerMode, 187
sEventMinutes, 189
sEventSeconds, 189
sGMTDays, 189
sGMTHours, 189
sGMTMillis, 189
sGMTMinutes, 189
sGMTSeconds, 189
sMETDays, 189
sMETHours, 189
sMETMillis, 189
sMETMinutes, 189
sMETSeconds, 189
StartEventTimer, 187
StartMET, 187
StopEventTimer, 187
dps::MDM, 190
~MDM, 191
MDM, 191
pmodules, 191
SCU_PROM, 191
dps::RAM16, 270
~RAM16, 270
GetDWord, 270
GetWord, 270
OnParseLine, 270
OnSaveState, 270
pMemory, 270
RAM16, 270
ReadImage, 270
SetWord, 270
WriteImage, 270
dps::ShuttleBus, 274
~ShuttleBus, 274
ShuttleBus, 274
DrawCommonHeader
CRT, 128
DrawDelta
CRT, 127
DrawPower
eps::IACElement, 163
eps::IDCElement, 164
dT
CRT, 130
DumpToLog
discsignals::DiscreteBundleManager, 143
dV
Atlantis, 66
DYNP_MAX
Atlantis.h, 326

eCurrent
Atlantis, 66
edit
Atlantis, 66
CRT, 130

EEGrappleMode
Atlantis, 66
EIU
mps::EIU, 147
el
Atlantis, 66
elbow_max
Atlantis, 66
elbow_min
Atlantis, 66
elbow_neutral
Atlantis, 66
elbow_range
Atlantis, 66
EnableAllRCS
Atlantis, 53
EnableControlSurfaces
Atlantis, 53
EnableCoordinateDisplayMode
vc::BasicPanel, 96
vc::PanelGroup, 251
EnableThrusters
Atlantis, 55
CRT, 127
EngControl
PanelC3, 239
engine
mps::EIU, 149
ENGINE_STATUS
mps, 22
EngineFail
Atlantis, 66
EngineFailTime
Atlantis, 66
EngineNullPosition
Atlantis, 66
EnterCommMode
CommModeHandler, 118
ep_angle
Atlantis, 66
ep_null
Atlantis, 66
eps, 16
ACPHASE, 16
PHASE1, 16
PHASE2, 16
PHASE3, 16
eps::IACElement, 163
DrawPower, 163
GetAmps, 163
GetVoltage, 163
eps::IDCElement, 164
DrawPower, 164
GetAmps, 164
GetPower, 164
GetVoltage, 164
Eq
Atlantis, 55
CRT, 126
Estimate
Atlantis, 55
ET_SEP_RATE
Atlantis.h, 326
ET_SEP_TIME
Atlantis.h, 326
ET_UMB_DOOR_OPERATING_SPEED
Atlantis.h, 326
eva_docking, 17
eva_docking::BasicExternalAirlock, 85
~BasicExternalAirlock, 86
AddMeshes, 86
BasicExternalAirlock, 86
GetSubsystemEmptyMass, 86
OnDumpToLog, 88
OnParseLine, 87
OnPlaybackEvent, 87
OnPostStep, 87
OnPreStep, 86
OnPropagate, 87
OnSaveState, 87
Realize, 86
eva_docking::ExtAirlock, 152
~ExtAirlock, 153
AddMeshes, 153
ExtAirlock, 153
fExtALPress, 153
fHatchState, 153
eva_docking::ODS, 213
~ODS, 214
AddMeshes, 214
fHookState, 214
fLatchState, 214
fRingState, 214
fVestPressure, 214
GetSubsystemEmptyMass, 214
midxODS, 214
ODS, 214
event_control
dps::MasterTimingUnit, 189
event_mode
dps::MasterTimingUnit, 189
EVENT_TIMER_CONTROL
dps, 14
EVENT_TIMER_MODE
dps, 14
EventTimer
vc::EventTimer, 151
EventTimerControl

PanelC2, 236
EventTimerMode
PanelC2, 236
EventTimerWheelRevs
PanelC2, 236
EVTTMR_WHEELMAX
PanelC2.cpp, 472
EVTTMR_WINDOW_GROUPS
PanelC2.cpp, 472
ex_main
Atlantis, 64
ex_retro
Atlantis, 66
EXEC
KeyboardInput, 179
ExistsBundle
discsignals::DiscreteBundleManager, 143
ExitModule
Atlantis.cpp, 306
AtlantisConfig.cpp, 338
CRT.cpp, 352
MLP.cpp, 448
ExtAirlock
eva_docking::ExtAirlock, 153
Extend
Atlantis, 66

F
Atlantis, 66
FailEngine
Atlantis, 55
fAPUControllerPower
AtlantisAPUHYD, 73
fAPUStartUpDelay
AtlantisAPUHYD, 73
fBrightness
vc::MDU, 198
fCountdown
MLP, 207
fDeployPos
KuBandSystem, 181
fdotdottheta
Atlantis, 66
fdotr
Atlantis, 66
fdottheta
Atlantis, 66
fEvent
dps::MasterTimingUnit, 189
fEventTime
vc::EventTimer, 151
fExtALPress
eva_docking::ExtAirlock, 153
fGGBedTemp
AtlantisAPUHYD, 73
fGGTemperature
AtlantisAPUHYD, 73
fGimbalLockState
KuBandSystem, 181
fGMT
dps::MasterTimingUnit, 189
fh
Atlantis, 66
fHatchState
eva_docking::ExtAirlock, 153
fhdotrh
Atlantis, 66
fHookState
eva_docking::ODS, 214
FindBundle
discsignals::DiscreteBundleManager, 143
FindComponent
vc::BasicPanel, 93
fInitialAnimState
vc::BasicVCComponent, 109
fInitialMousePos
vc::LockableLever, 183
FireAllNextManifold
Atlantis, 53
fLatchState
eva_docking::ODS, 214
FlatPlateCoeff
Atlantis.cpp, 307
fLines
discsignals::DiscreteBundle, 141
FLOWSTATE, 154
mdot, 154
P, 154
T, 154
fLTimeMJD
mission::Mission, 204
fLubeOilTemp
AtlantisAPUHYD, 73
fMET
dps::MasterTimingUnit, 189
fN2PressIsolVlv
OMSPod, 218
fN2TankMass
OMSPod, 218
fObjectTorque
MechActuator, 201
font
GDIParams, 155
fOutPos
MechActuator, 201
FPL
mps::SSME, 283
FPL_THRUST

mps::SSME, 283
FPOV_pos_CO
mps::SSME, 283
fps_to_ms
Atlantis.h, 326
fr
Atlantis, 66
fRecDisplayT0
CommModeHandler, 119
fRingState
eva_docking::ODS, 214
fRunning
vc::EventTimer, 151
fSendDisplayT0
CommModeHandler, 119
fSRBSteam
MLP, 207
fSSMESteam
MLP, 207
fState
discsignals::DiscreteBundle, 141
fT_SSSActive
MLP, 207
ftheta
Atlantis, 66
fTimeCameraLabel
Atlantis, 66
fTurbineRPM
AtlantisAPUHYD, 73
Fuel_Press
PanelR2, 260
fValve1Fail
AtlantisAPUHYD, 73
fValve1Pulse
AtlantisAPUHYD, 73
fValve2Fail
AtlantisAPUHYD, 73
fValve2Pulse
AtlantisAPUHYD, 73
fVestPressure
eva_docking::ODS, 214

g
Atlantis, 66
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AC_resource.h, 301
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AirDataProbeSystem.cpp, 302
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AirDataProbeSystem.h, 303
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Atlantis.cpp, 304
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Atlantis.h, 311
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Atlantis_SRB.cpp, 331
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Atlantis_Tank.cpp, 332
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AtlantisAPUHYD.cpp, 334
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AtlantisAPUHYD.h, 335
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AtlantisConfig.cpp, 337
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AtlantisSubsystem.cpp, 339
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/AtlantisSubsystem.h, 340
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/BasicValve.cpp, 341
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/BasicValve.h, 342
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/CommModeHandler.cpp, 343
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/CommModeHandler.h, 344
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Common.cpp, 345
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/CRT.cpp, 348
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/CRT.h, 353
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscInPort.cpp, 355
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscInPort.h, 356
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscOutPort.cpp, 357
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscOutPort.h, 358
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscreteBundle.cpp, 359
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscreteBundle.h, 360
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscreteBundleManager.cpp, 361
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscreteBundleManager.h, 362
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscretePort.cpp, 363
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/DiscretePort.h, 364
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/discsignals/discsignals.h, 365
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/AP101S.cpp, 366

G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/AP101S.h, 367
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/BIU.h, 368
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/dps_defs.h, 369
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/GPCVariables.h, 370
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/IDP.cpp, 371
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/IDP.h, 372
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/MasterTimingUnit.cpp, 374
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/MasterTimingUnit.h, 375
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/MDM.cpp, 377
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/MDM.h, 378
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/RAM16.h, 379
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/ShuttleBus.cpp, 380
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/dps/ShuttleBus.h, 381
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/eps/eps_defs.h, 382
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/eva_docking/BasicExtAirlock.cpp, 383
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/eva_docking/BasicExtAirlock.h, 384
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/eva_docking/ExtAirlock.cpp, 385
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/eva_docking/ExtAirlock.h, 386
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/eva_docking/ODS.h, 387
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/FSSRSS/meshres_FSS_RSS.h, 388
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/FSSRSS/SSU_FSS_RSS.cpp, 395
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/FSSRSS/SSU_FSS_RSS.h, 397
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/GearOp.cpp, 398
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/GearOp.h, 399
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/gnc/AA.h, 400
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/GPC.cpp, 401
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/HALFC/HALFC.cpp, 403
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/HALFC/StdAfx.cpp, 404
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/HALFC/StdAfx.h, 405
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/hydraulics/HydActuator.h, 406
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/ISubsystem.h, 407
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Keyboard.cpp, 408
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/Keyboard.h, 409
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/KuBandSystem.cpp, 410
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/KuBandSystem.h, 411
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/MechActuator.h, 412
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/meshres.h, 413
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/meshres_ku.h, 417
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/meshres_odsvc.h, 418
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/meshres_RMS.h, 423
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/meshres_vc.h, 425
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/meshres_vc_additions.h, 443
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mission/Mission.h, 447
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/MLP/MLP.cpp, 448
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/MLP/MLP.h, 449
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/ATVC.cpp, 450
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/ATVC.h, 451
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/BLOCK_II.cpp, 452
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/BLOCK_II.h, 453
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/EIU.cpp, 454
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/EIU.h, 455

G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/MPS.cpp, 456
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/MPS.h, 457
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/SSME.cpp, 459
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/mps/SSME.h, 460
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/OMSPod.cpp, 462
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/OMSPod.h, 463
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/OMSSubsystem.cpp, 465
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/OMSSubsystem.h, 466
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelA4.cpp, 467
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelA4.h, 468
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelA8.cpp, 469
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelA8.h, 470
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelC2.cpp, 471
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelC2.h, 473
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelC3.cpp, 475
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelC3.h, 476
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelF7.cpp, 478
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelF7.h, 480
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelO3.cpp, 482
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelO3.h, 483
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelR2.cpp, 485
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PanelR2.h, 486
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PlBayOp.cpp, 487
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/PlBayOp.h, 489
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/resource.h, 490
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/resource_RMS.h, 494
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/SubsystemDirector.cpp, 495
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/SubsystemDirector.h, 496
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/util/Stopwatch.cpp, 497
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/util/Stopwatch.h, 498
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/ValveTypeBool.cpp, 499
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/ValveTypeBool.h, 500
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicLight.h, 501
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicPanel.cpp, 502
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicPanel.h, 503
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicSwitch.cpp, 504
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicSwitch.h, 505
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicVCComponent.cpp, 506
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/BasicVCComponent.h, 507
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/EventTimer.h, 508
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/Indicator7Seg.cpp, 509
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/Indicator7Seg.h, 510
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/MDU.cpp, 511
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/MDU.h, 512
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelA7A8ODS.cpp, 513
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelA7A8ODS.h, 514
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelF6.cpp, 515
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelF6.h, 516
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelF7.cpp, 479
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelF7.h, 481
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelF8.cpp, 517
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelF8.h, 518
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelGroup.cpp, 519
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/PanelGroup.h, 520

G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/StandardSwitch.cpp, 521
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/StandardSwitch.h, 522
G:/orbiter2006/Orbitersdk/Space Shuttle Ultra/vc/vc_defs.h, 524
g_hc
Atlantis.cpp, 309
GearOp.cpp, 398
PanelA4.cpp, 467
PanelA8.cpp, 469
PanelC2.cpp, 472
PanelC3.cpp, 475
PanelF7.cpp, 478
PanelO3.cpp, 482
PanelR2.cpp, 485
PlBayOp.cpp, 488
g_MFDmode
CRT.cpp, 352
g_Param
Atlantis.cpp, 309
GearOp.cpp, 398
MDU.cpp, 511
PanelA4.cpp, 467
PanelA8.cpp, 469
PanelC2.cpp, 472
PanelC3.cpp, 475
PanelF7.cpp, 478
PanelO3.cpp, 482
PanelR2.cpp, 485
PlBayOp.cpp, 488
GAS_CONSTANT
OMSPod.h, 464
GDIParams, 155
clock_digits, 155
deu_characters, 155
DeuCharBitmapDC, 155
digits_7seg, 155
font, 155
hDLL, 155
tkbk_label, 155
GEAR_MAX_DEPLOY_SPEED
Atlantis.h, 326
GEAR_OPERATING_SPEED
Atlantis.h, 326
gear_proc
GearOp, 160
gear_status
GearOp, 160
GearArmed
GearOp, 159
GearDamaged
GearOp, 160
GearOp, 156
anim_gear, 160
anim_VC_F6, 160
ArmGear, 159
Atlantis, 57, 160
DamageGear, 159
DefineAnimations, 158
DefineVCAnimations, 158
gear_proc, 160
gear_status, 160
GearArmed, 159
GearDamaged, 160
GearOp, 158
GetGearAction, 158
OpenDialog, 158
OperateLandingGear, 159
ParseScenarioLine, 159
RegisterVC, 158
RevertLandingGear, 159
SaveState, 159
SetGearParameters, 159
Step, 158
sts, 160
SW_NOT_PUSH, 157
SW_PUSH, 157
SWC_CLOSE, 157
SWC_OPEN, 157
SwitchCover, 160
SwitchPush, 160
tkbk_state, 160
UpdateMesh, 158
UpdateVC, 158
VCDrawTalkback, 160
VCMouseEvent, 158
VCRedrawEvent, 158
GearOp.cpp
ActionString, 398
g_hc, 398
g_Param, 398
GetActPos
hydraulics::HydActuator, 162
GetAmps
eps::IACElement, 163
eps::IDCElement, 164
GetAPUTurbineRPM
AtlantisAPUHYD, 73
GetAzimuth
KuBandSystem, 181
GetBundleByIndex
discsignals::DiscreteBundleManager, 143
GetDirection
vc::BasicVCComponent, 106
GetDoorAction
PayloadBayOp, 266
GetDrivingIDP

vc::MDU, 197
GetDWord
dps::RAM16, 270
GetDynamicPressure
AirDataProbeSystem, 33
GetEdgekeyMenu
vc::MDU, 197
GetEdgeKeyMenuLabel
vc::MDU, 195
GetETPropellant
Atlantis, 53
GetEventTimerMin
dps::MasterTimingUnit, 187
GetEventTimerSec
dps::MasterTimingUnit, 187
GetGasGeneratorTemperature
AtlantisAPUHYD, 72
GetGearAction
GearOp, 158
GetGMTDay
dps::MasterTimingUnit, 187
GetGMTHour
dps::MasterTimingUnit, 187
GetGMTMilli
dps::MasterTimingUnit, 187
GetGMTMin
dps::MasterTimingUnit, 187
GetGMTSec
dps::MasterTimingUnit, 187
GetGPCLVLHVel
Atlantis, 53
GetGPCMET
Atlantis, 53
GetGPCRefHDot
Atlantis, 53
GetHandle
mps::SSME, 282
GetIdentifier
AtlantisSubsystem, 82
ISubsystem, 176
vc::BasicPanel, 96
vc::BasicVCComponent, 108
GetIdentity
discsignals::DiscreteBundle, 141
GetIDP
Atlantis, 53
GetIDPID
dps::IDP, 168
GetLabel
vc::BasicSwitch, 98
GetLastCreatedMFD
Atlantis, 53
GetLiftOffFlag
Atlantis, 53
GetMDU
Atlantis, 53
GetMETDay
dps::MasterTimingUnit, 187
GetMETHour
dps::MasterTimingUnit, 187
GetMETMilli
dps::MasterTimingUnit, 187
GetMETMin
dps::MasterTimingUnit, 187
GetMETSec
dps::MasterTimingUnit, 187
GetMissionName
mission::Mission, 204
GetN2TankPressure
OMSPod, 218
GetOMSPressure
Atlantis, 53
GetOrbiter
mission::Mission, 204
GetOrbiterCoGOffset
Atlantis, 53
GetPortConfig
vc::MDU, 197
GetPos
BasicValve, 101
ValveTypeBool, 298
GetPower
eps::IDCElement, 164
GetPYR
Atlantis, 55
GetPYR2
Atlantis, 55
GetQualifiedIdentifier
AtlantisSubsystem, 81
ISubsystem, 176
vc::BasicPanel, 96
vc::BasicVCComponent, 107
GetRadiatorAction
PayloadBayOp, 266
GetReference
vc::BasicVCComponent, 106
GetRotMatrixX
Atlantis, 55
GetRotMatrixY
Atlantis, 55
GetRotMatrixZ
Atlantis, 55
GetSelectedPort
vc::MDU, 197
GetSignalStrength
KuBandSystem, 181
GetSRB_State
Atlantis.cpp, 307

Common.cpp, 345
GetSRBChamberPressure
Atlantis, 53
GetState
vc::BasicVCComponent, 105
GetStaticPressure
AirDataProbeSystem, 33
GetSubsystemEmptyMass
AtlantisSubsystem, 82
eva_docking::BasicExternalAirlock, 86
eva_docking::ODS, 214
GetTotalTemperature
AirDataProbeSystem, 33
GetVCMeshIndex
vc::BasicPanel, 94
vc::PanelA7A8ODS, 228
GetViewAngle
vc::MDU, 197
GetVoltage
discsignals::DiscInPort, 137
discsignals::DiscreteBundle, 140
eps::IACElement, 163
eps::IDCElement, 164
GetWord
dps::RAM16, 270
GimbalLockState
KuBandSystem, 181
GimbalOMS
Atlantis, 55
GNC
dps, 15
gnc, 18
gnc::AccAssembly, 29
~AccAssembly, 30
AccAssembly, 29
outY, 30
outZ, 30
ref, 30
vY, 30
vZ, 30
gnc_ops
dps::IDP, 168
gnc_specdisp
dps::IDP, 168
GNCSYSSUMM1
CRT, 128
gop
Atlantis, 60
gParams
AtlantisConfig.cpp, 338
GPC
Atlantis, 55
GPC.cpp
linterp, 401
listerp, 401
tableterp, 401
tpir, 402
GPCMEMORY
CRT, 126
gpcReadValue
Atlantis.cpp, 307
gpcSetValue
Atlantis.cpp, 307
GPCV_GMT_HIGH
GPCVariables.h, 370
GPCV_GMT_LOW
GPCVariables.h, 370
GPCV_MET_HIGH
GPCVariables.h, 370
GPCV_MET_LOW
GPCVariables.h, 370
GPCV_OP_CURRENT
GPCVariables.h, 370
GPCV_OP_NEXT
GPCVariables.h, 370
GPCVariables.h
GPCV_GMT_HIGH, 370
GPCV_GMT_LOW, 370
GPCV_MET_HIGH, 370
GPCV_MET_LOW, 370
GPCV_OP_CURRENT, 370
GPCV_OP_NEXT, 370
NUM_GPC_VARIABLES, 370
Grapple
Atlantis, 66
GRAY
CRT.cpp, 350
GREEN
CRT.cpp, 350
GRP_080_VC
meshres_vc.h, 442
GRP_081_VC
meshres_vc.h, 442
GRP_082_VC
meshres_vc.h, 442
GRP_146_VC
meshres_vc.h, 442
GRP_147_VC
meshres_vc.h, 442
GRP_3_PRPLT_LEFT1_VC
meshres_vc.h, 442
GRP_A1_VC
meshres_vc.h, 442
GRP_A1Guard_VC
meshres_vc.h, 442
GRP_A1Lpanel_VC
meshres_vc.h, 442
GRP_A1LRS6_VC

meshres_vc.h, 442
GRP_A1LS11_VC
meshres_vc.h, 442
GRP_A1LS12_VC
meshres_vc.h, 442
GRP_A1LS13_VC
meshres_vc.h, 442
GRP_A1LS14_VC
meshres_vc.h, 442
GRP_A1LS15_VC
meshres_vc.h, 442
GRP_A1LS16_VC
meshres_vc.h, 442
GRP_A1LS17_VC
meshres_vc.h, 442
GRP_A1LS18_VC
meshres_vc.h, 442
GRP_A1LS19_VC
meshres_vc.h, 442
GRP_A1LS1_VC
meshres_vc.h, 442
GRP_A1LS20_VC
meshres_vc.h, 442
GRP_A1LS21_VC
meshres_vc.h, 442
GRP_A1LS22_VC
meshres_vc.h, 442
GRP_A1LS23_VC
meshres_vc.h, 442
GRP_A1LS24_VC
meshres_vc.h, 442
GRP_A1LS25_VC
meshres_vc.h, 442
GRP_A1LS26_VC
meshres_vc.h, 442
GRP_A1LS2_VC
meshres_vc.h, 442
GRP_A1LS4_VC
meshres_vc.h, 442
GRP_A1LS5_VC
meshres_vc.h, 442
GRP_A1LS8_VC
meshres_vc.h, 442
GRP_A1LS9_VC
meshres_vc.h, 442
GRP_A1R_VC
meshres_vc.h, 442
GRP_A1RRS10_VC
meshres_vc.h, 442
GRP_A1RRS14_VC
meshres_vc.h, 442
GRP_A1RRS15_VC
meshres_vc.h, 442
GRP_A1RRS16_VC
meshres_vc.h, 442
GRP_A1RRS4_VC
meshres_vc.h, 442
GRP_A1RRS8_VC
meshres_vc.h, 442
GRP_A1RS11_VC
meshres_vc.h, 442
GRP_A1RS12_VC
meshres_vc.h, 442
GRP_A1RS13_VC
meshres_vc.h, 442
GRP_A1RS17_VC
meshres_vc.h, 442
GRP_A1RS18_VC
meshres_vc.h, 442
GRP_A1RS19_VC
meshres_vc.h, 442
GRP_A1RS1_VC
meshres_vc.h, 442
GRP_A1RS20_VC
meshres_vc.h, 442
GRP_A1RS21_VC
meshres_vc.h, 442
GRP_A1RS22_VC
meshres_vc.h, 442
GRP_A1RS23_VC
meshres_vc.h, 442
GRP_A1RS24_VC
meshres_vc.h, 442
GRP_A1RS25_VC
meshres_vc.h, 442
GRP_A1RS26_VC
meshres_vc.h, 442
GRP_A1RS27_VC
meshres_vc.h, 442
GRP_A1RS28_VC
meshres_vc.h, 442
GRP_A1RS29_VC
meshres_vc.h, 442
GRP_A1RS2_VC
meshres_vc.h, 442
GRP_A1RS30_VC
meshres_vc.h, 442
GRP_A1RS31_VC
meshres_vc.h, 442
GRP_A1RS32_VC
meshres_vc.h, 442
GRP_A1RS33_VC
meshres_vc.h, 442
GRP_A1RS34_VC
meshres_vc.h, 442
GRP_A1RS35_VC
meshres_vc.h, 442
GRP_A1RS3_VC

meshres_vc.h, 442
GRP_A1RS5_VC
meshres_vc.h, 442
GRP_A1RS6_VC
meshres_vc.h, 442
GRP_A1RS7_VC
meshres_vc.h, 442
GRP_A1RS9_VC
meshres_vc.h, 442
GRP_A1U_VC
meshres_vc.h, 442
GRP_A1UM1_VC
meshres_vc.h, 442
GRP_A1URS15_VC
meshres_vc.h, 442
GRP_A1URS16_VC
meshres_vc.h, 442
GRP_A1URS7_VC
meshres_vc.h, 442
GRP_A1US10_VC
meshres_vc.h, 442
GRP_A1US12_VC
meshres_vc.h, 442
GRP_A1US13_VC
meshres_vc.h, 442
GRP_A1US14_VC
meshres_vc.h, 442
GRP_A1US18_VC
meshres_vc.h, 442
GRP_A1US1_VC
meshres_vc.h, 442
GRP_A1US2_VC
meshres_vc.h, 442
GRP_A1US3_VC
meshres_vc.h, 442
GRP_A1US4_VC
meshres_vc.h, 442
GRP_A1US6_VC
meshres_vc.h, 442
GRP_A1US8_VC
meshres_vc.h, 442
GRP_A2AZH_VC
meshres_vc.h, 442
GRP_A2AZV_VC
meshres_vc.h, 442
GRP_A2ELH_VC
meshres_vc.h, 442
GRP_A2ELV_VC
meshres_vc.h, 442
GRP_A2panel_VC
meshres_vc.h, 442
GRP_A2U1_VC
meshres_vc.h, 442
GRP_A2U2_VC
meshres_vc.h, 442
GRP_A4_Event_M10_VC
meshres_vc.h, 442
meshres_vc_additions.h, 446
GRP_A4_Event_M_VC
meshres_vc.h, 442
meshres_vc_additions.h, 446
GRP_A4_Event_S10_VC
meshres_vc.h, 442
meshres_vc_additions.h, 446
GRP_A4_Event_S_VC
meshres_vc.h, 442
meshres_vc_additions.h, 446
GRP_A4_MET_DAY1_VC
meshres_vc.h, 442
meshres_vc_additions.h, 446
GRP_A4_MET_DAY2_VC
meshres_vc.h, 442
meshres_vc_additions.h, 446
GRP_A4_MET_DAY3_VC
meshres_vc.h, 442
meshres_vc_additions.h, 446
GRP_A4_MET_H1_VC
meshres_vc.h, 442
meshres_vc_additions.h, 446
GRP_A4_MET_H2_VC
meshres_vc.h, 442
meshres_vc_additions.h, 446
GRP_A4_MET_M1_VC
meshres_vc.h, 442
meshres_vc_additions.h, 446
GRP_A4_MET_M2_VC
meshres_vc.h, 442
meshres_vc_additions.h, 446
GRP_A4_MET_S1_VC
meshres_vc.h, 442
meshres_vc_additions.h, 446
GRP_A4_MET_S2_VC
meshres_vc.h, 442
meshres_vc_additions.h, 446
GRP_A4MissionSwitch_VC
meshres_vc.h, 442
meshres_vc_additions.h, 446
GRP_A6U1_VC
meshres_vc.h, 442
GRP_A6U2_VC
meshres_vc.h, 442
GRP_A6U32_VC
meshres_vc.h, 442
GRP_A6U33_VC
meshres_vc.h, 442
GRP_A6U34_VC
meshres_vc.h, 442
GRP_A6U35_VC

meshres_vc.h, 442
GRP_A6U36_VC
meshres_vc.h, 442
GRP_A6U39_VC
meshres_vc.h, 442
GRP_A6U40_VC
meshres_vc.h, 442
GRP_A6U41_VC
meshres_vc.h, 442
GRP_A6U42_VC
meshres_vc.h, 442
GRP_A6U43_VC
meshres_vc.h, 442
GRP_A6U44_VC
meshres_vc.h, 442
GRP_A6U45_VC
meshres_vc.h, 442
GRP_A6U46_VC
meshres_vc.h, 442
GRP_A6U47_VC
meshres_vc.h, 442
GRP_A6U5_VC
meshres_vc.h, 442
GRP_A6U6_VC
meshres_vc.h, 442
GRP_A6U7_VC
meshres_vc.h, 442
GRP_A6Upanel_VC
meshres_vc.h, 442
GRP_A6URS1_VC
meshres_vc.h, 442
GRP_A6URS2_VC
meshres_vc.h, 442
GRP_A6URS3_VC
meshres_vc.h, 442
GRP_A6URS4_VC
meshres_vc.h, 442
GRP_A6URS5_VC
meshres_vc.h, 442
GRP_A6US37_VC
meshres_vc.h, 442
GRP_A6US4_VC
meshres_vc.h, 442
GRP_A6Utog1_VC
meshres_vc.h, 442
GRP_A6Utog2_VC
meshres_vc.h, 442
GRP_A6Utog3_VC
meshres_vc.h, 442
GRP_A6Utog4_VC
meshres_vc.h, 442
GRP_A7A3CB10_ODSVC
meshres_odsvc.h, 422
GRP_A7A3CB11_ODSVC
meshres_odsvc.h, 422
GRP_A7A3CB12_ODSVC
meshres_odsvc.h, 422
GRP_A7A3CB13_ODSVC
meshres_odsvc.h, 422
GRP_A7A3CB14_ODSVC
meshres_odsvc.h, 422
GRP_A7A3CB15_ODSVC
meshres_odsvc.h, 422
GRP_A7A3CB16_ODSVC
meshres_odsvc.h, 422
GRP_A7A3CB1_ODSVC
meshres_odsvc.h, 422
GRP_A7A3CB2_ODSVC
meshres_odsvc.h, 422
GRP_A7A3CB3_ODSVC
meshres_odsvc.h, 422
GRP_A7A3CB4_ODSVC
meshres_odsvc.h, 422
GRP_A7A3CB5_ODSVC
meshres_odsvc.h, 422
GRP_A7A3CB6_ODSVC
meshres_odsvc.h, 422
GRP_A7A3CB7_ODSVC
meshres_odsvc.h, 422
GRP_A7A3CB8_ODSVC
meshres_odsvc.h, 422
GRP_A7A3CB9_ODSVC
meshres_odsvc.h, 422
GRP_A7A3Guard_ODSVC
meshres_odsvc.h, 422
GRP_A7A3panel_ODSVC
meshres_odsvc.h, 422
GRP_A7A3S10_ODSVC
meshres_odsvc.h, 422
GRP_A7A3S11_ODSVC
meshres_odsvc.h, 422
GRP_A7A3S12_ODSVC
meshres_odsvc.h, 422
GRP_A7A3S13_ODSVC
meshres_odsvc.h, 422
GRP_A7A3S14_ODSVC
meshres_odsvc.h, 422
GRP_A7A3S15_ODSVC
meshres_odsvc.h, 422
GRP_A7A3S16_ODSVC
meshres_odsvc.h, 422
GRP_A7A3S1_ODSVC
meshres_odsvc.h, 422
GRP_A7A3S2_ODSVC
meshres_odsvc.h, 422
GRP_A7A3S3_ODSVC
meshres_odsvc.h, 422
GRP_A7A3S4_ODSVC

meshres_odsvc.h, 422
GRP_A7A3S5_ODSVC
meshres_odsvc.h, 422
GRP_A7A3S6_ODSVC
meshres_odsvc.h, 422
GRP_A7A3S7_ODSVC
meshres_odsvc.h, 422
GRP_A7A3S8_ODSVC
meshres_odsvc.h, 422
GRP_A7A3S9_ODSVC
meshres_odsvc.h, 422
GRP_A7U10_VC
meshres_vc.h, 442
GRP_A7U11_VC
meshres_vc.h, 442
GRP_A7U12_VC
meshres_vc.h, 442
GRP_A7U13_VC
meshres_vc.h, 442
GRP_A7U14_VC
meshres_vc.h, 442
GRP_A7U15_VC
meshres_vc.h, 442
GRP_A7U16_VC
meshres_vc.h, 442
GRP_A7U17_VC
meshres_vc.h, 442
GRP_A7U18_VC
meshres_vc.h, 442
GRP_A7U1_VC
meshres_vc.h, 442
GRP_A7U2_VC
meshres_vc.h, 442
GRP_A7U3_VC
meshres_vc.h, 442
GRP_A7U40_VC
meshres_vc.h, 442
GRP_A7U41_VC
meshres_vc.h, 442
GRP_A7U42_VC
meshres_vc.h, 442
GRP_A7U43_VC
meshres_vc.h, 442
GRP_A7U44_VC
meshres_vc.h, 442
GRP_A7U4_VC
meshres_vc.h, 442
GRP_A7U52_VC
meshres_vc.h, 442
GRP_A7U53_VC
meshres_vc.h, 442
GRP_A7U54_VC
meshres_vc.h, 442
GRP_A7U56_VC
meshres_vc.h, 442
GRP_A7U5_VC
meshres_vc.h, 442
GRP_A7U6_VC
meshres_vc.h, 442
GRP_A7U7_VC
meshres_vc.h, 442
GRP_A7U8_VC
meshres_vc.h, 442
GRP_A7U9_VC
meshres_vc.h, 442
GRP_A8A3panel_ODSVC
meshres_odsvc.h, 422
GRP_A8A3S10_ODSVC
meshres_odsvc.h, 422
GRP_A8A3S11_ODSVC
meshres_odsvc.h, 422
GRP_A8A3S12_ODSVC
meshres_odsvc.h, 422
GRP_A8A3S1_ODSVC
meshres_odsvc.h, 422
GRP_A8A3S2_ODSVC
meshres_odsvc.h, 422
GRP_A8A3S3_ODSVC
meshres_odsvc.h, 422
GRP_A8A3S4_ODSVC
meshres_odsvc.h, 422
GRP_A8A3S5_ODSVC
meshres_odsvc.h, 422
GRP_A8A3S6_ODSVC
meshres_odsvc.h, 422
GRP_A8A3S7_ODSVC
meshres_odsvc.h, 422
GRP_A8A3S8_ODSVC
meshres_odsvc.h, 422
GRP_A8A3S9_ODSVC
meshres_odsvc.h, 422
GRP_A8b10_VC
meshres_vc_additions.h, 446
GRP_A8b11_VC
meshres_vc_additions.h, 446
GRP_A8b12_VC
meshres_vc_additions.h, 446
GRP_A8b13_VC
meshres_vc_additions.h, 446
GRP_A8b14_VC
meshres_vc_additions.h, 446
GRP_A8b15_VC
meshres_vc_additions.h, 446
GRP_A8b16_VC
meshres_vc_additions.h, 446
GRP_A8b17_VC
meshres_vc_additions.h, 446
GRP_A8b1_VC

meshres_vc_additions.h, 446
GRP_A8b2_VC
meshres_vc_additions.h, 446
GRP_A8b3_VC
meshres_vc_additions.h, 446
GRP_A8b4_VC
meshres_vc_additions.h, 446
GRP_A8b5_VC
meshres_vc_additions.h, 446
GRP_A8b6_VC
meshres_vc_additions.h, 446
GRP_A8b7_VC
meshres_vc_additions.h, 446
GRP_A8b8_VC
meshres_vc_additions.h, 446
GRP_A8b9_VC
meshres_vc_additions.h, 446
GRP_A8bot_VC
meshres_vc.h, 442
GRP_A8Guardb_VC
meshres_vc.h, 442
GRP_A8top_VC
meshres_vc.h, 442
GRP_ACK_VC
meshres_vc.h, 442
GRP_Act_Hooks_Firing_ODSVC
meshres_odsvc.h, 422
GRP_ADS_ODSVC
meshres_odsvc.h, 422
GRP_AFT_HOOKS
meshres.h, 416
GRP_aft_VC
meshres_vc.h, 442
GRP_AftMonitor_VC
meshres_vc.h, 442
GRP_AILERONL
meshres.h, 416
GRP_AILERONR
meshres.h, 416
GRP_APDS_Circ_Prot_OFF_ODSVC
meshres_odsvc.h, 422
GRP_APDSCC_APDS_Circ_Prot_OFF_ODSVC
meshres_odsvc.h, 422
GRP_APDSCC_CloseHooks_ODSVC
meshres_odsvc.h, 422
GRP_APDSCC_CloseLatches_ODSVC
meshres_odsvc.h, 422
GRP_APDSCC_FixerOff_ODSVC
meshres_odsvc.h, 422
GRP_APDSCC_PowerOff_ODSVC
meshres_odsvc.h, 422
GRP_APDSCC_PowerOn_ODSVC
meshres_odsvc.h, 422
GRP_APDSCC_RingIn_ODSVC
meshres_odsvc.h, 422
GRP_APDSCC_RingOut_ODSVC
meshres_odsvc.h, 422
GRP_B_VC
meshres_vc.h, 442
GRP_base
meshres_RMS.h, 424
GRP_base_KU
meshres_ku.h, 417
GRP_BDS_ODSVC
meshres_odsvc.h, 422
GRP_BODYFLAP
meshres.h, 416
GRP_bottomcup_VC
meshres_vc.h, 442
GRP_bottomdial_VC
meshres_vc.h, 442
GRP_Box01
meshres_FSS_RSS.h, 394
GRP_Box02
meshres_FSS_RSS.h, 394
GRP_Box03
meshres_FSS_RSS.h, 394
GRP_Box04
meshres_FSS_RSS.h, 394
GRP_Box05
meshres_FSS_RSS.h, 394
GRP_Box06
meshres_FSS_RSS.h, 394
GRP_Box07
meshres_FSS_RSS.h, 394
GRP_Box08
meshres_FSS_RSS.h, 394
GRP_Box09
meshres_FSS_RSS.h, 394
GRP_Box10
meshres_FSS_RSS.h, 394
GRP_Box11
meshres_FSS_RSS.h, 394
GRP_Box12
meshres_FSS_RSS.h, 394
GRP_Box13
meshres_FSS_RSS.h, 394
GRP_Box14
meshres_FSS_RSS.h, 394
GRP_Box15
meshres_FSS_RSS.h, 394
GRP_Box16
meshres_FSS_RSS.h, 394
GRP_Box17
meshres_FSS_RSS.h, 394
GRP_Box18
meshres_FSS_RSS.h, 394
GRP_Box19

meshres_FSS_RSS.h, 394
GRP_Box20
meshres_FSS_RSS.h, 394
GRP_Box21
meshres_FSS_RSS.h, 394
GRP_Box22
meshres_FSS_RSS.h, 394
GRP_Box23
meshres_FSS_RSS.h, 394
GRP_Box24
meshres_FSS_RSS.h, 394
GRP_Box25
meshres_FSS_RSS.h, 394
GRP_Box26
meshres_FSS_RSS.h, 394
GRP_Box27
meshres_FSS_RSS.h, 394
GRP_Box28
meshres_FSS_RSS.h, 394
GRP_Box29
meshres_FSS_RSS.h, 394
GRP_Box30
meshres_FSS_RSS.h, 394
GRP_Box31
meshres_FSS_RSS.h, 394
GRP_Box32
meshres_FSS_RSS.h, 394
GRP_Box33
meshres_FSS_RSS.h, 394
GRP_Box34
meshres_FSS_RSS.h, 394
GRP_Box35
meshres_FSS_RSS.h, 394
GRP_Box36
meshres_FSS_RSS.h, 394
GRP_Box37
meshres_FSS_RSS.h, 394
GRP_C2_VC
meshres_vc.h, 442
GRP_C2b10_VC
meshres_vc.h, 442
meshres_vc_additions.h, 446
GRP_C2b11_VC
meshres_vc.h, 442
GRP_C2b12_VC
meshres_vc_additions.h, 446
GRP_C2b1_VC
meshres_vc.h, 442
meshres_vc_additions.h, 446
GRP_C2b2_VC
meshres_vc.h, 442
meshres_vc_additions.h, 446
GRP_C2b3_VC
meshres_vc.h, 442
meshres_vc_additions.h, 446
GRP_C2b4_VC
meshres_vc.h, 442
meshres_vc_additions.h, 446
GRP_C2b5_VC
meshres_vc.h, 442
meshres_vc_additions.h, 446
GRP_C2b6_VC
meshres_vc.h, 442
meshres_vc_additions.h, 446
GRP_C2b7_VC
meshres_vc_additions.h, 446
GRP_C2b7v_VC
meshres_vc.h, 442
GRP_C2b8_VC
meshres_vc_additions.h, 446
GRP_C2b8v_VC
meshres_vc.h, 442
GRP_C2b9_VC
meshres_vc.h, 442
meshres_vc_additions.h, 446
GRP_C2Evt10MW_VC
meshres_vc_additions.h, 446
GRP_C2Evt10SW_VC
meshres_vc_additions.h, 446
GRP_C2Evt1MW_VC
meshres_vc_additions.h, 446
GRP_C2Evt1SW_VC
meshres_vc_additions.h, 446
GRP_C2guards_VC
meshres_vc.h, 442
GRP_C2tog1_VC
meshres_vc.h, 442
GRP_C2tog2_VC
meshres_vc.h, 442
GRP_C2tog3_VC
meshres_vc.h, 442
GRP_C2tog4_VC
meshres_vc.h, 442
GRP_C3_VC
meshres_vc.h, 442
GRP_C3b10_VC
meshres_vc.h, 442
GRP_C3b11_VC
meshres_vc.h, 442
GRP_C3b12_VC
meshres_vc.h, 442
GRP_C3b13_VC
meshres_vc.h, 442
GRP_C3b14v_VC
meshres_vc.h, 442
GRP_C3b15_VC
meshres_vc.h, 442
GRP_C3b16_VC

meshres_vc.h, 442
GRP_C3b17_VC
meshres_vc.h, 442
GRP_C3b18v_VC
meshres_vc.h, 442
GRP_C3b19_VC
meshres_vc.h, 442
GRP_C3b1_VC
meshres_vc.h, 442
GRP_C3b20_VC
meshres_vc.h, 442
GRP_C3b21_VC
meshres_vc.h, 442
GRP_C3b22_VC
meshres_vc.h, 442
GRP_C3b23_VC
meshres_vc.h, 442
GRP_C3b24_VC
meshres_vc.h, 442
GRP_C3b25_VC
meshres_vc.h, 442
GRP_C3b26_VC
meshres_vc.h, 442
GRP_C3b27_VC
meshres_vc.h, 442
GRP_C3b28_VC
meshres_vc.h, 442
GRP_C3b29_VC
meshres_vc.h, 442
GRP_C3b2_VC
meshres_vc.h, 442
GRP_C3b30_VC
meshres_vc.h, 442
GRP_C3b31_VC
meshres_vc.h, 442
GRP_C3b32_VC
meshres_vc.h, 442
GRP_C3b33_VC
meshres_vc.h, 442
GRP_C3b34_VC
meshres_vc.h, 442
GRP_C3b35_VC
meshres_vc.h, 442
GRP_C3b36_VC
meshres_vc.h, 442
GRP_C3b37_VC
meshres_vc.h, 442
GRP_C3b3_VC
meshres_vc.h, 442
GRP_C3b4_VC
meshres_vc.h, 442
GRP_C3b5_VC
meshres_vc.h, 442
GRP_C3b6_VC
meshres_vc.h, 442
GRP_C3b7_VC
meshres_vc.h, 442
GRP_C3b8_VC
meshres_vc.h, 442
GRP_C3b9_VC
meshres_vc.h, 442
GRP_C3cover1_VC
meshres_vc.h, 442
GRP_C3cover2_VC
meshres_vc.h, 442
GRP_C3cover3_VC
meshres_vc.h, 442
GRP_C3cover4_VC
meshres_vc.h, 442
GRP_C3cover5_VC
meshres_vc.h, 442
GRP_C3Dial_VC
meshres_vc.h, 442
GRP_C_VC
meshres_vc.h, 442
GRP_cable_one_KU
meshres_ku.h, 417
GRP_cable_VC
meshres_vc.h, 442
GRP_cambase
meshres_RMS.h, 424
GRP_CAMERABL
meshres.h, 416
GRP_CAMERABR
meshres.h, 416
GRP_CAMERAFL
meshres.h, 416
GRP_CAMERAFR
meshres.h, 416
GRP_CAMERAPIVOTBL
meshres.h, 416
GRP_CAMERAPIVOTBR
meshres.h, 416
GRP_CAMERAPIVOTFL
meshres.h, 416
GRP_CAMERAPIVOTFR
meshres.h, 416
GRP_camswivel
meshres_RMS.h, 424
GRP_CAPTURE_ODSVC
meshres_odsvc.h, 422
GRP_CARGOBAY
meshres.h, 416
GRP_CARGODOORINL
meshres.h, 416
GRP_CARGODOORINR
meshres.h, 416
GRP_CARGODOOROUTL

meshres.h, 416
GRP_CARGODOOROUTR
meshres.h, 416
GRP_CDR1_VC
meshres_vc.h, 442
GRP_CDR1LABEL_VC
meshres_vc.h, 442
GRP_CDR2_VC
meshres_vc.h, 442
GRP_CDR2LABEL_VC
meshres_vc.h, 442
GRP_CDS_ODSVC
meshres_odsvc.h, 422
GRP_clear_VC
meshres_vc.h, 442
GRP_Clock1_VC
meshres_vc.h, 442
GRP_Clock2_VC
meshres_vc.h, 442
GRP_Clock3_VC
meshres_vc.h, 442
GRP_Clock4_VC
meshres_vc.h, 442
GRP_Clock5_VC
meshres_vc.h, 442
GRP_Clock6_VC
meshres_vc.h, 442
GRP_Clock7_VC
meshres_vc.h, 442
GRP_Clock8_VC
meshres_vc.h, 442
GRP_Coas_VC
meshres_vc.h, 442
GRP_covers_VC
meshres_vc.h, 442
GRP_Cylinder01
meshres_FSS_RSS.h, 394
GRP_Cylinder02
meshres_FSS_RSS.h, 394
GRP_Cylinder03
meshres_FSS_RSS.h, 394
GRP_Cylinder04
meshres_FSS_RSS.h, 394
GRP_Cylinder05
meshres_FSS_RSS.h, 394
GRP_Cylinder06
meshres_FSS_RSS.h, 394
GRP_Cylinder07
meshres_FSS_RSS.h, 394
GRP_Cylinder08
meshres_FSS_RSS.h, 394
GRP_Cylinder09
meshres_FSS_RSS.h, 394
GRP_Cylinder10
meshres_FSS_RSS.h, 394
GRP_Cylinder100
meshres_FSS_RSS.h, 394
GRP_Cylinder101
meshres_FSS_RSS.h, 394
GRP_Cylinder102
meshres_FSS_RSS.h, 394
GRP_Cylinder103
meshres_FSS_RSS.h, 394
GRP_Cylinder104
meshres_FSS_RSS.h, 394
GRP_Cylinder105
meshres_FSS_RSS.h, 394
GRP_Cylinder106
meshres_FSS_RSS.h, 394
GRP_Cylinder107
meshres_FSS_RSS.h, 394
GRP_Cylinder108
meshres_FSS_RSS.h, 394
GRP_Cylinder109
meshres_FSS_RSS.h, 394
GRP_Cylinder11
meshres_FSS_RSS.h, 394
GRP_Cylinder110
meshres_FSS_RSS.h, 394
GRP_Cylinder111
meshres_FSS_RSS.h, 394
GRP_Cylinder112
meshres_FSS_RSS.h, 394
GRP_Cylinder113
meshres_FSS_RSS.h, 394
GRP_Cylinder114
meshres_FSS_RSS.h, 394
GRP_Cylinder115
meshres_FSS_RSS.h, 394
GRP_Cylinder116
meshres_FSS_RSS.h, 394
GRP_Cylinder117
meshres_FSS_RSS.h, 394
GRP_Cylinder118
meshres_FSS_RSS.h, 394
GRP_Cylinder119
meshres_FSS_RSS.h, 394
GRP_Cylinder12
meshres_FSS_RSS.h, 394
GRP_Cylinder120
meshres_FSS_RSS.h, 394
GRP_Cylinder121
meshres_FSS_RSS.h, 394
GRP_Cylinder122
meshres_FSS_RSS.h, 394
GRP_Cylinder123
meshres_FSS_RSS.h, 394
GRP_Cylinder13

meshres_FSS_RSS.h, 394
GRP_Cylinder14
meshres_FSS_RSS.h, 394
GRP_Cylinder15
meshres_FSS_RSS.h, 394
GRP_Cylinder16
meshres_FSS_RSS.h, 394
GRP_Cylinder17
meshres_FSS_RSS.h, 394
GRP_Cylinder18
meshres_FSS_RSS.h, 394
GRP_Cylinder19
meshres_FSS_RSS.h, 394
GRP_Cylinder20
meshres_FSS_RSS.h, 394
GRP_Cylinder21
meshres_FSS_RSS.h, 394
GRP_Cylinder22
meshres_FSS_RSS.h, 394
GRP_Cylinder23
meshres_FSS_RSS.h, 394
GRP_Cylinder24
meshres_FSS_RSS.h, 394
GRP_Cylinder25
meshres_FSS_RSS.h, 394
GRP_Cylinder26
meshres_FSS_RSS.h, 394
GRP_Cylinder27
meshres_FSS_RSS.h, 394
GRP_Cylinder28
meshres_FSS_RSS.h, 394
GRP_Cylinder29
meshres_FSS_RSS.h, 394
GRP_Cylinder30
meshres_FSS_RSS.h, 394
GRP_Cylinder31
meshres_FSS_RSS.h, 394
GRP_Cylinder32
meshres_FSS_RSS.h, 394
GRP_Cylinder33
meshres_FSS_RSS.h, 394
GRP_Cylinder34
meshres_FSS_RSS.h, 394
GRP_Cylinder35
meshres_FSS_RSS.h, 394
GRP_Cylinder36
meshres_FSS_RSS.h, 394
GRP_Cylinder37
meshres_FSS_RSS.h, 394
GRP_Cylinder38
meshres_FSS_RSS.h, 394
GRP_Cylinder39
meshres_FSS_RSS.h, 394
GRP_Cylinder40
meshres_FSS_RSS.h, 394
GRP_Cylinder41
meshres_FSS_RSS.h, 394
GRP_Cylinder42
meshres_FSS_RSS.h, 394
GRP_Cylinder43
meshres_FSS_RSS.h, 394
GRP_Cylinder44
meshres_FSS_RSS.h, 394
GRP_Cylinder45
meshres_FSS_RSS.h, 394
GRP_Cylinder46
meshres_FSS_RSS.h, 394
GRP_Cylinder47
meshres_FSS_RSS.h, 394
GRP_Cylinder48
meshres_FSS_RSS.h, 394
GRP_Cylinder49
meshres_FSS_RSS.h, 394
GRP_Cylinder50
meshres_FSS_RSS.h, 394
GRP_Cylinder51
meshres_FSS_RSS.h, 394
GRP_Cylinder52
meshres_FSS_RSS.h, 394
GRP_Cylinder53
meshres_FSS_RSS.h, 394
GRP_Cylinder54
meshres_FSS_RSS.h, 394
GRP_Cylinder55
meshres_FSS_RSS.h, 394
GRP_Cylinder56
meshres_FSS_RSS.h, 394
GRP_Cylinder57
meshres_FSS_RSS.h, 394
GRP_Cylinder58
meshres_FSS_RSS.h, 394
GRP_Cylinder59
meshres_FSS_RSS.h, 394
GRP_Cylinder60
meshres_FSS_RSS.h, 394
GRP_Cylinder61
meshres_FSS_RSS.h, 394
GRP_Cylinder62
meshres_FSS_RSS.h, 394
GRP_Cylinder63
meshres_FSS_RSS.h, 394
GRP_Cylinder64
meshres_FSS_RSS.h, 394
GRP_Cylinder65
meshres_FSS_RSS.h, 394
GRP_Cylinder66
meshres_FSS_RSS.h, 394
GRP_Cylinder67

meshres_FSS_RSS.h, 394
GRP_Cylinder68
meshres_FSS_RSS.h, 394
GRP_Cylinder69
meshres_FSS_RSS.h, 394
GRP_Cylinder70
meshres_FSS_RSS.h, 394
GRP_Cylinder71
meshres_FSS_RSS.h, 394
GRP_Cylinder72
meshres_FSS_RSS.h, 394
GRP_Cylinder73
meshres_FSS_RSS.h, 394
GRP_Cylinder74
meshres_FSS_RSS.h, 394
GRP_Cylinder75
meshres_FSS_RSS.h, 394
GRP_Cylinder76
meshres_FSS_RSS.h, 394
GRP_Cylinder77
meshres_FSS_RSS.h, 394
GRP_Cylinder78
meshres_FSS_RSS.h, 394
GRP_Cylinder79
meshres_FSS_RSS.h, 394
GRP_Cylinder80
meshres_FSS_RSS.h, 394
GRP_Cylinder81
meshres_FSS_RSS.h, 394
GRP_Cylinder82
meshres_FSS_RSS.h, 394
GRP_Cylinder83
meshres_FSS_RSS.h, 394
GRP_Cylinder84
meshres_FSS_RSS.h, 394
GRP_Cylinder85
meshres_FSS_RSS.h, 394
GRP_Cylinder86
meshres_FSS_RSS.h, 394
GRP_Cylinder87
meshres_FSS_RSS.h, 394
GRP_Cylinder88
meshres_FSS_RSS.h, 394
GRP_Cylinder89
meshres_FSS_RSS.h, 394
GRP_Cylinder90
meshres_FSS_RSS.h, 394
GRP_Cylinder91
meshres_FSS_RSS.h, 394
GRP_Cylinder92
meshres_FSS_RSS.h, 394
GRP_Cylinder93
meshres_FSS_RSS.h, 394
GRP_Cylinder94
meshres_FSS_RSS.h, 394
GRP_Cylinder95
meshres_FSS_RSS.h, 394
GRP_Cylinder96
meshres_FSS_RSS.h, 394
GRP_Cylinder97
meshres_FSS_RSS.h, 394
GRP_Cylinder98
meshres_FSS_RSS.h, 394
GRP_Cylinder99
meshres_FSS_RSS.h, 394
GRP_D_VC
meshres_vc.h, 442
GRP_Dial_VC
meshres_vc.h, 442
GRP_Dialsmall_VC
meshres_vc.h, 442
GRP_E_VC
meshres_vc.h, 442
GRP_eight_VC
meshres_vc.h, 442
GRP_elbowcam
meshres_RMS.h, 424
GRP_Endeffector
Atlantis.cpp, 306
GRP_ETUMBDOORL
meshres.h, 416
GRP_ETUMBDOORR
meshres.h, 416
GRP_Eventtimer_VC
meshres_vc.h, 442
meshres_vc_additions.h, 446
GRP_F3S1_VC
meshres_vc.h, 442
GRP_F3S2_VC
meshres_vc.h, 442
GRP_F3S3_VC
meshres_vc.h, 442
GRP_F3S4_VC
meshres_vc.h, 442
GRP_F3S5_VC
meshres_vc.h, 442
GRP_F3S6_VC
meshres_vc.h, 442
GRP_F6button1_VC
meshres_vc.h, 442
GRP_F6button2_VC
meshres_vc.h, 442
GRP_F6button3_VC
meshres_vc.h, 442
GRP_F6button4_VC
meshres_vc.h, 442
GRP_F6button5_VC
meshres_vc.h, 442

GRP_F6button6_VC
meshres_vc.h, 442
GRP_F6button7_VC
meshres_vc.h, 442
GRP_F6cover1_VC
meshres_vc.h, 442
GRP_F6cover2_VC
meshres_vc.h, 442
GRP_F6Dial_VC
meshres_vc.h, 442
GRP_F6Guard_VC
meshres_vc.h, 442
GRP_F6HUDbrt_VC
meshres_vc.h, 442
GRP_F6HUDdim_VC
meshres_vc.h, 442
GRP_F6HUDtest_VC
meshres_vc.h, 442
GRP_F6panel1_VC
meshres_vc.h, 442
GRP_F6panel_VC
meshres_vc.h, 442
GRP_F6switch_1_VC
meshres_vc.h, 442
GRP_F6switch_2_VC
meshres_vc.h, 442
GRP_F6switch_3_VC
meshres_vc.h, 442
GRP_F6switch_4_VC
meshres_vc.h, 442
GRP_F6switch_5_VC
meshres_vc.h, 442
GRP_F6switch_6_VC
meshres_vc.h, 442
GRP_F6switch_7_VC
meshres_vc.h, 442
GRP_F6switch_8_VC
meshres_vc.h, 442
GRP_F6switch_9_VC
meshres_vc.h, 442
GRP_F6TalkBack1_VC
meshres_vc.h, 442
GRP_F6TalkBack2_VC
meshres_vc.h, 442
GRP_F6TalkBack3_VC
meshres_vc.h, 442
GRP_F7_Event_M10_VC
meshres_vc.h, 442
meshres_vc_additions.h, 446
GRP_F7_Event_M_VC
meshres_vc.h, 442
meshres_vc_additions.h, 446
GRP_F7_Event_S10_VC
meshres_vc.h, 442
meshres_vc_additions.h, 446
GRP_F7_Event_S_VC
meshres_vc.h, 442
meshres_vc_additions.h, 446
GRP_F7CW_VC
meshres_vc.h, 442
GRP_F8button1_VC
meshres_vc.h, 442
GRP_F8button2_VC
meshres_vc.h, 442
GRP_F8button3_VC
meshres_vc.h, 442
GRP_F8cover1_VC
meshres_vc.h, 442
GRP_F8cover2_VC
meshres_vc.h, 442
GRP_F8Guard_VC
meshres_vc.h, 442
GRP_F8HUDbrt_VC
meshres_vc.h, 442
GRP_F8HUDdim_VC
meshres_vc.h, 442
GRP_F8HUDtest_VC
meshres_vc.h, 442
GRP_F8pan1_VC
meshres_vc.h, 442
GRP_F8panel_VC
meshres_vc.h, 442
GRP_F8switch1_VC
meshres_vc.h, 442
GRP_F8switch2_VC
meshres_vc.h, 442
GRP_F8switch3_VC
meshres_vc.h, 442
GRP_F8switch4_VC
meshres_vc.h, 442
GRP_F8switch5_VC
meshres_vc.h, 442
GRP_F8switch6_VC
meshres_vc.h, 442
GRP_F8switch7_VC
meshres_vc.h, 442
GRP_F8switch8_VC
meshres_vc.h, 442
GRP_F8switch9_VC
meshres_vc.h, 442
GRP_F8TalkBack1_VC
meshres_vc.h, 442
GRP_F8TalkBack2_VC
meshres_vc.h, 442
GRP_F8TalkBack3_VC
meshres_vc.h, 442
GRP_F9needle1_VC
meshres_vc.h, 442

GRP_F9needle2_VC
meshres_vc.h, 442
GRP_F9needle3_VC
meshres_vc.h, 442
GRP_F9panel_VC
meshres_vc.h, 442
GRP_Fault_VC
meshres_vc.h, 442
GRP_Fcovers_VC
meshres_vc.h, 442
GRP_five_VC
meshres_vc.h, 442
GRP_Fixers_OFF_ODSVC
meshres_odsvc.h, 422
GRP_FLAPL
meshres.h, 416
GRP_FLAPR
meshres.h, 416
GRP_FSS_WPS_Z_bracket
meshres_FSS_RSS.h, 394
GRP_FUSELAGE
meshres.h, 416
GRP_FWD_HOOKS
meshres.h, 416
GRP_FWD_RSS_drive_truck
meshres_FSS_RSS.h, 394
GRP_GEARDOORL
meshres.h, 416
GRP_GEARDOORR
meshres.h, 416
GRP_GEARL
meshres.h, 416
GRP_GEARR
meshres.h, 416
GRP_GOX_Cylinder06
meshres_FSS_RSS.h, 394
GRP_GOX_Cylinder342
meshres_FSS_RSS.h, 394
GRP_GOX_vent_arm_truss
meshres_FSS_RSS.h, 394
GRP_GPC_Status_VC
meshres_vc.h, 442
GRP_GPC_VC
meshres_vc.h, 442
GRP_Guardb_VC
meshres_vc.h, 442
GRP_guards_ODSVC
meshres_odsvc.h, 422
GRP_guards_VC
meshres_vc.h, 442
GRP_Hooks1_CLOSED_ODSVC
meshres_odsvc.h, 422
GRP_Hooks1_OPEN_ODSVC
meshres_odsvc.h, 422
GRP_Hooks2_CLOSED_ODSVC
meshres_odsvc.h, 422
GRP_Hooks2_OPEN_ODSVC
meshres_odsvc.h, 422
GRP_HUDL
meshres.h, 416
GRP_HUDR
meshres.h, 416
GRP_Humerus
meshres_RMS.h, 424
GRP_Initial_Contact_ODSVC
meshres_odsvc.h, 422
GRP_Inner_FSS_WPS_panel_track
meshres_FSS_RSS.h, 394
GRP_Inner_WPS_panel
meshres_FSS_RSS.h, 394
GRP_Interp_Sealed_ODSVC
meshres_odsvc.h, 422
GRP_invisiblebottom_VC
meshres_vc.h, 442
GRP_invisibletop_VC
meshres_vc.h, 442
GRP_IORESET_VC
meshres_vc.h, 442
GRP_ITEM_VC
meshres_vc.h, 442
GRP_KEYB_A_VC
meshres_vc.h, 442
GRP_KEYB_DOT_VC
meshres_vc.h, 442
GRP_KEYB_EXEC_VC
meshres_vc.h, 442
GRP_KEYB_F_VC
meshres_vc.h, 442
GRP_KEYB_FOUR_VC
meshres_vc.h, 442
GRP_KEYB_ONE_VC
meshres_vc.h, 442
GRP_KEYB_OPS_VC
meshres_vc.h, 442
GRP_KEYB_RESUME_VC
meshres_vc.h, 442
GRP_KEYB_SPEC_VC
meshres_vc.h, 442
GRP_KEYB_TWO_VC
meshres_vc.h, 442
GRP_KUBAND1
meshres.h, 416
GRP_KUBAND2
meshres.h, 416
GRP_KUBAND_BOX_KU
meshres_ku.h, 417
GRP_KUDISH_KU
meshres_ku.h, 417

GRP_KUGIMBAL_KU
meshres_ku.h, 417
GRP_L1b10_VC
meshres_vc.h, 442
GRP_L1b11_VC
meshres_vc.h, 442
GRP_L1b12_VC
meshres_vc.h, 442
GRP_L1b13_VC
meshres_vc.h, 442
GRP_L1b14_VC
meshres_vc.h, 442
GRP_L1b15_VC
meshres_vc.h, 442
GRP_L1b16_VC
meshres_vc.h, 442
GRP_L1b17_VC
meshres_vc.h, 442
GRP_L1b18_VC
meshres_vc.h, 442
GRP_L1b19_VC
meshres_vc.h, 442
GRP_L1b1_VC
meshres_vc.h, 442
GRP_L1b20_VC
meshres_vc.h, 442
GRP_L1b21_VC
meshres_vc.h, 442
GRP_L1b22_VC
meshres_vc.h, 442
GRP_L1b23_VC
meshres_vc.h, 442
GRP_L1b24_VC
meshres_vc.h, 442
GRP_L1b25_VC
meshres_vc.h, 442
GRP_L1b26_VC
meshres_vc.h, 442
GRP_L1b27_VC
meshres_vc.h, 442
GRP_L1b28_VC
meshres_vc.h, 442
GRP_L1b29_VC
meshres_vc.h, 442
GRP_L1b2_VC
meshres_vc.h, 442
GRP_L1b30_VC
meshres_vc.h, 442
GRP_L1b31_VC
meshres_vc.h, 442
GRP_L1b32_VC
meshres_vc.h, 442
GRP_L1b33_VC
meshres_vc.h, 442
GRP_L1b34_VC
meshres_vc.h, 442
GRP_L1b35_VC
meshres_vc.h, 442
GRP_L1b36_VC
meshres_vc.h, 442
GRP_L1b37_VC
meshres_vc.h, 442
GRP_L1b38_VC
meshres_vc.h, 442
GRP_L1b39_VC
meshres_vc.h, 442
GRP_L1b3_VC
meshres_vc.h, 442
GRP_L1b40_VC
meshres_vc.h, 442
GRP_L1b41_VC
meshres_vc.h, 442
GRP_L1b42_VC
meshres_vc.h, 442
GRP_L1b43_VC
meshres_vc.h, 442
GRP_L1b44_VC
meshres_vc.h, 442
GRP_L1b45_VC
meshres_vc.h, 442
GRP_L1b46_VC
meshres_vc.h, 442
GRP_L1b47_VC
meshres_vc.h, 442
GRP_L1b4_VC
meshres_vc.h, 442
GRP_L1b5_VC
meshres_vc.h, 442
GRP_L1b6_VC
meshres_vc.h, 442
GRP_L1b7_VC
meshres_vc.h, 442
GRP_L1b8_VC
meshres_vc.h, 442
GRP_L1b9_VC
meshres_vc.h, 442
GRP_L1cover1_VC
meshres_vc.h, 442
GRP_L1cover2_VC
meshres_vc.h, 442
GRP_L1cover3_VC
meshres_vc.h, 442
GRP_L1dial1_VC
meshres_vc.h, 442
GRP_L1dial2_VC
meshres_vc.h, 442
GRP_L1dial3_VC
meshres_vc.h, 442

GRP_L1Panel_VC
meshres_vc.h, 442
GRP_L2b10_VC
meshres_vc.h, 442
GRP_L2b11v_VC
meshres_vc.h, 442
GRP_L2b12_VC
meshres_vc.h, 442
GRP_L2b13_VC
meshres_vc.h, 442
GRP_L2b14_VC
meshres_vc.h, 442
GRP_L2b15_VC
meshres_vc.h, 442
GRP_L2b16_VC
meshres_vc.h, 442
GRP_L2b17v_VC
meshres_vc.h, 442
GRP_L2b18_VC
meshres_vc.h, 442
GRP_L2b19_VC
meshres_vc.h, 442
GRP_L2b1_VC
meshres_vc.h, 442
GRP_L2b20_VC
meshres_vc.h, 442
GRP_L2b21_VC
meshres_vc.h, 442
GRP_L2b22_VC
meshres_vc.h, 442
GRP_L2b23_VC
meshres_vc.h, 442
GRP_L2b24_VC
meshres_vc.h, 442
GRP_L2b25_VC
meshres_vc.h, 442
GRP_L2b2_VC
meshres_vc.h, 442
GRP_L2b3_VC
meshres_vc.h, 442
GRP_L2b4_VC
meshres_vc.h, 442
GRP_L2b5_VC
meshres_vc.h, 442
GRP_L2b6_VC
meshres_vc.h, 442
GRP_L2b7_VC
meshres_vc.h, 442
GRP_L2b8_VC
meshres_vc.h, 442
GRP_L2b9_VC
meshres_vc.h, 442
GRP_L2panel_VC
meshres_vc.h, 442
GRP_LampTest_ODSVC
meshres_odsvc.h, 422
GRP_Latches_CLOSED_ODSVC
meshres_odsvc.h, 422
GRP_Latches_OPEN_ODSVC
meshres_odsvc.h, 422
GRP_leftface_VC
meshres_vc.h, 442
GRP_LEFTFLAPPERIN
meshres.h, 416
GRP_LEFTFLAPPEROUT
meshres.h, 416
GRP_Lightning_mast
meshres_FSS_RSS.h, 394
GRP_Lightning_mast_foundation
meshres_FSS_RSS.h, 394
GRP_Line01
meshres_FSS_RSS.h, 394
GRP_Main_FSS
meshres_FSS_RSS.h, 394
GRP_MAPU_part_1
meshres_FSS_RSS.h, 394
GRP_MAPU_part_2
meshres_FSS_RSS.h, 394
GRP_MDUs_VC
meshres_vc.h, 442
GRP_measure_strips_VC
meshres_vc.h, 442
GRP_MFD11_VC
meshres_vc.h, 442
GRP_MFD1_VC
meshres_vc.h, 442
GRP_MFD1LABEL_VC
meshres_vc.h, 442
GRP_MFD2_VC
meshres_vc.h, 442
GRP_MFD2LABEL_VC
meshres_vc.h, 442
GRP_MFD3_VC
meshres_vc.h, 442
GRP_MFD3LABEL_VC
meshres_vc.h, 442
GRP_MFD4_VC
meshres_vc.h, 442
GRP_MFD4LABEL_VC
meshres_vc.h, 442
GRP_MFD5_VC
meshres_vc.h, 442
GRP_MFD5LABEL_VC
meshres_vc.h, 442
GRP_MFD_aft_VC
meshres_vc.h, 442
GRP_MFD_VC
meshres_vc.h, 442

GRP_MID_AFT_HOOKS
meshres.h, 416
GRP_MID_FWD_HOOKS
meshres.h, 416
GRP_minus_VC
meshres_vc.h, 442
GRP_MSG_VC
meshres_vc.h, 442
GRP_nine_VC
meshres_vc.h, 442
GRP_North_GN2_vent_pipe
meshres_FSS_RSS.h, 394
GRP_NOSEDOORL
meshres.h, 416
GRP_NOSEDOORR
meshres.h, 416
GRP_NOSEGEAR
meshres.h, 416
GRP_NOSEWHEEL
meshres.h, 416
GRP_O1_VC
meshres_vc.h, 442
GRP_O1M1P_VC
meshres_vc.h, 442
GRP_O1M1T_VC
meshres_vc.h, 442
GRP_O1M2F_VC
meshres_vc.h, 442
GRP_O1M2T_VC
meshres_vc.h, 442
GRP_O1M3F_VC
meshres_vc.h, 442
GRP_O1M3P_VC
meshres_vc.h, 442
GRP_O1M4P_VC
meshres_vc.h, 442
GRP_O1M4ppo_VC
meshres_vc.h, 442
GRP_O1S1_VC
meshres_vc.h, 442
GRP_O1S2_VC
meshres_vc.h, 442
GRP_O1S3_VC
meshres_vc.h, 442
GRP_O1S4_VC
meshres_vc.h, 442
GRP_O1S5_VC
meshres_vc.h, 442
GRP_O1S6_VC
meshres_vc.h, 442
GRP_O2_VC
meshres_vc.h, 442
GRP_O2M5T_VC
meshres_vc.h, 442
GRP_O2M6P_VC
meshres_vc.h, 442
GRP_O2M6Q_VC
meshres_vc.h, 442
GRP_O2M7P_VC
meshres_vc.h, 442
GRP_O2M7Q_VC
meshres_vc.h, 442
GRP_O2M8T_VC
meshres_vc.h, 442
GRP_O2S7_VC
meshres_vc.h, 442
GRP_O2S8_VC
meshres_vc.h, 442
GRP_O2S9_VC
meshres_vc.h, 442
GRP_O3_guards_VC
meshres_vc.h, 442
GRP_O3_MET_DAY1_VC
meshres_vc.h, 442
GRP_O3_MET_DAY2_VC
meshres_vc.h, 442
GRP_O3_MET_DAY3_VC
meshres_vc.h, 442
GRP_O3_MET_H1_VC
meshres_vc.h, 442
GRP_O3_MET_H2_VC
meshres_vc.h, 442
GRP_O3_MET_M1_VC
meshres_vc.h, 442
GRP_O3_MET_M2_VC
meshres_vc.h, 442
GRP_O3_MET_S1_VC
meshres_vc.h, 442
GRP_O3_MET_S2_VC
meshres_vc.h, 442
GRP_O3_PRPLT_FWD1_VC
meshres_vc.h, 442
GRP_O3_PRPLT_FWD2_VC
meshres_vc.h, 442
GRP_O3_PRPLT_LEFT2_VC
meshres_vc.h, 442
GRP_O3_PRPLT_RIGHT1_VC
meshres_vc.h, 442
GRP_O3_PRPLT_RIGHT2_VC
meshres_vc.h, 442
GRP_O3_VC
meshres_vc.h, 442
GRP_O3M10F_VC
meshres_vc.h, 442
GRP_O3M10O_VC
meshres_vc.h, 442
GRP_O3M11F_VC
meshres_vc.h, 442

GRP_O3M11O_VC
meshres_vc.h, 442
GRP_O3M9F_VC
meshres_vc.h, 442
GRP_O3M9O_VC
meshres_vc.h, 442
GRP_O3S10_VC
meshres_vc.h, 442
GRP_O3S11_VC
meshres_vc.h, 442
GRP_O3S12_VC
meshres_vc.h, 442
GRP_O6Guard_VC
meshres_vc.h, 442
GRP_O6panel_VC
meshres_vc.h, 442
GRP_O6R1_VC
meshres_vc.h, 442
GRP_O6R2_VC
meshres_vc.h, 442
GRP_O6R3_VC
meshres_vc.h, 442
GRP_O6R4_VC
meshres_vc.h, 442
GRP_O6R5_VC
meshres_vc.h, 442
GRP_O6R6_VC
meshres_vc.h, 442
GRP_O6R7_VC
meshres_vc.h, 442
GRP_O6RS6_VC
meshres_vc.h, 442
GRP_O6S12_VC
meshres_vc.h, 442
GRP_O6S13_VC
meshres_vc.h, 442
GRP_O6S14_VC
meshres_vc.h, 442
GRP_O6S15_VC
meshres_vc.h, 442
GRP_O6S16_VC
meshres_vc.h, 442
GRP_O6S17_VC
meshres_vc.h, 442
GRP_O6S18_VC
meshres_vc.h, 442
GRP_O6S19_VC
meshres_vc.h, 442
GRP_O6S1_VC
meshres_vc.h, 442
GRP_O6S20_VC
meshres_vc.h, 442
GRP_O6S21_VC
meshres_vc.h, 442
GRP_O6S22_VC
meshres_vc.h, 442
GRP_O6S23_VC
meshres_vc.h, 442
GRP_O6S24_VC
meshres_vc.h, 442
GRP_O6S25_VC
meshres_vc.h, 442
GRP_O6S26_VC
meshres_vc.h, 442
GRP_O6S27_VC
meshres_vc.h, 442
GRP_O6S28_VC
meshres_vc.h, 442
GRP_O6S29_VC
meshres_vc.h, 442
GRP_O6S2_VC
meshres_vc.h, 442
GRP_O6S30_VC
meshres_vc.h, 442
GRP_O6S31_VC
meshres_vc.h, 442
GRP_O6S32_VC
meshres_vc.h, 442
GRP_O6S33_VC
meshres_vc.h, 442
GRP_O6S34_VC
meshres_vc.h, 442
GRP_O6S35_VC
meshres_vc.h, 442
GRP_O6S36_VC
meshres_vc.h, 442
GRP_O6S37_VC
meshres_vc.h, 442
GRP_O6S38_VC
meshres_vc.h, 442
GRP_O6S39_VC
meshres_vc.h, 442
GRP_O6S3_VC
meshres_vc.h, 442
GRP_O6S45_VC
meshres_vc.h, 442
GRP_O6S46_VC
meshres_vc.h, 442
GRP_O6S47_VC
meshres_vc.h, 442
GRP_O6S48_VC
meshres_vc.h, 442
GRP_O6S49_VC
meshres_vc.h, 442
GRP_O6S4_VC
meshres_vc.h, 442
GRP_O6S50_VC
meshres_vc.h, 442

GRP_O6S51_VC
meshres_vc.h, 442
GRP_O6S52_VC
meshres_vc.h, 442
GRP_O6S5_VC
meshres_vc.h, 442
GRP_O6S6_VC
meshres_vc.h, 442
GRP_O6switchA_VC
meshres_vc.h, 442
GRP_O6switchB_VC
meshres_vc.h, 442
GRP_O6switchC_VC
meshres_vc.h, 442
GRP_O7Guard_VC
meshres_vc.h, 442
GRP_O7panel_VC
meshres_vc.h, 442
GRP_O7RS1_VC
meshres_vc.h, 442
GRP_O7RS3_VC
meshres_vc.h, 442
GRP_O7RS5_VC
meshres_vc.h, 442
GRP_O7S10_VC
meshres_vc.h, 442
GRP_O7S11_VC
meshres_vc.h, 442
GRP_O7S13_VC
meshres_vc.h, 442
GRP_O7S14_VC
meshres_vc.h, 442
GRP_O7S16a_VC
meshres_vc.h, 442
GRP_O7S16b_VC
meshres_vc.h, 442
GRP_O7S17a_VC
meshres_vc.h, 442
GRP_O7S17b_VC
meshres_vc.h, 442
GRP_O7S18a_VC
meshres_vc.h, 442
GRP_O7S18b_VC
meshres_vc.h, 442
GRP_O7S22_VC
meshres_vc.h, 442
GRP_O7S23_VC
meshres_vc.h, 442
GRP_O7S24_VC
meshres_vc.h, 442
GRP_O7S25_VC
meshres_vc.h, 442
GRP_O7S26_VC
meshres_vc.h, 442
GRP_O7S27_VC
meshres_vc.h, 442
GRP_O7S28_VC
meshres_vc.h, 442
GRP_O7S29_VC
meshres_vc.h, 442
GRP_O7S2_VC
meshres_vc.h, 442
GRP_O7S30_VC
meshres_vc.h, 442
GRP_O7S31_VC
meshres_vc.h, 442
GRP_O7S32_VC
meshres_vc.h, 442
GRP_O7S33_VC
meshres_vc.h, 442
GRP_O7S34_VC
meshres_vc.h, 442
GRP_O7S35_VC
meshres_vc.h, 442
GRP_O7S36_VC
meshres_vc.h, 442
GRP_O7S4_VC
meshres_vc.h, 442
GRP_O7S6_VC
meshres_vc.h, 442
GRP_O8Guard_VC
meshres_vc.h, 442
GRP_O8panel_VC
meshres_vc.h, 442
GRP_O8R1_VC
meshres_vc.h, 442
GRP_O8R2_VC
meshres_vc.h, 442
GRP_O8R3_VC
meshres_vc.h, 442
GRP_O8R4_VC
meshres_vc.h, 442
GRP_O8R5_VC
meshres_vc.h, 442
GRP_O8R6_VC
meshres_vc.h, 442
GRP_O8S10_VC
meshres_vc.h, 442
GRP_O8S12_VC
meshres_vc.h, 442
GRP_O8S13_VC
meshres_vc.h, 442
GRP_O8S14_VC
meshres_vc.h, 442
GRP_O8S15_VC
meshres_vc.h, 442
GRP_O8S16_VC
meshres_vc.h, 442

GRP_O8S17_VC
meshres_vc.h, 442
GRP_O8S18_VC
meshres_vc.h, 442
GRP_O8S19_VC
meshres_vc.h, 442
GRP_O8S1_VC
meshres_vc.h, 442
GRP_O8S1a_VC
meshres_vc.h, 442
GRP_O8S20_VC
meshres_vc.h, 442
GRP_O8S21_VC
meshres_vc.h, 442
GRP_O8S22_VC
meshres_vc.h, 442
GRP_O8S23_VC
meshres_vc.h, 442
GRP_O8S24_VC
meshres_vc.h, 442
GRP_O8S26_VC
meshres_vc.h, 442
GRP_O8S27_VC
meshres_vc.h, 442
GRP_O8S28_VC
meshres_vc.h, 442
GRP_O8S29_VC
meshres_vc.h, 442
GRP_O8S2a_VC
meshres_vc.h, 442
GRP_O8S30_VC
meshres_vc.h, 442
GRP_O8S31_VC
meshres_vc.h, 442
GRP_O8S32_VC
meshres_vc.h, 442
GRP_O8S33_VC
meshres_vc.h, 442
GRP_O8S34_VC
meshres_vc.h, 442
GRP_O8S3_VC
meshres_vc.h, 442
GRP_O8S4_VC
meshres_vc.h, 442
GRP_O8S5_VC
meshres_vc.h, 442
GRP_O8S6_VC
meshres_vc.h, 442
GRP_O8S7_VC
meshres_vc.h, 442
GRP_O8S8_VC
meshres_vc.h, 442
GRP_O8S9_VC
meshres_vc.h, 442
GRP_OMSL
meshres.h, 416
GRP_OMSR
meshres.h, 416
GRP_OPEN_Hooks_ODSVC
meshres_odsvc.h, 422
GRP_OPEN_Latches_ODSVC
meshres_odsvc.h, 422
GRP_Orbiter_Access_Arm
meshres_FSS_RSS.h, 394
GRP_Outer_WPS_panel
meshres_FSS_RSS.h, 394
GRP_panelA4_VC
meshres_vc.h, 442
meshres_vc_additions.h, 446
GRP_panelR4_VC
meshres_vc.h, 442
GRP_Pas_Hooks_Firing_ODSVC
meshres_odsvc.h, 422
GRP_plateguard1_ODSVC
meshres_odsvc.h, 422
GRP_plateguard2_ODSVC
meshres_odsvc.h, 422
GRP_PLT1_VC
meshres_vc.h, 442
GRP_PLT1LABEL_VC
meshres_vc.h, 442
GRP_PLT2_VC
meshres_vc.h, 442
GRP_PLT2LABEL_VC
meshres_vc.h, 442
GRP_plus_VC
meshres_vc.h, 442
GRP_PORT_CLAMP
meshres.h, 416
GRP_PORTDOOR_PULL
meshres.h, 416
GRP_PORTPUSH_ROD
meshres.h, 416
GRP_Power_ON_ODSVC
meshres_odsvc.h, 422
GRP_PRO_VC
meshres_vc.h, 442
GRP_PROBEL
meshres.h, 416
GRP_PROBER
meshres.h, 416
GRP_Pyro_Circ_Prot_OFF_ODSVC
meshres_odsvc.h, 422
GRP_Pyro_Circ_Prot_ON_ODSVC
meshres_odsvc.h, 422
GRP_R13L_VC
meshres_vc.h, 442
GRP_R13talk1_VC

meshres_vc.h, 442
GRP_R13talk2_VC
meshres_vc.h, 442
GRP_R13talk3_VC
meshres_vc.h, 442
GRP_R13talk4_VC
meshres_vc.h, 442
GRP_R13talk5_VC
meshres_vc.h, 442
GRP_R13talk6_VC
meshres_vc.h, 442
GRP_R13talk7_VC
meshres_vc.h, 442
GRP_R13talk8_VC
meshres_vc.h, 442
GRP_R1b1_VC
meshres_vc.h, 442
GRP_R1b2_VC
meshres_vc.h, 442
GRP_R1b3_VC
meshres_vc.h, 442
GRP_R1b4_VC
meshres_vc.h, 442
GRP_R1b5_VC
meshres_vc.h, 442
GRP_R2b10_VC
meshres_vc.h, 442
GRP_R2b11_VC
meshres_vc.h, 442
GRP_R2b12_VC
meshres_vc.h, 442
GRP_R2b13_VC
meshres_vc.h, 442
GRP_R2b14_VC
meshres_vc.h, 442
GRP_R2b15_VC
meshres_vc.h, 442
GRP_R2b16_VC
meshres_vc.h, 442
GRP_R2b17_VC
meshres_vc.h, 442
GRP_R2b18_VC
meshres_vc.h, 442
GRP_R2b19_VC
meshres_vc.h, 442
GRP_R2b1_VC
meshres_vc.h, 442
GRP_R2b20_VC
meshres_vc.h, 442
GRP_R2b21_VC
meshres_vc.h, 442
GRP_R2b22_VC
meshres_vc.h, 442
GRP_R2b23_VC
meshres_vc.h, 442
GRP_R2b24_VC
meshres_vc.h, 442
GRP_R2b25_VC
meshres_vc.h, 442
GRP_R2b26_VC
meshres_vc.h, 442
GRP_R2b27_VC
meshres_vc.h, 442
GRP_R2b28_VC
meshres_vc.h, 442
GRP_R2b29_VC
meshres_vc.h, 442
GRP_R2b2_VC
meshres_vc.h, 442
GRP_R2b30_VC
meshres_vc.h, 442
GRP_R2b31_VC
meshres_vc.h, 442
GRP_R2b32_VC
meshres_vc.h, 442
GRP_R2b33_VC
meshres_vc.h, 442
GRP_R2b34_VC
meshres_vc.h, 442
GRP_R2b35_VC
meshres_vc.h, 442
GRP_R2b36_VC
meshres_vc.h, 442
GRP_R2b37_VC
meshres_vc.h, 442
GRP_R2b38_VC
meshres_vc.h, 442
GRP_R2b39_VC
meshres_vc.h, 442
GRP_R2b3_VC
meshres_vc.h, 442
GRP_R2b40_VC
meshres_vc.h, 442
GRP_R2b41_VC
meshres_vc.h, 442
GRP_R2b42_VC
meshres_vc.h, 442
GRP_R2b43_VC
meshres_vc.h, 442
GRP_R2b44_VC
meshres_vc.h, 442
GRP_R2b45_VC
meshres_vc.h, 442
GRP_R2b46_VC
meshres_vc.h, 442
GRP_R2b47_VC
meshres_vc.h, 442
GRP_R2b48_VC

meshres_vc.h, 442
GRP_R2b49_VC
meshres_vc.h, 442
GRP_R2b4_VC
meshres_vc.h, 442
GRP_R2b50_VC
meshres_vc.h, 442
GRP_R2b51_VC
meshres_vc.h, 442
GRP_R2b52_VC
meshres_vc.h, 442
GRP_R2b53_VC
meshres_vc.h, 442
GRP_R2b54_VC
meshres_vc.h, 442
GRP_R2b55_VC
meshres_vc.h, 442
GRP_R2b56_VC
meshres_vc.h, 442
GRP_R2b5_VC
meshres_vc.h, 442
GRP_R2b6_VC
meshres_vc.h, 442
GRP_R2b7_VC
meshres_vc.h, 442
GRP_R2b8_VC
meshres_vc.h, 442
GRP_R2b9_VC
meshres_vc.h, 442
GRP_R2guards_VC
meshres_vc.h, 442
GRP_R2panel_VC
meshres_vc.h, 442
GRP_R4Guard_VC
meshres_vc.h, 442
GRP_R4S10_VC
meshres_vc.h, 442
GRP_R4S11_VC
meshres_vc.h, 442
GRP_R4S12_VC
meshres_vc.h, 442
GRP_R4S13_VC
meshres_vc.h, 442
GRP_R4S14_VC
meshres_vc.h, 442
GRP_R4S15_VC
meshres_vc.h, 442
GRP_R4S16_VC
meshres_vc.h, 442
GRP_R4S17_VC
meshres_vc.h, 442
GRP_R4S18_VC
meshres_vc.h, 442
GRP_R4S19_VC
meshres_vc.h, 442
GRP_R4S1_VC
meshres_vc.h, 442
GRP_R4S21_VC
meshres_vc.h, 442
GRP_R4S22_VC
meshres_vc.h, 442
GRP_R4S23_VC
meshres_vc.h, 442
GRP_R4S24_VC
meshres_vc.h, 442
GRP_R4S25_VC
meshres_vc.h, 442
GRP_R4S26_VC
meshres_vc.h, 442
GRP_R4S27_VC
meshres_vc.h, 442
GRP_R4S28_VC
meshres_vc.h, 442
GRP_R4S29_VC
meshres_vc.h, 442
GRP_R4S2_VC
meshres_vc.h, 442
GRP_R4S30_VC
meshres_vc.h, 442
GRP_R4S3_VC
meshres_vc.h, 442
GRP_R4S4_VC
meshres_vc.h, 442
GRP_R4S5_VC
meshres_vc.h, 442
GRP_R4S6_VC
meshres_vc.h, 442
GRP_R4S7_VC
meshres_vc.h, 442
GRP_R4S8_VC
meshres_vc.h, 442
GRP_R4S9_VC
meshres_vc.h, 442
GRP_RADIATORBL
meshres.h, 416
GRP_RADIATORBR
meshres.h, 416
GRP_RADIATORFL
meshres.h, 416
GRP_RADIATORFR
meshres.h, 416
GRP_Radii
meshres_RMS.h, 424
GRP_READY_To_Hook_ODSVC
meshres_odsvc.h, 422
GRP_Rear_RSS_drive_truck
meshres_FSS_RSS.h, 394
GRP_rightface_VC

meshres_vc.h, 442
GRP_RIGHTFLAPPERIN
meshres.h, 416
GRP_RIGHTFLAPPEROUT
meshres.h, 416
GRP_Ring_Aligned_ODSVC
meshres_odsvc.h, 422
GRP_Ring_Final_Pos_ODSVC
meshres_odsvc.h, 422
GRP_Ring_Forward_Pos_ODSVC
meshres_odsvc.h, 422
GRP_Ring_Initial_Pos_ODSVC
meshres_odsvc.h, 422
GRP_RMS_MPMs
meshres_RMS.h, 424
GRP_RSS
meshres_FSS_RSS.h, 394
GRP_RSS_cab
meshres_FSS_RSS.h, 394
GRP_RSS_misc
meshres_FSS_RSS.h, 394
GRP_RSS_sheild
meshres_FSS_RSS.h, 394
GRP_RUDDERL
meshres.h, 416
GRP_RUDDERR
meshres.h, 416
GRP_seven_VC
meshres_vc.h, 442
GRP_Shoulder_Yaw
meshres_RMS.h, 424
GRP_six_VC
meshres_vc.h, 442
GRP_South_GN2_vent_pipe
meshres_FSS_RSS.h, 394
GRP_SSMEL
meshres.h, 416
GRP_SSMER
meshres.h, 416
GRP_SSMET
meshres.h, 416
GRP_STARTRACKERS
meshres.h, 416
GRP_STBD_CLAMP
meshres.h, 416
GRP_STBDDOOR_PULL
meshres.h, 416
GRP_STBDPUSH_ROD
meshres.h, 416
GRP_stripbases_VC
meshres_vc.h, 442
GRP_SwitchR13L_10_VC
meshres_vc.h, 442
GRP_SwitchR13L_11_VC
meshres_vc.h, 442
GRP_SwitchR13L_12_VC
meshres_vc.h, 442
GRP_SwitchR13L_13_VC
meshres_vc.h, 442
GRP_SwitchR13L_1_VC
meshres_vc.h, 442
GRP_SwitchR13L_2_VC
meshres_vc.h, 442
GRP_SwitchR13L_3_VC
meshres_vc.h, 442
GRP_SwitchR13L_4_VC
meshres_vc.h, 442
GRP_SwitchR13L_5_VC
meshres_vc.h, 442
GRP_SwitchR13L_6_VC
meshres_vc.h, 442
GRP_SwitchR13L_7_VC
meshres_vc.h, 442
GRP_SwitchR13L_8_VC
meshres_vc.h, 442
GRP_SwitchR13L_9_VC
meshres_vc.h, 442
GRP_Sys_VC
meshres_vc.h, 442
GRP_three_VC
meshres_vc.h, 442
GRP_topcup_VC
meshres_vc.h, 442
GRP_topdial_VC
meshres_vc.h, 442
GRP_topface_VC
meshres_vc.h, 442
GRP_TRACKBOX
meshres.h, 416
GRP_TRACKERPLATES
meshres.h, 416
GRP_transboot_VC
meshres_vc.h, 442
GRP_transguard_VC
meshres_vc.h, 442
GRP_translational_VC
meshres_vc.h, 442
GRP_transRMS_VC
meshres_vc.h, 442
GRP_transRMSboot_VC
meshres_vc.h, 442
GRP_Undock_Complet_ODSVC
meshres_odsvc.h, 422
GRP_Undocking_ODSVC
meshres_odsvc.h, 422
GRP_Vent_hood
meshres_FSS_RSS.h, 394
GRP_VirtualHUD_VC

meshres_vc.h, 442
GRP_WHEELL
meshres.h, 416
GRP_WHEELR
meshres.h, 416
GRP_White_Room_Mat
meshres_FSS_RSS.h, 394
GRP_White_Room_Part_1
meshres_FSS_RSS.h, 394
GRP_White_Room_Part_2
meshres_FSS_RSS.h, 394
GRP_WINDOWS
meshres.h, 416
GRP_WPS_panel
meshres_FSS_RSS.h, 394
GRP_WPS_panels_Y_support_struts
meshres_FSS_RSS.h, 394
GRP_Wrist_Yaw
meshres_RMS.h, 424
GRP_Wristpitch
meshres_RMS.h, 424
GRP_Y_FSS_WPS_struts
meshres_FSS_RSS.h, 394
GRP_YSTARTRACKDOOR
meshres.h, 416
GRP_zero_VC
meshres_vc.h, 442
GRP_ZSTARTRACKDOOR
meshres.h, 416
grpIndex
vc::LockableLever, 183
vc::StandardSwitch, 289
Guide
Atlantis, 55
GVA_OPERATING_SPEED
SSU_FSS_RSS.h, 397
GVA_proc
SSU_FSS_RSS, 285
GVA_status
SSU_FSS_RSS, 285
GVesselPos
Atlantis, 66
CRT, 130
GVesselVel
Atlantis, 66
CRT, 130

h
Atlantis, 66
hAC_arm
Atlantis, 66
hAC_sat
Atlantis, 66
hAC_satref
Atlantis, 66
HALFC.cpp
main, 403
HasExternalAirlock
Atlantis, 53
HasOwnVCMesh
vc::BasicPanel, 96
HasSwitch
vc::BasicPanel, 96
hbodyflap
Atlantis, 66
hCRTFont
CRT, 130
MFD_GDIPARAM, 202
hDlg
PayloadBayOp, 269
hDLL
GDIParams, 155
MFD_GDIPARAM, 202
HDP_POS
MLP.h, 449
HE_ISOLATION_A
PanelR2, 260
HE_ISOLATION_B
PanelR2, 260
height
CRT, 130
helevator
Atlantis, 66
hh
Atlantis, 66
HideMidDeck
Atlantis, 53
HideODS
Atlantis, 53
hInst
AtlantisConfig.cpp, 338
hKUBandMesh
Atlantis, 60
hlaileron
Atlantis, 66
HLiftCoeff
Atlantis.cpp, 307
hMidDeckMesh
Atlantis, 60
hMMU
Atlantis, 66
hODSMesh
Atlantis, 60
hOrbiterCockpitMesh
Atlantis, 60
hOrbiterMesh
Atlantis, 60
hOrbiterRMSMesh

Atlantis, 60
hOrbiterVCMesh
Atlantis, 60
hraileron
Atlantis, 66
hrudder
Atlantis, 66
hSAT
Atlantis, 66
hSRBMesh
Atlantis, 60
Atlantis_SRB, 69
hSRBMesh_Left
Atlantis_SRB, 69
hSRBMesh_Right
Atlantis_SRB, 69
HT
Atlantis, 66
hT
Atlantis, 66
hTankMesh
Atlantis, 60
Atlantis_Tank, 70
huds
Atlantis, 66
hv
Atlantis, 66
HYD_CIRC_PUMP
PanelR2, 260
HYD_MAIN_PUMP_PRESS
PanelR2, 260
HydActuator
hydraulics::HydActuator, 162
Hydraulic_Press
Atlantis, 66
CRT, 130
PanelR2, 260
hydraulics, 19
hydraulics::HydActuator, 161
~HydActuator, 162
GetActPos, 162
HydActuator, 162
InA, 162
InB, 162
InC, 162
InD, 162
OutA, 162
OutB, 162
OutC, 162
OutD, 162

iBranchCode
mission::Mission, 204
ID
mps::EIU, 149
mps::SSME, 283
id
CRT, 130
IDB_7SEGDIGITS
resource.h, 493
IDB_CLOCKDIGITS
resource.h, 493
IDB_DEUCHARACTERS
resource.h, 493
IDB_TKBKLABEL
resource.h, 493
IDC_BUTTON1
resource.h, 493
IDC_BUTTON10
resource.h, 493
IDC_BUTTON14
resource.h, 493
IDC_BUTTON15
resource.h, 493
IDC_BUTTON16
resource.h, 493
IDC_BUTTON17
resource.h, 493
IDC_BUTTON2
resource.h, 493
IDC_BUTTON5
resource.h, 493
IDC_BUTTON6
resource.h, 493
IDC_BUTTON9
resource.h, 493
IDC_CAM_BL
resource.h, 493
IDC_CAM_BR
resource.h, 493
IDC_CAM_DOWN
resource.h, 493
IDC_CAM_FL
resource.h, 493
IDC_CAM_FR
resource.h, 493
IDC_CAM_HIGHRATE
resource.h, 493
IDC_CAM_LEFT
resource.h, 493
IDC_CAM_LOWRATE
resource.h, 493
IDC_CAM_RIGHT
resource.h, 493
IDC_CAM_ROT
resource.h, 493
IDC_CAM_SEL
resource.h, 493

IDC_CAM_UP
resource.h, 493
IDC_DEG01
resource.h, 493
IDC_DEG1
resource.h, 493
IDC_DEG10
resource.h, 493
IDC_ELBOW_PITCHDOWN
resource.h, 493
IDC_ELBOW_PITCHUP
resource.h, 493
IDC_GRAPPLE
resource.h, 493
IDC_KU
resource.h, 493
IDC_KU_DIRECT
resource.h, 493
IDC_KU_TLKBK
resource.h, 493
IDC_LATCHA
resource.h, 493
IDC_LATCHB
resource.h, 493
IDC_LATCHP_TLKBK
resource.h, 493
IDC_LATCHS_TLKBK
resource.h, 493
IDC_MECH1
resource.h, 493
IDC_MECH2
resource.h, 493
IDC_PAYCAM
resource.h, 493
IDC_PAYLOAD
resource.h, 493
IDC_PLBAYOP
resource.h, 493
IDC_PLBD
resource.h, 493
IDC_PLBD1
resource.h, 493
IDC_PLBD2
resource.h, 493
IDC_PLBD_TLKBK
resource.h, 493
IDC_RADA
resource.h, 493
IDC_RADB
resource.h, 493
IDC_RADIO1
AC_resource.h, 301
IDC_RADIO2
AC_resource.h, 301
IDC_RADIO3
AC_resource.h, 301
IDC_RADIO4
AC_resource.h, 301
IDC_RADP_TLKBK
resource.h, 493
IDC_RADS_TLKBK
resource.h, 493
IDC_RELEASE
resource.h, 493
IDC_RMSOP
resource.h, 493
IDC_SHOULDER_PITCHDOWN
resource.h, 493
IDC_SHOULDER_PITCHUP
resource.h, 493
IDC_SHOULDER_YAWLEFT
resource.h, 493
IDC_SHOULDER_YAWRIGHT
resource.h, 493
IDC_SHOWGRAPPLE
resource.h, 493
IDC_STOW
resource.h, 493
IDC_TRANS_MX
resource.h, 493
IDC_TRANS_MY
resource.h, 493
IDC_TRANS_MZ
resource.h, 493
IDC_TRANS_PX
resource.h, 493
IDC_TRANS_PY
resource.h, 493
IDC_TRANS_PZ
resource.h, 493
IDC_WRIST_PITCHDOWN
resource.h, 493
IDC_WRIST_PITCHUP
resource.h, 493
IDC_WRIST_ROLLLEFT
resource.h, 493
IDC_WRIST_ROLLRIGHT
resource.h, 493
IDC_WRIST_YAWLEFT
resource.h, 493
IDC_WRIST_YAWRIGHT
resource.h, 493
IDD_ACONFIG
AC_resource.h, 301
IDD_CTRL
resource.h, 493
IDD_PAYLOADCAM
resource.h, 493

IDD_PLBAY
resource.h, 493
IDD_RMS
resource.h, 493
ident
AtlantisSubsystem, 82
discsignals::DiscreteBundle, 141
vc::BasicVCComponent, 109
IDI_DOWN
resource.h, 493
IDI_LEFT
resource.h, 493
IDI_RIGHT
resource.h, 493
IDI_RLEFT
resource.h, 493
IDI_RRIGHT
resource.h, 493
IDI_UP
resource.h, 493
IDP
dps::IDP, 166
IDPSoftware
dps::IDPSoftware, 171
iETVersion
mission::Mission, 204
Ignition
mps::SSME, 277
IgnitionTime
Atlantis, 66
iMenu
CommModeHandler, 119
InA
hydraulics::HydActuator, 162
InB
hydraulics::HydActuator, 162
InC
hydraulics::HydActuator, 162
InD
hydraulics::HydActuator, 162
Indicator7Seg
vc::Indicator7Seg, 173
InertialOrientationRad
Atlantis, 66
InitDialog
AtlantisConfig, 75
InitialAnimState
vc::BasicVCComponent, 104
InitializeAutopilot
Atlantis, 55
InitModule
Atlantis.cpp, 307
AtlantisConfig.cpp, 338
CRT.cpp, 352
MLP.cpp, 448
Input
Atlantis, 55
CRT, 125
input
KeyboardInput, 179
vc::BasicLight, 90
vc::LockableLever, 183
vc::StandardSwitch, 289
InputSize
KeyboardInput, 179
IPL
dps::IDP, 166
ipl_software
dps::IDP, 168
iRevisionCode
mission::Mission, 204
IsCRTBufferEnabled
vc::MDU, 196
IsDeployed
AirDataProbeSystem, 33
IsInCommMode
CommModeHandler, 118
IsOwnRegion
vc::BasicVCComponent, 106
Isp
Atlantis, 66
ISP0
mps::SSME, 283
ISP1
mps::SSME, 283
IsPointOver
vc::BasicVCComponent, 106
IsSet
discsignals::DiscInPort, 137
ISubsystem, 174
CountComponents, 175
DelComponent, 175
GetIdentifier, 176
GetQualifiedIdentifier, 176
OnDumpToLog, 176
OnFail, 176
OnParseLine, 176
OnPlaybackEvent, 175
OnPostStep, 175
OnPreStep, 175
OnPropagate, 175
OnSaveState, 176
Realize, 175
RegisterComponent, 175
ISubsystem.h
AFX_ISUBSYSTEM_H__A4896D4F_-46E0_4D68_AEA4_F6B9CF259711__-INCLUDED_, 407

IsValidSPEC
Atlantis, 53
ITEM
KeyboardInput, 179
item
Atlantis, 66
AtlantisConfig.cpp, 338
CRT, 130

JetsEnabled
Atlantis, 66
Jettison
Atlantis, 53
jettison_time
Atlantis, 66

Keyboard, 177
Atlantis, 57, 178
ClearInput, 178
CRT, 130, 178
Keyboard, 178
keyboard, 178
RegisterVC, 178
sts, 178
VCMouseEvent, 178
keyboard
Keyboard, 178
Keyboard.h
DEU_KEY_RESUME, 409
KeyboardInput, 179
EXEC, 179
input, 179
InputSize, 179
ITEM, 179
NewEntry, 179
OPS, 179
PRO, 179
SPEC, 179
KeybSelectA
dps::IDP, 168
KeybSelectB
dps::IDP, 168
kg_to_pounds
Atlantis.h, 326
KU_DEPLOY
PayloadBayOp, 264
KU_DIRECT_OFF
PayloadBayOp, 264
KU_DIRECT_ON
PayloadBayOp, 264
KU_GND
PayloadBayOp, 264
KU_STOW
PayloadBayOp, 264
KU_OPERATING_SPEED
Atlantis.h, 326
KuAntennaStatus
PayloadBayOp, 269
KuBandSystem, 180
~KuBandSystem, 181
alpha_angle, 181
beta_angle, 181
bHeatersOn, 181
DeployState, 181
fDeployPos, 181
fGimbalLockState, 181
GetAzimuth, 181
GetSignalStrength, 181
GimbalLockState, 181
KuBandSystem, 181
ParseLine, 181
PostStep, 181
PreStep, 181
Propagate, 181
SaveState, 181
SetSlew, 181
usGPCMode, 181
KuCtrl
PayloadBayOp, 269
KuDirectCtrl
PayloadBayOp, 269

L_ISOLVLV
OMSPod.h, 464
labels
vc::BasicSwitch, 99
landed
CRT, 130
last_mfd
Atlantis, 66
last_target_pitch
Atlantis, 66
last_target_roll
Atlantis, 66
last_target_yaw
Atlantis, 66
LATCH_OPERATING_SPEED
Atlantis.h, 326
LATCH_STATE
Atlantis.h, 323
LAUNCH_SITE
Atlantis.h, 326
Launch_time
Atlantis, 58
CRT, 130
LaunchClamps
Atlantis, 53
LBM

Atlantis.h, 326
LC_LATCH
PayloadBayOp, 264
LC_OFF
PayloadBayOp, 264
LC_RELEASE
PayloadBayOp, 264
ldoor_drag
Atlantis, 60
LDoorLatches
PanelR2, 260
LeaveCommMode
CommModeHandler, 118
left_deploy
AirDataProbeSystem, 34
LEFT_DOOR
PanelR2, 260
LEFT_LATCH
PanelR2, 260
left_mode
AirDataProbeSystem, 34
left_state
AirDataProbeSystem, 34
left_stow_state
AirDataProbeSystem, 34
leftPod
OMSSubsystem, 221
LETUmbDoorStatus
PanelR2, 260
liCurrent
Stopwatch, 294
liFrequency
Stopwatch, 294
liLast
Stopwatch, 294
linterp
Atlantis.cpp, 307
GPC.cpp, 401
Listen
dps::BIU, 111
listerp
Atlantis.cpp, 307
GPC.cpp, 401
ll
Atlantis, 66
lMemory
dps::AP101S, 36
LOADBMP
Atlantis.cpp, 306
LoadManeuver
Atlantis, 55
CRT, 127
LoadMemoryConfiguration
dps::AP101S, 36
LoadMission
mission::Mission, 204
LoadSoftware
dps::AP101S, 36
LoadState
SSU_FSS_RSS, 285
local_vcidx
vc::BasicPanel, 96
LockableLever
vc::LockableLever, 183
LogicPwrOn
MechActuator, 201
LT_CLOSING
Atlantis.h, 323
LT_LATCHED
Atlantis.h, 323
LT_OPENING
Atlantis.h, 323
LT_RELEASED
Atlantis.h, 323
lu
Atlantis, 66
LVLH_X
Atlantis, 66
LVLH_Y
Atlantis, 66
LVLH_Z
Atlantis, 66
LVLHError
Atlantis, 66
LVLHOrientationReqd
Atlantis, 66
LVLHRateVector
Atlantis, 66

m
Atlantis, 66
mach
Atlantis.cpp, 310
main
HALFC.cpp, 403
majfunc
dps::IDP, 168
MajorCycle
Atlantis, 55
MajorFuncGNC
dps::IDP, 168
MajorFuncPL
dps::IDP, 168
MAJORFUNCTION
dps, 15
MakeBusMaster
dps::BIU, 111
Maneuver

Atlantis, 55
ManeuverinProg
Atlantis, 66
ManeuverStatus
Atlantis, 66
Mass
Atlantis, 66
MasterTimingUnit
dps::MasterTimingUnit, 187
MasterTimingUnit.h
AFX_MASTERTIMINGUNIT_H__-FEF22983_033D_412B_B3A9_-876EE3A3EC04__INCLUDED_, 376
MAX_ATT_LAUNCH
Atlantis.h, 326
MAX_GRAPPLING_ANGLE
Atlantis.h, 326
MAX_GRAPPLING_DIST
Atlantis.h, 326
MAX_ROLL_SRB
Atlantis.h, 326
maxrate
BasicValve, 101
MaxThrust
Atlantis, 66
MDM
dps::MDM, 191
MDM.h
AFX_MDM_H__BC1B13B9_53D9_4DC9_-A6D6_2E7EA9F155A6__INCLUDED_, 378
mdot
FLOWSTATE, 154
MDU
vc::MDU, 194
MDU.cpp
g_Param, 511
MDU_ADI
vc::MDU, 194
MDU_ADI_AVVI
vc::MDU, 194
MDU_COMP_ADI_HSI
vc::MDU, 194
MDU_CONFIGSTATUS
vc::MDU, 194
MDU_CST
vc::MDU, 194
MDU_DPS
vc::MDU, 194
MDU_FAULTSUMM
vc::MDU, 194
MDU_HSI_AMI
vc::MDU, 194
MDU_HYD_APU
vc::MDU, 194
MDU_MEMORYMGMT
vc::MDU, 194
MDU_OMS_MPS
vc::MDU, 194
MDU_PORT_SELECT
vc::MDU, 194
MDU_SPI
vc::MDU, 194
mdu_list
dps::IDP, 168
MDU_MODE
vc::MDU, 194
MDUID_AFD
vc, 25
MDUID_CDR1
vc, 25
MDUID_CDR2
vc, 25
MDUID_CRT1
vc, 25
MDUID_CRT2
vc, 25
MDUID_CRT3
vc, 25
MDUID_CRT4
vc, 25
MDUID_MFD1
vc, 25
MDUID_MFD2
vc, 25
MDUID_PLT1
vc, 25
MDUID_PLT2
vc, 25
mdus
Atlantis, 66
MechActuator, 199
~MechActuator, 200
CmdDriveFwd, 201
CmdDriveRwd, 201
fObjectTorque, 201
fOutPos, 201
LogicPwrOn, 201
MechActuator, 200
object_anim, 201
OnPostStep, 200
OnPreStep, 200
OnPropagate, 200
PosLimit0, 201
PosLimit1, 201
SetMotorParams, 201
SetObjectAnim, 201
SetObjectParams, 201

MechPwr
PayloadBayOp, 269
mesh_cargo_static
Atlantis, 58
mesh_cockpit
Atlantis, 58
mesh_kuband
Atlantis, 58
mesh_middeck
Atlantis, 58
mesh_ods
Atlantis, 58
mesh_orbiter
Atlantis, 58
mesh_rms
Atlantis, 58
mesh_srb
Atlantis, 58
mesh_tank
Atlantis, 58
MESH_UNDEFINED
Atlantis.h, 326
mesh_vc
Atlantis, 58
meshres.h
GRP_AFT_HOOKS, 416
GRP_AILERONL, 416
GRP_AILERONR, 416
GRP_BODYFLAP, 416
GRP_CAMERABL, 416
GRP_CAMERABR, 416
GRP_CAMERAFL, 416
GRP_CAMERAFR, 416
GRP_CAMERAPIVOTBL, 416
GRP_CAMERAPIVOTBR, 416
GRP_CAMERAPIVOTFL, 416
GRP_CAMERAPIVOTFR, 416
GRP_CARGOBAY, 416
GRP_CARGODOORINL, 416
GRP_CARGODOORINR, 416
GRP_CARGODOOROUTL, 416
GRP_CARGODOOROUTR, 416
GRP_ETUMBDOORL, 416
GRP_ETUMBDOORR, 416
GRP_FLAPL, 416
GRP_FLAPR, 416
GRP_FUSELAGE, 416
GRP_FWD_HOOKS, 416
GRP_GEARDOORL, 416
GRP_GEARDOORR, 416
GRP_GEARL, 416
GRP_GEARR, 416
GRP_HUDL, 416
GRP_HUDR, 416
GRP_KUBAND1, 416
GRP_KUBAND2, 416
GRP_LEFTFLAPPERIN, 416
GRP_LEFTFLAPPEROUT, 416
GRP_MID_AFT_HOOKS, 416
GRP_MID_FWD_HOOKS, 416
GRP_NOSEDOORL, 416
GRP_NOSEDOORR, 416
GRP_NOSEGEAR, 416
GRP_NOSEWHEEL, 416
GRP_OMSL, 416
GRP_OMSR, 416
GRP_PORT_CLAMP, 416
GRP_PORTDOOR_PULL, 416
GRP_PORTPUSH_ROD, 416
GRP_PROBEL, 416
GRP_PROBER, 416
GRP_RADIATORBL, 416
GRP_RADIATORBR, 416
GRP_RADIATORFL, 416
GRP_RADIATORFR, 416
GRP_RIGHTFLAPPERIN, 416
GRP_RIGHTFLAPPEROUT, 416
GRP_RUDDERL, 416
GRP_RUDDERR, 416
GRP_SSMEL, 416
GRP_SSMER, 416
GRP_SSMET, 416
GRP_STARTRACKERS, 416
GRP_STBD_CLAMP, 416
GRP_STBDDOOR_PULL, 416
GRP_STBDPUSH_ROD, 416
GRP_TRACKBOX, 416
GRP_TRACKERPLATES, 416
GRP_WHEELL, 416
GRP_WHEELR, 416
GRP_WINDOWS, 416
GRP_YSTARTRACKDOOR, 416
GRP_ZSTARTRACKDOOR, 416
NGRP, 416
NMAT, 416
NTEX, 416
meshres_FSS_RSS.h
GRP_Box01, 394
GRP_Box02, 394
GRP_Box03, 394
GRP_Box04, 394
GRP_Box05, 394
GRP_Box06, 394
GRP_Box07, 394
GRP_Box08, 394
GRP_Box09, 394
GRP_Box10, 394
GRP_Box11, 394

GRP_Box12, 394
GRP_Box13, 394
GRP_Box14, 394
GRP_Box15, 394
GRP_Box16, 394
GRP_Box17, 394
GRP_Box18, 394
GRP_Box19, 394
GRP_Box20, 394
GRP_Box21, 394
GRP_Box22, 394
GRP_Box23, 394
GRP_Box24, 394
GRP_Box25, 394
GRP_Box26, 394
GRP_Box27, 394
GRP_Box28, 394
GRP_Box29, 394
GRP_Box30, 394
GRP_Box31, 394
GRP_Box32, 394
GRP_Box33, 394
GRP_Box34, 394
GRP_Box35, 394
GRP_Box36, 394
GRP_Box37, 394
GRP_Cylinder01, 394
GRP_Cylinder02, 394
GRP_Cylinder03, 394
GRP_Cylinder04, 394
GRP_Cylinder05, 394
GRP_Cylinder06, 394
GRP_Cylinder07, 394
GRP_Cylinder08, 394
GRP_Cylinder09, 394
GRP_Cylinder10, 394
GRP_Cylinder100, 394
GRP_Cylinder101, 394
GRP_Cylinder102, 394
GRP_Cylinder103, 394
GRP_Cylinder104, 394
GRP_Cylinder105, 394
GRP_Cylinder106, 394
GRP_Cylinder107, 394
GRP_Cylinder108, 394
GRP_Cylinder109, 394
GRP_Cylinder11, 394
GRP_Cylinder110, 394
GRP_Cylinder111, 394
GRP_Cylinder112, 394
GRP_Cylinder113, 394
GRP_Cylinder114, 394
GRP_Cylinder115, 394
GRP_Cylinder116, 394
GRP_Cylinder117, 394
GRP_Cylinder118, 394
GRP_Cylinder119, 394
GRP_Cylinder12, 394
GRP_Cylinder120, 394
GRP_Cylinder121, 394
GRP_Cylinder122, 394
GRP_Cylinder123, 394
GRP_Cylinder13, 394
GRP_Cylinder14, 394
GRP_Cylinder15, 394
GRP_Cylinder16, 394
GRP_Cylinder17, 394
GRP_Cylinder18, 394
GRP_Cylinder19, 394
GRP_Cylinder20, 394
GRP_Cylinder21, 394
GRP_Cylinder22, 394
GRP_Cylinder23, 394
GRP_Cylinder24, 394
GRP_Cylinder25, 394
GRP_Cylinder26, 394
GRP_Cylinder27, 394
GRP_Cylinder28, 394
GRP_Cylinder29, 394
GRP_Cylinder30, 394
GRP_Cylinder31, 394
GRP_Cylinder32, 394
GRP_Cylinder33, 394
GRP_Cylinder34, 394
GRP_Cylinder35, 394
GRP_Cylinder36, 394
GRP_Cylinder37, 394
GRP_Cylinder38, 394
GRP_Cylinder39, 394
GRP_Cylinder40, 394
GRP_Cylinder41, 394
GRP_Cylinder42, 394
GRP_Cylinder43, 394
GRP_Cylinder44, 394
GRP_Cylinder45, 394
GRP_Cylinder46, 394
GRP_Cylinder47, 394
GRP_Cylinder48, 394
GRP_Cylinder49, 394
GRP_Cylinder50, 394
GRP_Cylinder51, 394
GRP_Cylinder52, 394
GRP_Cylinder53, 394
GRP_Cylinder54, 394
GRP_Cylinder55, 394
GRP_Cylinder56, 394
GRP_Cylinder57, 394
GRP_Cylinder58, 394

GRP_Cylinder59, 394
GRP_Cylinder60, 394
GRP_Cylinder61, 394
GRP_Cylinder62, 394
GRP_Cylinder63, 394
GRP_Cylinder64, 394
GRP_Cylinder65, 394
GRP_Cylinder66, 394
GRP_Cylinder67, 394
GRP_Cylinder68, 394
GRP_Cylinder69, 394
GRP_Cylinder70, 394
GRP_Cylinder71, 394
GRP_Cylinder72, 394
GRP_Cylinder73, 394
GRP_Cylinder74, 394
GRP_Cylinder75, 394
GRP_Cylinder76, 394
GRP_Cylinder77, 394
GRP_Cylinder78, 394
GRP_Cylinder79, 394
GRP_Cylinder80, 394
GRP_Cylinder81, 394
GRP_Cylinder82, 394
GRP_Cylinder83, 394
GRP_Cylinder84, 394
GRP_Cylinder85, 394
GRP_Cylinder86, 394
GRP_Cylinder87, 394
GRP_Cylinder88, 394
GRP_Cylinder89, 394
GRP_Cylinder90, 394
GRP_Cylinder91, 394
GRP_Cylinder92, 394
GRP_Cylinder93, 394
GRP_Cylinder94, 394
GRP_Cylinder95, 394
GRP_Cylinder96, 394
GRP_Cylinder97, 394
GRP_Cylinder98, 394
GRP_Cylinder99, 394
GRP_FSS_WPS_Z_bracket, 394
GRP_FWD_RSS_drive_truck, 394
GRP_GOX_Cylinder06, 394
GRP_GOX_Cylinder342, 394
GRP_GOX_vent_arm_truss, 394
GRP_Inner_FSS_WPS_panel_track, 394
GRP_Inner_WPS_panel, 394
GRP_Lightning_mast, 394
GRP_Lightning_mast_foundation, 394
GRP_Line01, 394
GRP_Main_FSS, 394
GRP_MAPU_part_1, 394
GRP_MAPU_part_2, 394
GRP_North_GN2_vent_pipe, 394
GRP_Orbiter_Access_Arm, 394
GRP_Outer_WPS_panel, 394
GRP_Rear_RSS_drive_truck, 394
GRP_RSS, 394
GRP_RSS_cab, 394
GRP_RSS_misc, 394
GRP_RSS_sheild, 394
GRP_South_GN2_vent_pipe, 394
GRP_Vent_hood, 394
GRP_White_Room_Mat, 394
GRP_White_Room_Part_1, 394
GRP_White_Room_Part_2, 394
GRP_WPS_panel, 394
GRP_WPS_panels_Y_support_struts, 394
GRP_Y_FSS_WPS_struts, 394
NGRP, 394
NMAT, 394
NTEX, 394
meshres_ku.h
GRP_base_KU, 417
GRP_cable_one_KU, 417
GRP_KUBAND_BOX_KU, 417
GRP_KUDISH_KU, 417
GRP_KUGIMBAL_KU, 417
NGRP_KU, 417
NMAT_KU, 417
NTEX_KU, 417
meshres_odsvc.h
GRP_A7A3CB10_ODSVC, 422
GRP_A7A3CB11_ODSVC, 422
GRP_A7A3CB12_ODSVC, 422
GRP_A7A3CB13_ODSVC, 422
GRP_A7A3CB14_ODSVC, 422
GRP_A7A3CB15_ODSVC, 422
GRP_A7A3CB16_ODSVC, 422
GRP_A7A3CB1_ODSVC, 422
GRP_A7A3CB2_ODSVC, 422
GRP_A7A3CB3_ODSVC, 422
GRP_A7A3CB4_ODSVC, 422
GRP_A7A3CB5_ODSVC, 422
GRP_A7A3CB6_ODSVC, 422
GRP_A7A3CB7_ODSVC, 422
GRP_A7A3CB8_ODSVC, 422
GRP_A7A3CB9_ODSVC, 422
GRP_A7A3Guard_ODSVC, 422
GRP_A7A3panel_ODSVC, 422
GRP_A7A3S10_ODSVC, 422
GRP_A7A3S11_ODSVC, 422
GRP_A7A3S12_ODSVC, 422
GRP_A7A3S13_ODSVC, 422
GRP_A7A3S14_ODSVC, 422
GRP_A7A3S15_ODSVC, 422
GRP_A7A3S16_ODSVC, 422

GRP_A7A3S1_ODSVC, 422
GRP_A7A3S2_ODSVC, 422
GRP_A7A3S3_ODSVC, 422
GRP_A7A3S4_ODSVC, 422
GRP_A7A3S5_ODSVC, 422
GRP_A7A3S6_ODSVC, 422
GRP_A7A3S7_ODSVC, 422
GRP_A7A3S8_ODSVC, 422
GRP_A7A3S9_ODSVC, 422
GRP_A8A3panel_ODSVC, 422
GRP_A8A3S10_ODSVC, 422
GRP_A8A3S11_ODSVC, 422
GRP_A8A3S12_ODSVC, 422
GRP_A8A3S1_ODSVC, 422
GRP_A8A3S2_ODSVC, 422
GRP_A8A3S3_ODSVC, 422
GRP_A8A3S4_ODSVC, 422
GRP_A8A3S5_ODSVC, 422
GRP_A8A3S6_ODSVC, 422
GRP_A8A3S7_ODSVC, 422
GRP_A8A3S8_ODSVC, 422
GRP_A8A3S9_ODSVC, 422
GRP_Act_Hooks_Firing_ODSVC, 422
GRP_ADS_ODSVC, 422
GRP_APDS_Circ_Prot_OFF_ODSVC, 422
GRP_APDSCC_APDS_Circ_Prot_OFF_-ODSVC, 422
GRP_APDSCC_CloseHooks_ODSVC, 422
GRP_APDSCC_CloseLatches_ODSVC, 422
GRP_APDSCC_FixerOff_ODSVC, 422
GRP_APDSCC_PowerOff_ODSVC, 422
GRP_APDSCC_PowerOn_ODSVC, 422
GRP_APDSCC_RingIn_ODSVC, 422
GRP_APDSCC_RingOut_ODSVC, 422
GRP_BDS_ODSVC, 422
GRP_CAPTURE_ODSVC, 422
GRP_CDS_ODSVC, 422
GRP_Fixers_OFF_ODSVC, 422
GRP_guards_ODSVC, 422
GRP_Hooks1_CLOSED_ODSVC, 422
GRP_Hooks1_OPEN_ODSVC, 422
GRP_Hooks2_CLOSED_ODSVC, 422
GRP_Hooks2_OPEN_ODSVC, 422
GRP_Initial_Contact_ODSVC, 422
GRP_Interp_Sealed_ODSVC, 422
GRP_LampTest_ODSVC, 422
GRP_Latches_CLOSED_ODSVC, 422
GRP_Latches_OPEN_ODSVC, 422
GRP_OPEN_Hooks_ODSVC, 422
GRP_OPEN_Latches_ODSVC, 422
GRP_Pas_Hooks_Firing_ODSVC, 422
GRP_plateguard1_ODSVC, 422
GRP_plateguard2_ODSVC, 422
GRP_Power_ON_ODSVC, 422
GRP_Pyro_Circ_Prot_OFF_ODSVC, 422
GRP_Pyro_Circ_Prot_ON_ODSVC, 422
GRP_READY_To_Hook_ODSVC, 422
GRP_Ring_Aligned_ODSVC, 422
GRP_Ring_Final_Pos_ODSVC, 422
GRP_Ring_Forward_Pos_ODSVC, 422
GRP_Ring_Initial_Pos_ODSVC, 422
GRP_Undock_Complet_ODSVC, 422
GRP_Undocking_ODSVC, 422
NGRPODSVC, 422
NMATODSVC, 422
NTEXODSVC, 422
meshres_RMS.h
GRP_base, 424
GRP_cambase, 424
GRP_camswivel, 424
GRP_elbowcam, 424
GRP_Humerus, 424
GRP_Radii, 424
GRP_RMS_MPMs, 424
GRP_Shoulder_Yaw, 424
GRP_Wrist_Yaw, 424
GRP_Wristpitch, 424
NGRP_RMS, 424
NMAT_RMS, 424
NTEX_RMS, 424
meshres_vc.h
GRP_080_VC, 442
GRP_081_VC, 442
GRP_082_VC, 442
GRP_146_VC, 442
GRP_147_VC, 442
GRP_3_PRPLT_LEFT1_VC, 442
GRP_A1_VC, 442
GRP_A1Guard_VC, 442
GRP_A1Lpanel_VC, 442
GRP_A1LRS6_VC, 442
GRP_A1LS11_VC, 442
GRP_A1LS12_VC, 442
GRP_A1LS13_VC, 442
GRP_A1LS14_VC, 442
GRP_A1LS15_VC, 442
GRP_A1LS16_VC, 442
GRP_A1LS17_VC, 442
GRP_A1LS18_VC, 442
GRP_A1LS19_VC, 442
GRP_A1LS1_VC, 442
GRP_A1LS20_VC, 442
GRP_A1LS21_VC, 442
GRP_A1LS22_VC, 442
GRP_A1LS23_VC, 442
GRP_A1LS24_VC, 442
GRP_A1LS25_VC, 442
GRP_A1LS26_VC, 442

GRP_A1LS2_VC, 442
GRP_A1LS4_VC, 442
GRP_A1LS5_VC, 442
GRP_A1LS8_VC, 442
GRP_A1LS9_VC, 442
GRP_A1R_VC, 442
GRP_A1RRS10_VC, 442
GRP_A1RRS14_VC, 442
GRP_A1RRS15_VC, 442
GRP_A1RRS16_VC, 442
GRP_A1RRS4_VC, 442
GRP_A1RRS8_VC, 442
GRP_A1RS11_VC, 442
GRP_A1RS12_VC, 442
GRP_A1RS13_VC, 442
GRP_A1RS17_VC, 442
GRP_A1RS18_VC, 442
GRP_A1RS19_VC, 442
GRP_A1RS1_VC, 442
GRP_A1RS20_VC, 442
GRP_A1RS21_VC, 442
GRP_A1RS22_VC, 442
GRP_A1RS23_VC, 442
GRP_A1RS24_VC, 442
GRP_A1RS25_VC, 442
GRP_A1RS26_VC, 442
GRP_A1RS27_VC, 442
GRP_A1RS28_VC, 442
GRP_A1RS29_VC, 442
GRP_A1RS2_VC, 442
GRP_A1RS30_VC, 442
GRP_A1RS31_VC, 442
GRP_A1RS32_VC, 442
GRP_A1RS33_VC, 442
GRP_A1RS34_VC, 442
GRP_A1RS35_VC, 442
GRP_A1RS3_VC, 442
GRP_A1RS5_VC, 442
GRP_A1RS6_VC, 442
GRP_A1RS7_VC, 442
GRP_A1RS9_VC, 442
GRP_A1U_VC, 442
GRP_A1UM1_VC, 442
GRP_A1URS15_VC, 442
GRP_A1URS16_VC, 442
GRP_A1URS7_VC, 442
GRP_A1US10_VC, 442
GRP_A1US12_VC, 442
GRP_A1US13_VC, 442
GRP_A1US14_VC, 442
GRP_A1US18_VC, 442
GRP_A1US1_VC, 442
GRP_A1US2_VC, 442
GRP_A1US3_VC, 442
GRP_A1US4_VC, 442
GRP_A1US6_VC, 442
GRP_A1US8_VC, 442
GRP_A2AZH_VC, 442
GRP_A2AZV_VC, 442
GRP_A2ELH_VC, 442
GRP_A2ELV_VC, 442
GRP_A2panel_VC, 442
GRP_A2U1_VC, 442
GRP_A2U2_VC, 442
GRP_A4_Event_M10_VC, 442
GRP_A4_Event_M_VC, 442
GRP_A4_Event_S10_VC, 442
GRP_A4_Event_S_VC, 442
GRP_A4_MET_DAY1_VC, 442
GRP_A4_MET_DAY2_VC, 442
GRP_A4_MET_DAY3_VC, 442
GRP_A4_MET_H1_VC, 442
GRP_A4_MET_H2_VC, 442
GRP_A4_MET_M1_VC, 442
GRP_A4_MET_M2_VC, 442
GRP_A4_MET_S1_VC, 442
GRP_A4_MET_S2_VC, 442
GRP_A4MissionSwitch_VC, 442
GRP_A6U1_VC, 442
GRP_A6U2_VC, 442
GRP_A6U32_VC, 442
GRP_A6U33_VC, 442
GRP_A6U34_VC, 442
GRP_A6U35_VC, 442
GRP_A6U36_VC, 442
GRP_A6U39_VC, 442
GRP_A6U40_VC, 442
GRP_A6U41_VC, 442
GRP_A6U42_VC, 442
GRP_A6U43_VC, 442
GRP_A6U44_VC, 442
GRP_A6U45_VC, 442
GRP_A6U46_VC, 442
GRP_A6U47_VC, 442
GRP_A6U5_VC, 442
GRP_A6U6_VC, 442
GRP_A6U7_VC, 442
GRP_A6Upanel_VC, 442
GRP_A6URS1_VC, 442
GRP_A6URS2_VC, 442
GRP_A6URS3_VC, 442
GRP_A6URS4_VC, 442
GRP_A6URS5_VC, 442
GRP_A6US37_VC, 442
GRP_A6US4_VC, 442
GRP_A6Utog1_VC, 442
GRP_A6Utog2_VC, 442
GRP_A6Utog3_VC, 442

GRP_A6Utog4_VC, 442
GRP_A7U10_VC, 442
GRP_A7U11_VC, 442
GRP_A7U12_VC, 442
GRP_A7U13_VC, 442
GRP_A7U14_VC, 442
GRP_A7U15_VC, 442
GRP_A7U16_VC, 442
GRP_A7U17_VC, 442
GRP_A7U18_VC, 442
GRP_A7U1_VC, 442
GRP_A7U2_VC, 442
GRP_A7U3_VC, 442
GRP_A7U40_VC, 442
GRP_A7U41_VC, 442
GRP_A7U42_VC, 442
GRP_A7U43_VC, 442
GRP_A7U44_VC, 442
GRP_A7U4_VC, 442
GRP_A7U52_VC, 442
GRP_A7U53_VC, 442
GRP_A7U54_VC, 442
GRP_A7U56_VC, 442
GRP_A7U5_VC, 442
GRP_A7U6_VC, 442
GRP_A7U7_VC, 442
GRP_A7U8_VC, 442
GRP_A7U9_VC, 442
GRP_A8bot_VC, 442
GRP_A8Guardb_VC, 442
GRP_A8top_VC, 442
GRP_ACK_VC, 442
GRP_aft_VC, 442
GRP_AftMonitor_VC, 442
GRP_B_VC, 442
GRP_bottomcup_VC, 442
GRP_bottomdial_VC, 442
GRP_C2_VC, 442
GRP_C2b10_VC, 442
GRP_C2b11_VC, 442
GRP_C2b1_VC, 442
GRP_C2b2_VC, 442
GRP_C2b3_VC, 442
GRP_C2b4_VC, 442
GRP_C2b5_VC, 442
GRP_C2b6_VC, 442
GRP_C2b7v_VC, 442
GRP_C2b8v_VC, 442
GRP_C2b9_VC, 442
GRP_C2guards_VC, 442
GRP_C2tog1_VC, 442
GRP_C2tog2_VC, 442
GRP_C2tog3_VC, 442
GRP_C2tog4_VC, 442
GRP_C3_VC, 442
GRP_C3b10_VC, 442
GRP_C3b11_VC, 442
GRP_C3b12_VC, 442
GRP_C3b13_VC, 442
GRP_C3b14v_VC, 442
GRP_C3b15_VC, 442
GRP_C3b16_VC, 442
GRP_C3b17_VC, 442
GRP_C3b18v_VC, 442
GRP_C3b19_VC, 442
GRP_C3b1_VC, 442
GRP_C3b20_VC, 442
GRP_C3b21_VC, 442
GRP_C3b22_VC, 442
GRP_C3b23_VC, 442
GRP_C3b24_VC, 442
GRP_C3b25_VC, 442
GRP_C3b26_VC, 442
GRP_C3b27_VC, 442
GRP_C3b28_VC, 442
GRP_C3b29_VC, 442
GRP_C3b2_VC, 442
GRP_C3b30_VC, 442
GRP_C3b31_VC, 442
GRP_C3b32_VC, 442
GRP_C3b33_VC, 442
GRP_C3b34_VC, 442
GRP_C3b35_VC, 442
GRP_C3b36_VC, 442
GRP_C3b37_VC, 442
GRP_C3b3_VC, 442
GRP_C3b4_VC, 442
GRP_C3b5_VC, 442
GRP_C3b6_VC, 442
GRP_C3b7_VC, 442
GRP_C3b8_VC, 442
GRP_C3b9_VC, 442
GRP_C3cover1_VC, 442
GRP_C3cover2_VC, 442
GRP_C3cover3_VC, 442
GRP_C3cover4_VC, 442
GRP_C3cover5_VC, 442
GRP_C3Dial_VC, 442
GRP_C_VC, 442
GRP_cable_VC, 442
GRP_CDR1_VC, 442
GRP_CDR1LABEL_VC, 442
GRP_CDR2_VC, 442
GRP_CDR2LABEL_VC, 442
GRP_clear_VC, 442
GRP_Clock1_VC, 442
GRP_Clock2_VC, 442
GRP_Clock3_VC, 442

GRP_Clock4_VC, 442
GRP_Clock5_VC, 442
GRP_Clock6_VC, 442
GRP_Clock7_VC, 442
GRP_Clock8_VC, 442
GRP_Coas_VC, 442
GRP_covers_VC, 442
GRP_D_VC, 442
GRP_Dial_VC, 442
GRP_Dialsmall_VC, 442
GRP_E_VC, 442
GRP_eight_VC, 442
GRP_Eventtimer_VC, 442
GRP_F3S1_VC, 442
GRP_F3S2_VC, 442
GRP_F3S3_VC, 442
GRP_F3S4_VC, 442
GRP_F3S5_VC, 442
GRP_F3S6_VC, 442
GRP_F6button1_VC, 442
GRP_F6button2_VC, 442
GRP_F6button3_VC, 442
GRP_F6button4_VC, 442
GRP_F6button5_VC, 442
GRP_F6button6_VC, 442
GRP_F6button7_VC, 442
GRP_F6cover1_VC, 442
GRP_F6cover2_VC, 442
GRP_F6Dial_VC, 442
GRP_F6Guard_VC, 442
GRP_F6HUDbrt_VC, 442
GRP_F6HUDdim_VC, 442
GRP_F6HUDtest_VC, 442
GRP_F6panel1_VC, 442
GRP_F6panel_VC, 442
GRP_F6switch_1_VC, 442
GRP_F6switch_2_VC, 442
GRP_F6switch_3_VC, 442
GRP_F6switch_4_VC, 442
GRP_F6switch_5_VC, 442
GRP_F6switch_6_VC, 442
GRP_F6switch_7_VC, 442
GRP_F6switch_8_VC, 442
GRP_F6switch_9_VC, 442
GRP_F6TalkBack1_VC, 442
GRP_F6TalkBack2_VC, 442
GRP_F6TalkBack3_VC, 442
GRP_F7_Event_M10_VC, 442
GRP_F7_Event_M_VC, 442
GRP_F7_Event_S10_VC, 442
GRP_F7_Event_S_VC, 442
GRP_F7CW_VC, 442
GRP_F8button1_VC, 442
GRP_F8button2_VC, 442
GRP_F8button3_VC, 442
GRP_F8cover1_VC, 442
GRP_F8cover2_VC, 442
GRP_F8Guard_VC, 442
GRP_F8HUDbrt_VC, 442
GRP_F8HUDdim_VC, 442
GRP_F8HUDtest_VC, 442
GRP_F8pan1_VC, 442
GRP_F8panel_VC, 442
GRP_F8switch1_VC, 442
GRP_F8switch2_VC, 442
GRP_F8switch3_VC, 442
GRP_F8switch4_VC, 442
GRP_F8switch5_VC, 442
GRP_F8switch6_VC, 442
GRP_F8switch7_VC, 442
GRP_F8switch8_VC, 442
GRP_F8switch9_VC, 442
GRP_F8TalkBack1_VC, 442
GRP_F8TalkBack2_VC, 442
GRP_F8TalkBack3_VC, 442
GRP_F9needle1_VC, 442
GRP_F9needle2_VC, 442
GRP_F9needle3_VC, 442
GRP_F9panel_VC, 442
GRP_Fault_VC, 442
GRP_Fcovers_VC, 442
GRP_five_VC, 442
GRP_GPC_Status_VC, 442
GRP_GPC_VC, 442
GRP_Guardb_VC, 442
GRP_guards_VC, 442
GRP_invisiblebottom_VC, 442
GRP_invisibletop_VC, 442
GRP_IORESET_VC, 442
GRP_ITEM_VC, 442
GRP_KEYB_A_VC, 442
GRP_KEYB_DOT_VC, 442
GRP_KEYB_EXEC_VC, 442
GRP_KEYB_F_VC, 442
GRP_KEYB_FOUR_VC, 442
GRP_KEYB_ONE_VC, 442
GRP_KEYB_OPS_VC, 442
GRP_KEYB_RESUME_VC, 442
GRP_KEYB_SPEC_VC, 442
GRP_KEYB_TWO_VC, 442
GRP_L1b10_VC, 442
GRP_L1b11_VC, 442
GRP_L1b12_VC, 442
GRP_L1b13_VC, 442
GRP_L1b14_VC, 442
GRP_L1b15_VC, 442
GRP_L1b16_VC, 442
GRP_L1b17_VC, 442

GRP_L1b18_VC, 442
GRP_L1b19_VC, 442
GRP_L1b1_VC, 442
GRP_L1b20_VC, 442
GRP_L1b21_VC, 442
GRP_L1b22_VC, 442
GRP_L1b23_VC, 442
GRP_L1b24_VC, 442
GRP_L1b25_VC, 442
GRP_L1b26_VC, 442
GRP_L1b27_VC, 442
GRP_L1b28_VC, 442
GRP_L1b29_VC, 442
GRP_L1b2_VC, 442
GRP_L1b30_VC, 442
GRP_L1b31_VC, 442
GRP_L1b32_VC, 442
GRP_L1b33_VC, 442
GRP_L1b34_VC, 442
GRP_L1b35_VC, 442
GRP_L1b36_VC, 442
GRP_L1b37_VC, 442
GRP_L1b38_VC, 442
GRP_L1b39_VC, 442
GRP_L1b3_VC, 442
GRP_L1b40_VC, 442
GRP_L1b41_VC, 442
GRP_L1b42_VC, 442
GRP_L1b43_VC, 442
GRP_L1b44_VC, 442
GRP_L1b45_VC, 442
GRP_L1b46_VC, 442
GRP_L1b47_VC, 442
GRP_L1b4_VC, 442
GRP_L1b5_VC, 442
GRP_L1b6_VC, 442
GRP_L1b7_VC, 442
GRP_L1b8_VC, 442
GRP_L1b9_VC, 442
GRP_L1cover1_VC, 442
GRP_L1cover2_VC, 442
GRP_L1cover3_VC, 442
GRP_L1dial1_VC, 442
GRP_L1dial2_VC, 442
GRP_L1dial3_VC, 442
GRP_L1Panel_VC, 442
GRP_L2b10_VC, 442
GRP_L2b11v_VC, 442
GRP_L2b12_VC, 442
GRP_L2b13_VC, 442
GRP_L2b14_VC, 442
GRP_L2b15_VC, 442
GRP_L2b16_VC, 442
GRP_L2b17v_VC, 442
GRP_L2b18_VC, 442
GRP_L2b19_VC, 442
GRP_L2b1_VC, 442
GRP_L2b20_VC, 442
GRP_L2b21_VC, 442
GRP_L2b22_VC, 442
GRP_L2b23_VC, 442
GRP_L2b24_VC, 442
GRP_L2b25_VC, 442
GRP_L2b2_VC, 442
GRP_L2b3_VC, 442
GRP_L2b4_VC, 442
GRP_L2b5_VC, 442
GRP_L2b6_VC, 442
GRP_L2b7_VC, 442
GRP_L2b8_VC, 442
GRP_L2b9_VC, 442
GRP_L2panel_VC, 442
GRP_leftface_VC, 442
GRP_MDUs_VC, 442
GRP_measure_strips_VC, 442
GRP_MFD11_VC, 442
GRP_MFD1_VC, 442
GRP_MFD1LABEL_VC, 442
GRP_MFD2_VC, 442
GRP_MFD2LABEL_VC, 442
GRP_MFD3_VC, 442
GRP_MFD3LABEL_VC, 442
GRP_MFD4_VC, 442
GRP_MFD4LABEL_VC, 442
GRP_MFD5_VC, 442
GRP_MFD5LABEL_VC, 442
GRP_MFD_aft_VC, 442
GRP_MFD_VC, 442
GRP_minus_VC, 442
GRP_MSG_VC, 442
GRP_nine_VC, 442
GRP_O1_VC, 442
GRP_O1M1P_VC, 442
GRP_O1M1T_VC, 442
GRP_O1M2F_VC, 442
GRP_O1M2T_VC, 442
GRP_O1M3F_VC, 442
GRP_O1M3P_VC, 442
GRP_O1M4P_VC, 442
GRP_O1M4ppo_VC, 442
GRP_O1S1_VC, 442
GRP_O1S2_VC, 442
GRP_O1S3_VC, 442
GRP_O1S4_VC, 442
GRP_O1S5_VC, 442
GRP_O1S6_VC, 442
GRP_O2_VC, 442
GRP_O2M5T_VC, 442

GRP_O2M6P_VC, 442
GRP_O2M6Q_VC, 442
GRP_O2M7P_VC, 442
GRP_O2M7Q_VC, 442
GRP_O2M8T_VC, 442
GRP_O2S7_VC, 442
GRP_O2S8_VC, 442
GRP_O2S9_VC, 442
GRP_O3_guards_VC, 442
GRP_O3_MET_DAY1_VC, 442
GRP_O3_MET_DAY2_VC, 442
GRP_O3_MET_DAY3_VC, 442
GRP_O3_MET_H1_VC, 442
GRP_O3_MET_H2_VC, 442
GRP_O3_MET_M1_VC, 442
GRP_O3_MET_M2_VC, 442
GRP_O3_MET_S1_VC, 442
GRP_O3_MET_S2_VC, 442
GRP_O3_PRPLT_FWD1_VC, 442
GRP_O3_PRPLT_FWD2_VC, 442
GRP_O3_PRPLT_LEFT2_VC, 442
GRP_O3_PRPLT_RIGHT1_VC, 442
GRP_O3_PRPLT_RIGHT2_VC, 442
GRP_O3_VC, 442
GRP_O3M10F_VC, 442
GRP_O3M10O_VC, 442
GRP_O3M11F_VC, 442
GRP_O3M11O_VC, 442
GRP_O3M9F_VC, 442
GRP_O3M9O_VC, 442
GRP_O3S10_VC, 442
GRP_O3S11_VC, 442
GRP_O3S12_VC, 442
GRP_O6Guard_VC, 442
GRP_O6panel_VC, 442
GRP_O6R1_VC, 442
GRP_O6R2_VC, 442
GRP_O6R3_VC, 442
GRP_O6R4_VC, 442
GRP_O6R5_VC, 442
GRP_O6R6_VC, 442
GRP_O6R7_VC, 442
GRP_O6RS6_VC, 442
GRP_O6S12_VC, 442
GRP_O6S13_VC, 442
GRP_O6S14_VC, 442
GRP_O6S15_VC, 442
GRP_O6S16_VC, 442
GRP_O6S17_VC, 442
GRP_O6S18_VC, 442
GRP_O6S19_VC, 442
GRP_O6S1_VC, 442
GRP_O6S20_VC, 442
GRP_O6S21_VC, 442
GRP_O6S22_VC, 442
GRP_O6S23_VC, 442
GRP_O6S24_VC, 442
GRP_O6S25_VC, 442
GRP_O6S26_VC, 442
GRP_O6S27_VC, 442
GRP_O6S28_VC, 442
GRP_O6S29_VC, 442
GRP_O6S2_VC, 442
GRP_O6S30_VC, 442
GRP_O6S31_VC, 442
GRP_O6S32_VC, 442
GRP_O6S33_VC, 442
GRP_O6S34_VC, 442
GRP_O6S35_VC, 442
GRP_O6S36_VC, 442
GRP_O6S37_VC, 442
GRP_O6S38_VC, 442
GRP_O6S39_VC, 442
GRP_O6S3_VC, 442
GRP_O6S45_VC, 442
GRP_O6S46_VC, 442
GRP_O6S47_VC, 442
GRP_O6S48_VC, 442
GRP_O6S49_VC, 442
GRP_O6S4_VC, 442
GRP_O6S50_VC, 442
GRP_O6S51_VC, 442
GRP_O6S52_VC, 442
GRP_O6S5_VC, 442
GRP_O6S6_VC, 442
GRP_O6switchA_VC, 442
GRP_O6switchB_VC, 442
GRP_O6switchC_VC, 442
GRP_O7Guard_VC, 442
GRP_O7panel_VC, 442
GRP_O7RS1_VC, 442
GRP_O7RS3_VC, 442
GRP_O7RS5_VC, 442
GRP_O7S10_VC, 442
GRP_O7S11_VC, 442
GRP_O7S13_VC, 442
GRP_O7S14_VC, 442
GRP_O7S16a_VC, 442
GRP_O7S16b_VC, 442
GRP_O7S17a_VC, 442
GRP_O7S17b_VC, 442
GRP_O7S18a_VC, 442
GRP_O7S18b_VC, 442
GRP_O7S22_VC, 442
GRP_O7S23_VC, 442
GRP_O7S24_VC, 442
GRP_O7S25_VC, 442
GRP_O7S26_VC, 442

GRP_O7S27_VC, 442
GRP_O7S28_VC, 442
GRP_O7S29_VC, 442
GRP_O7S2_VC, 442
GRP_O7S30_VC, 442
GRP_O7S31_VC, 442
GRP_O7S32_VC, 442
GRP_O7S33_VC, 442
GRP_O7S34_VC, 442
GRP_O7S35_VC, 442
GRP_O7S36_VC, 442
GRP_O7S4_VC, 442
GRP_O7S6_VC, 442
GRP_O8Guard_VC, 442
GRP_O8panel_VC, 442
GRP_O8R1_VC, 442
GRP_O8R2_VC, 442
GRP_O8R3_VC, 442
GRP_O8R4_VC, 442
GRP_O8R5_VC, 442
GRP_O8R6_VC, 442
GRP_O8S10_VC, 442
GRP_O8S12_VC, 442
GRP_O8S13_VC, 442
GRP_O8S14_VC, 442
GRP_O8S15_VC, 442
GRP_O8S16_VC, 442
GRP_O8S17_VC, 442
GRP_O8S18_VC, 442
GRP_O8S19_VC, 442
GRP_O8S1_VC, 442
GRP_O8S1a_VC, 442
GRP_O8S20_VC, 442
GRP_O8S21_VC, 442
GRP_O8S22_VC, 442
GRP_O8S23_VC, 442
GRP_O8S24_VC, 442
GRP_O8S26_VC, 442
GRP_O8S27_VC, 442
GRP_O8S28_VC, 442
GRP_O8S29_VC, 442
GRP_O8S2a_VC, 442
GRP_O8S30_VC, 442
GRP_O8S31_VC, 442
GRP_O8S32_VC, 442
GRP_O8S33_VC, 442
GRP_O8S34_VC, 442
GRP_O8S3_VC, 442
GRP_O8S4_VC, 442
GRP_O8S5_VC, 442
GRP_O8S6_VC, 442
GRP_O8S7_VC, 442
GRP_O8S8_VC, 442
GRP_O8S9_VC, 442
GRP_panelA4_VC, 442
GRP_panelR4_VC, 442
GRP_PLT1_VC, 442
GRP_PLT1LABEL_VC, 442
GRP_PLT2_VC, 442
GRP_PLT2LABEL_VC, 442
GRP_plus_VC, 442
GRP_PRO_VC, 442
GRP_R13L_VC, 442
GRP_R13talk1_VC, 442
GRP_R13talk2_VC, 442
GRP_R13talk3_VC, 442
GRP_R13talk4_VC, 442
GRP_R13talk5_VC, 442
GRP_R13talk6_VC, 442
GRP_R13talk7_VC, 442
GRP_R13talk8_VC, 442
GRP_R1b1_VC, 442
GRP_R1b2_VC, 442
GRP_R1b3_VC, 442
GRP_R1b4_VC, 442
GRP_R1b5_VC, 442
GRP_R2b10_VC, 442
GRP_R2b11_VC, 442
GRP_R2b12_VC, 442
GRP_R2b13_VC, 442
GRP_R2b14_VC, 442
GRP_R2b15_VC, 442
GRP_R2b16_VC, 442
GRP_R2b17_VC, 442
GRP_R2b18_VC, 442
GRP_R2b19_VC, 442
GRP_R2b1_VC, 442
GRP_R2b20_VC, 442
GRP_R2b21_VC, 442
GRP_R2b22_VC, 442
GRP_R2b23_VC, 442
GRP_R2b24_VC, 442
GRP_R2b25_VC, 442
GRP_R2b26_VC, 442
GRP_R2b27_VC, 442
GRP_R2b28_VC, 442
GRP_R2b29_VC, 442
GRP_R2b2_VC, 442
GRP_R2b30_VC, 442
GRP_R2b31_VC, 442
GRP_R2b32_VC, 442
GRP_R2b33_VC, 442
GRP_R2b34_VC, 442
GRP_R2b35_VC, 442
GRP_R2b36_VC, 442
GRP_R2b37_VC, 442
GRP_R2b38_VC, 442
GRP_R2b39_VC, 442

GRP_R2b3_VC, 442
GRP_R2b40_VC, 442
GRP_R2b41_VC, 442
GRP_R2b42_VC, 442
GRP_R2b43_VC, 442
GRP_R2b44_VC, 442
GRP_R2b45_VC, 442
GRP_R2b46_VC, 442
GRP_R2b47_VC, 442
GRP_R2b48_VC, 442
GRP_R2b49_VC, 442
GRP_R2b4_VC, 442
GRP_R2b50_VC, 442
GRP_R2b51_VC, 442
GRP_R2b52_VC, 442
GRP_R2b53_VC, 442
GRP_R2b54_VC, 442
GRP_R2b55_VC, 442
GRP_R2b56_VC, 442
GRP_R2b5_VC, 442
GRP_R2b6_VC, 442
GRP_R2b7_VC, 442
GRP_R2b8_VC, 442
GRP_R2b9_VC, 442
GRP_R2guards_VC, 442
GRP_R2panel_VC, 442
GRP_R4Guard_VC, 442
GRP_R4S10_VC, 442
GRP_R4S11_VC, 442
GRP_R4S12_VC, 442
GRP_R4S13_VC, 442
GRP_R4S14_VC, 442
GRP_R4S15_VC, 442
GRP_R4S16_VC, 442
GRP_R4S17_VC, 442
GRP_R4S18_VC, 442
GRP_R4S19_VC, 442
GRP_R4S1_VC, 442
GRP_R4S21_VC, 442
GRP_R4S22_VC, 442
GRP_R4S23_VC, 442
GRP_R4S24_VC, 442
GRP_R4S25_VC, 442
GRP_R4S26_VC, 442
GRP_R4S27_VC, 442
GRP_R4S28_VC, 442
GRP_R4S29_VC, 442
GRP_R4S2_VC, 442
GRP_R4S30_VC, 442
GRP_R4S3_VC, 442
GRP_R4S4_VC, 442
GRP_R4S5_VC, 442
GRP_R4S6_VC, 442
GRP_R4S7_VC, 442
GRP_R4S8_VC, 442
GRP_R4S9_VC, 442
GRP_rightface_VC, 442
GRP_seven_VC, 442
GRP_six_VC, 442
GRP_stripbases_VC, 442
GRP_SwitchR13L_10_VC, 442
GRP_SwitchR13L_11_VC, 442
GRP_SwitchR13L_12_VC, 442
GRP_SwitchR13L_13_VC, 442
GRP_SwitchR13L_1_VC, 442
GRP_SwitchR13L_2_VC, 442
GRP_SwitchR13L_3_VC, 442
GRP_SwitchR13L_4_VC, 442
GRP_SwitchR13L_5_VC, 442
GRP_SwitchR13L_6_VC, 442
GRP_SwitchR13L_7_VC, 442
GRP_SwitchR13L_8_VC, 442
GRP_SwitchR13L_9_VC, 442
GRP_Sys_VC, 442
GRP_three_VC, 442
GRP_topcup_VC, 442
GRP_topdial_VC, 442
GRP_topface_VC, 442
GRP_transboot_VC, 442
GRP_transguard_VC, 442
GRP_translational_VC, 442
GRP_transRMS_VC, 442
GRP_transRMSboot_VC, 442
GRP_VirtualHUD_VC, 442
GRP_zero_VC, 442
NGRP_VC, 442
NMAT_VC, 442
NTEX_VC, 442
meshres_vc_additions.h
GRP_A4_Event_M10_VC, 446
GRP_A4_Event_M_VC, 446
GRP_A4_Event_S10_VC, 446
GRP_A4_Event_S_VC, 446
GRP_A4_MET_DAY1_VC, 446
GRP_A4_MET_DAY2_VC, 446
GRP_A4_MET_DAY3_VC, 446
GRP_A4_MET_H1_VC, 446
GRP_A4_MET_H2_VC, 446
GRP_A4_MET_M1_VC, 446
GRP_A4_MET_M2_VC, 446
GRP_A4_MET_S1_VC, 446
GRP_A4_MET_S2_VC, 446
GRP_A4MissionSwitch_VC, 446
GRP_A8b10_VC, 446
GRP_A8b11_VC, 446
GRP_A8b12_VC, 446
GRP_A8b13_VC, 446
GRP_A8b14_VC, 446

GRP_A8b15_VC, 446
GRP_A8b16_VC, 446
GRP_A8b17_VC, 446
GRP_A8b1_VC, 446
GRP_A8b2_VC, 446
GRP_A8b3_VC, 446
GRP_A8b4_VC, 446
GRP_A8b5_VC, 446
GRP_A8b6_VC, 446
GRP_A8b7_VC, 446
GRP_A8b8_VC, 446
GRP_A8b9_VC, 446
GRP_C2b10_VC, 446
GRP_C2b12_VC, 446
GRP_C2b1_VC, 446
GRP_C2b2_VC, 446
GRP_C2b3_VC, 446
GRP_C2b4_VC, 446
GRP_C2b5_VC, 446
GRP_C2b6_VC, 446
GRP_C2b7_VC, 446
GRP_C2b8_VC, 446
GRP_C2b9_VC, 446
GRP_C2Evt10MW_VC, 446
GRP_C2Evt10SW_VC, 446
GRP_C2Evt1MW_VC, 446
GRP_C2Evt1SW_VC, 446
GRP_Eventtimer_VC, 446
GRP_F7_Event_M10_VC, 446
GRP_F7_Event_M_VC, 446
GRP_F7_Event_S10_VC, 446
GRP_F7_Event_S_VC, 446
GRP_panelA4_VC, 446
TEX_A1SHADOW_VC, 446
TEX_A8BOTTOM_VC, 446
TEX_A8TOP_VC, 446
TEX_AFTTIMER_VC, 446
TEX_ATLANTIS, 446
TEX_CLOCKNUMS_VC, 446
TEX_COAS_VC, 446
TEX_DIGITS_VC, 446
TEX_FPANELS_VC, 446
TEX_GPC_VC, 446
TEX_LABEL_VC, 446
TEX_PANELA1L_VC, 446
TEX_PANELA1R_VC, 446
TEX_PANELA1U_VC, 446
TEX_PANELC2_VC, 446
TEX_PANELC3_VC, 446
TEX_PANELF6_VC, 446
TEX_PANELF8_VC, 446
TEX_PANELO1_VC, 446
TEX_PANELO3_VC, 446
TEX_PANELR2_VC, 446
TEX_R13L_VC, 446
TEX_R1L1_VC, 446
TEX_STSVC01_VC, 446
TEX_STSVC02_VC, 446
TEX_STSVC03_VC, 446
TEX_TALKBACK_VC, 446
TEX_TIMER_VC, 446
MET
Atlantis, 58
CRT, 130
met
Atlantis, 58
MET_Add
Atlantis, 58
CRT, 130
metCutoff
Atlantis, 66
mfd
CRT.cpp, 352
MFD_GDIPARAM, 202
hCRTFont, 202
hDLL, 202
mfd_gparam
CRT.cpp, 352
mfdbright
Atlantis, 66
MFDGROUPS
Atlantis.h, 326
mfds
Atlantis, 66
mfdspec
vc::MDU, 198
MIDX_COCKPIT
Atlantis.h, 326
MIDX_ET
Atlantis.h, 328
MIDX_KU_ANTENNA
Atlantis.h, 328
MIDX_LSRB
Atlantis.h, 328
MIDX_MIDDECK
Atlantis.h, 328
MIDX_ODS
Atlantis.h, 328
MIDX_ODS_VC
Atlantis.h, 328
midx_odspanel
vc::PanelA7A8ODS, 229
MIDX_ORBITER
Atlantis.h, 328
MIDX_PLPANEL_A7
Atlantis.h, 328
MIDX_PLPANEL_A8
Atlantis.h, 328

MIDX_PLPANEL_A9
Atlantis.h, 328
MIDX_RMS
Atlantis.h, 328
MIDX_RSRB
Atlantis.h, 328
MIDX_VC
Atlantis.h, 328
midxODS
eva_docking::ODS, 214
Mission
mission::Mission, 204
mission, 20
mission::Mission, 203
~Mission, 204
bUseRMS, 204
CheckValidity, 204
CloneNewBranch, 204
CloneRevision, 204
fLTimeMJD, 204
GetMissionName, 204
GetOrbiter, 204
iBranchCode, 204
iETVersion, 204
iRevisionCode, 204
LoadMission, 204
Mission, 204
SaveMission, 204
strMissionName, 204
strOrbiter, 204
MLP, 205
~MLP, 206
ActivateSSS, 206
ahHDP, 207
bSSS_Active, 207
bStartSequence, 207
CalculateSteamProduction, 206
clbkLoadStateEx, 206
clbkLoadVC, 206
clbkPreStep, 206
clbkSaveState, 206
clbkSetClassCaps, 206
fCountdown, 207
fSRBSteam, 207
fSSMESteam, 207
fT_SSSActive, 207
MLP, 206
mshMLP, 207
SignalGSEStart, 206
TriggerHDP, 206
MLP.cpp
ExitModule, 448
InitModule, 448
ovcExit, 448
ovcInit, 448
MLP.h
DIR_MPS_SMOKE, 449
DIR_SRB_SMOKE, 449
HDP_POS, 449
POS_MPS_SMOKE, 449
POS_SRB_SMOKE, 449
MNVR
Atlantis, 66
CRT, 126
MNVR_COMPLETE
Atlantis, 51
MNVR_IN_PROGRESS
Atlantis, 51
MNVR_OFF
Atlantis, 51
MNVR_STARTING
Atlantis, 51
MNVR_OPTION
Atlantis, 66
MNVR_TIME
Atlantis, 66
MNVRLOAD
Atlantis, 66
mode
CRT, 130
CRT::SavePrm, 132
Move
BasicValve, 101
MP_OFF
PayloadBayOp, 263
MP_ON
PayloadBayOp, 263
MPL
mps::SSME, 283
MPM_Microswitches
Atlantis, 66
mpos
BasicValve, 101
MPS
mps::MPS, 209
mps, 21
_DCU, 22
_ENGINE_STATUS, 22
_MPS_VALVE, 23
BLOCK_II_FPL_THRUST, 23
BLOCK_II_ISP0, 23
BLOCK_II_ISP1, 23
BLOCK_II_RPL_THRUST, 23
CHECKOUT_COMPONENTCHECKOUT, 22
CHECKOUT_STANDBY, 22
DATA_TABLE, 22
DCU, 22

DCU_A, 22
DCU_B, 22
ENGINE_STATUS, 22
MPS_VALVE, 22
PD1, 23
PD2, 23
PD3, 23
POSTSHUTDOWN_OXIDIZERDUMP, 23
POSTSHUTDOWN_STANDBY, 22
POSTSHUTDOWN_-
TERMINATESEQUENCE, 23
PROM_STANDBY, 23
PV1, 23
PV10, 23
PV11, 23
PV12, 23
PV13, 23
PV17, 23
PV18, 23
PV19, 23
PV2, 23
PV20, 23
PV21, 23
PV22, 23
PV3, 23
PV4, 23
PV5, 23
PV6, 23
PV7, 23
PV8, 23
PV9, 23
SHUTDOWN_FAILSAFEPNEUMATIC, 22
SHUTDOWN_-
PROPELLANTVALVESCLOSED,
22
SHUTDOWN_-
THROTTLINGTOZEROTHRUST,
22
STARTMAINSTAGE_-
ELECTRICALLOCKUP, 22
STARTMAINSTAGE_FIXEDDENSITY, 22
STARTMAINSTAGE_-
HYDRAULICLOCKUP, 22
STARTMAINSTAGE_-
MAINSTAGEPHASENORMALCONTROL,
22
STARTMAINSTAGE_-
STARTPHASESTARTINITIATION,
22
STARTMAINSTAGE_-
STARTPHASETHRUSTBUILDUP,
22
STARTMAINSTAGE_THRUSTLIMITING,
22
STARTPREPARATION_ENGINEREADY,
22
STARTPREPARATION_-
PURGESEQUENCE1, 22
STARTPREPARATION_-
PURGESEQUENCE2, 22
STARTPREPARATION_-
PURGESEQUENCE3, 22
STARTPREPARATION_-
PURGESEQUENCE4, 22
MPS2FPS
Atlantis.h, 328
mps::_DATA_TABLE, 27
cmdPC, 28
posCCV, 28
posFPOV, 28
posMFV, 28
posMOV, 28
posOPOV, 28
powerlevel, 28
ShutdownEnable, 28
StartEnable, 28
STATUS, 28
timeCO, 28
timeESC, 28
timeTAG, 28
mps::ATVC, 83
~ATVC, 83
ATVC, 83
OnPostStep, 83
mps::BLOCK_II, 112
~BLOCK_II, 113
BLOCK_II, 113
dcCCV_CO, 114
dcCCV_ESC, 113
dcCCV_MS, 114
dcFPOV_CO, 114
dcFPOV_ESC, 113
dcFPOV_MS, 114
dcMFV_CO, 114
dcMFV_ESC, 113
dcMOV_CO, 113
dcMOV_ESC, 113
dcOPOV_CO, 114
dcOPOV_ESC, 113
dcOPOV_MS, 114
dcPC_CO, 113
dcPC_ESC, 113
dcPC_MS, 114
OnParseLine, 114
OnSaveState, 114
mps::EIU, 146
~EIU, 147
BUFFER_SSME_DATA_TABLE, 149

EIU, 147
engine, 149
ID, 149
OnPostStep, 148
RequestCMD, 148
RequestDATA, 148
mps::MPS, 208
~MPS, 209
MPS, 209
MPSValveClose, 209
MPSValveOpen, 209
MPSValveStatus, 209
OnParseLine, 210
OnPostStep, 210
OnSaveState, 210
posPD1, 212
posPD2, 212
posPD3, 212
ptrPV1, 212
ptrPV10, 212
ptrPV11, 212
ptrPV12, 212
ptrPV13, 212
ptrPV17, 212
ptrPV18, 212
ptrPV19, 212
ptrPV2, 212
ptrPV20, 212
ptrPV21, 212
ptrPV22, 212
ptrPV3, 212
ptrPV4, 212
ptrPV5, 212
ptrPV6, 212
ptrPV7, 212
ptrPV8, 212
ptrPV9, 212
mps::SSME, 275
~SSME, 277
activeDCU, 283
cmdChannelReset, 281
cmdCheckoutStandby, 281
cmdControllerReset, 281
cmdExitPROM, 281
cmdFRT1, 281
cmdOxidizerDump, 281
cmdPowerOff, 281
cmdPowerOn, 281
cmdPurgeSequence1, 280
cmdPurgeSequence2, 281
cmdPurgeSequence3, 281
cmdPurgeSequence4, 281
cmdShutdown, 281
cmdShutdownEnable, 281
cmdStart, 281
cmdStartEnable, 281
cmdTerminateSequence, 281
cmdThrottle, 281
COtimecoef, 283
dataDataTable, 281
dcCCV_CO, 279
dcCCV_ESC, 279
dcCCV_MS, 280
dcFPOV_CO, 279
dcFPOV_ESC, 279
dcFPOV_MS, 280
dcMFV_CO, 279
dcMFV_ESC, 279
dcMOV_CO, 279
dcMOV_ESC, 278
dcOPOV_CO, 279
dcOPOV_ESC, 279
dcOPOV_MS, 280
dcPC_CO, 279
dcPC_ESC, 278
dcPC_MS, 279
dir, 283
FPL, 283
FPL_THRUST, 283
FPOV_pos_CO, 283
GetHandle, 282
ID, 283
Ignition, 277
ISP0, 283
ISP1, 283
MPL, 283
OnParseLine, 283
OnPostStep, 282
OnSaveState, 282
OPOV_pos_CO, 283
PCfromOSFStoSTS, 278
PCfromPCTtoPSI, 278
PCfromSTStoOSFS, 278
phET, 283
pos, 283
PSN4time, 283
Realize, 281
SetCOTime, 278
Shutdown, 277
SSME, 277
SSME_DATA_TABLE, 283
Throttling, 278
thSSME, 283
MPS_ENGINE_POWER
PanelR2, 260
MPS_He_Reg_Press
Atlantis, 66
CRT, 130

MPS_He_Tank_Press
Atlantis, 66
CRT, 130
mps_helium_tank
Atlantis, 66
MPS_Manifold_Press
Atlantis, 66
CRT, 130
MPS_STATE
PanelR2, 260
MPS_VALVE
mps, 22
MPSValveClose
mps::MPS, 209
MPSValveOpen
mps::MPS, 209
MPSValveStatus
mps::MPS, 209
mrate
BasicValve, 101
MRL
Atlantis, 66
MRL_AftMicroswitches
Atlantis, 66
MRL_FwdMicroswitches
Atlantis, 66
MRL_MidMicroswitches
Atlantis, 66
MsgProc
CRT, 124
msh_hires_bkup
AtlantisConfig.cpp, 338
msh_lores_bkup
AtlantisConfig.cpp, 338
MshEnableHires
AtlantisConfig, 75
MshHiresEnabled
AtlantisConfig, 75
mshMLP
MLP, 207
MTU_ACCU_INDEX
dps, 14
MTU_EVTTMR_INDEX
dps, 14
mu
Atlantis, 66
MultiplyByMatrix
Atlantis, 55
myGroup
vc::MDU, 198

N2_MOLARMASS
OMSPod.h, 464
N2_PIPE_AREA
OMSPod.h, 464
N2_TANK_VOLUME
OMSPod.h, 464
N2Accu
OMSPod, 218
N2CActuator1
OMSPod, 218
N2CActuator2
OMSPod, 218
N2CheckValve
OMSPod, 218
N2CValve1
OMSPod, 218
N2CValve2
OMSPod, 218
N2PressIsolVlv
OMSPod, 218
N2PurgeIn
OMSPod, 218
N2RegPress
OMSPod, 218
N2RegulState
OMSPod, 218
N2ReliefValve
OMSPod, 218
N2Tank
OMSPod, 218
N2ValvesIn
OMSPod, 218
N2Vertex1
OMSPod, 218
n_aoa1
Atlantis.cpp, 310
n_mach
Atlantis.cpp, 310
n_trimExt
Atlantis.cpp, 310
Name
AtlantisConfig, 74
name
vc::BasicPanel, 96
Navigate
Atlantis, 55
NewEntry
KeyboardInput, 179
newmfd
Atlantis, 66
NGRP
meshres.h, 416
meshres_FSS_RSS.h, 394
NGRP_KU
meshres_ku.h, 417
NGRP_RMS
meshres_RMS.h, 424

NGRP_VC
meshres_vc.h, 442
NGRPODSVC
meshres_odsvc.h, 422
nhCameraLabel
Atlantis, 66
nhLastReceived
CommModeHandler, 119
nhLastSend
CommModeHandler, 119
nhOptionPos
CommModeHandler, 119
NMAT
meshres.h, 416
meshres_FSS_RSS.h, 394
NMAT_KU
meshres_ku.h, 417
NMAT_RMS
meshres_RMS.h, 424
NMAT_VC
meshres_vc.h, 442
NMATODSVC
meshres_odsvc.h, 422
NO_SLAG_TIME
Atlantis.h, 328
Normalize
Atlantis, 55
NormZ
Atlantis, 55
NosePitchThrusters
Atlantis.h, 328
NoseRollThrusters
Atlantis.h, 328
NoseRotThrusters
Atlantis.h, 328
NoseThrusters
Atlantis, 66
NoseYawThrusters
Atlantis.h, 328
NTEX
meshres.h, 416
meshres_FSS_RSS.h, 394
NTEX_KU
meshres_ku.h, 417
NTEX_RMS
meshres_RMS.h, 424
NTEX_VC
meshres_vc.h, 442
NTEXODSVC
meshres_odsvc.h, 422
NullRates
Atlantis, 66
NullStartAngle
Atlantis, 55
NUM_GPC_VARIABLES
GPCVariables.h, 370
NUM_O3_ANIMATIONS
PanelO3.h, 484
NUM_O3_SWITCHES
PanelO3.h, 484

OAA_OPERATING_SPEED
SSU_FSS_RSS.h, 397
OAA_proc
SSU_FSS_RSS, 285
OAA_status
SSU_FSS_RSS, 285
object_anim
MechActuator, 201
obss_attach
Atlantis, 60
ODS
eva_docking::ODS, 214
ODS.h
ODS_MESHNAME, 387
ODS_MESHNAME
ODS.h, 387
ODS_POS
Atlantis.h, 328
ODSPANEL_MESHNAME
Atlantis.h, 328
OFS_LAUNCH_LEFTSRB
Atlantis.h, 328
OFS_LAUNCH_ORBITER
Atlantis.h, 328
OFS_LAUNCH_RIGHTSRB
Atlantis.h, 328
OFS_LAUNCH_TANK
Atlantis.h, 328
OFS_MMU
Atlantis.h, 328
ofs_sts_sat
Atlantis, 60
OFS_WITHTANK_ORBITER
Atlantis.h, 328
OFS_WITHTANK_TANK
Atlantis.h, 328
OFS_ZERO
Atlantis.h, 328
old_switch_state
PanelA4, 224
PanelC2, 236
PanelO3, 254
oldwheelnumber
PanelC2, 236
OM
Atlantis, 66
omega

Atlantis, 66
OMS
Atlantis, 66
OMS_KIT
Atlantis.h, 323
OMS_LEFT
Atlantis.h, 323
OMS_RIGHT
Atlantis.h, 323
OMS_Assist
Atlantis, 66
OMS_Eng
PanelC3, 239
oms_helium_tank
Atlantis, 66
OMS_HELIUM_TANK_MASS
Atlantis.h, 328
OMS_REF
Atlantis.h, 323
OMSGimbal
Atlantis, 66
CRT, 127
OMSMPS
CRT, 126
OMSPod, 215
CalculateN2System, 218
CommandN2PressIsolValve, 218
director, 218
fN2PressIsolVlv, 218
fN2TankMass, 218
GetN2TankPressure, 218
N2Accu, 218
N2CActuator1, 218
N2CActuator2, 218
N2CheckValve, 218
N2CValve1, 218
N2CValve2, 218
N2PressIsolVlv, 218
N2PurgeIn, 218
N2RegPress, 218
N2RegulState, 218
N2ReliefValve, 218
N2Tank, 218
N2ValvesIn, 218
N2Vertex1, 218
OMSPod, 218
PreStep, 218
Propagate, 218
SetN2TankFillState, 218
SetOMSThrusterPos, 218
OMSPod.h
GAS_CONSTANT, 464
L_ISOLVLV, 464
N2_MOLARMASS, 464
N2_PIPE_AREA, 464
N2_TANK_VOLUME, 464
OMSSubsystem, 220
~OMSSubsystem, 221
leftPod, 221
OMSSubsystem, 221
ParseLine, 221
PreStep, 221
Propagate, 221
rightPod, 221
SaveState, 221
SetClassCaps, 221
OnDumpToLog
AtlantisSubsystem, 81
eva_docking::BasicExternalAirlock, 88
ISubsystem, 176
vc::BasicVCComponent, 107
OnFail
AtlantisSubsystem, 82
ISubsystem, 176
vc::BasicVCComponent, 108
OnLooseFocus
CommModeHandler, 118
OnMouseEvent
vc::BasicSwitch, 98
vc::BasicVCComponent, 105
vc::MDU, 195
OnNewMessage
dps::BIU, 111
OnParseLine
AtlantisSubsystem, 81
dps::MasterTimingUnit, 188
dps::RAM16, 270
eva_docking::BasicExternalAirlock, 87
ISubsystem, 176
mps::BLOCK_II, 114
mps::MPS, 210
mps::SSME, 283
vc::BasicPanel, 96
vc::BasicVCComponent, 107
vc::PanelGroup, 251
OnPlaybackEvent
AtlantisSubsystem, 81
eva_docking::BasicExternalAirlock, 87
ISubsystem, 175
vc::BasicVCComponent, 107
OnPositionChange
vc::BasicSwitch, 99
vc::StandardSwitch, 288
OnPositionDown
vc::BasicSwitch, 99
OnPositionUp
vc::BasicSwitch, 98
OnPostStep

AirDataProbeSystem, 33
AtlantisSubsystem, 80
eva_docking::BasicExternalAirlock, 87
ISubsystem, 175
MechActuator, 200
mps::ATVC, 83
mps::EIU, 148
mps::MPS, 210
mps::SSME, 282
vc::BasicPanel, 96
vc::BasicVCComponent, 107
vc::PanelGroup, 251
OnPreStep
AirDataProbeSystem, 33
AtlantisSubsystem, 80
dps::MasterTimingUnit, 187
eva_docking::BasicExternalAirlock, 86
ISubsystem, 175
MechActuator, 200
vc::BasicPanel, 95
vc::BasicVCComponent, 107
vc::PanelGroup, 251
OnPropagate
AirDataProbeSystem, 32
AtlantisSubsystem, 80
dps::MasterTimingUnit, 188
eva_docking::BasicExternalAirlock, 87
ISubsystem, 175
MechActuator, 200
vc::BasicPanel, 96
vc::BasicVCComponent, 107
vc::PanelGroup, 251
OnReceiveFocus
CommModeHandler, 117
OnSaveState
AtlantisSubsystem, 81
dps::MasterTimingUnit, 188
dps::RAM16, 270
eva_docking::BasicExternalAirlock, 87
ISubsystem, 176
mps::BLOCK_II, 114
mps::MPS, 210
mps::SSME, 282
vc::BasicPanel, 96
vc::BasicSwitch, 99
vc::BasicVCComponent, 107
vc::PanelGroup, 251
OnVCMouseEvent
vc::BasicPanel, 94
vc::PanelGroup, 251
OnVCRedrawEvent
vc::BasicPanel, 95
vc::BasicVCComponent, 107
vc::PanelGroup, 251
oparam
Atlantis, 66
Open
BasicValve, 101
ValveTypeBool, 297
OpenDialog
GearOp, 158
PayloadBayOp, 265
OperateLandingGear
GearOp, 159
OperateSpeedbrake
Atlantis, 53
OPOV_pos_CO
mps::SSME, 283
OPS
KeyboardInput, 179
ops
Atlantis, 66
CRT, 130
ORBITER_CS
Atlantis.h, 328
ORBITER_CS_GEAR
Atlantis.h, 328
ORBITER_CW
Atlantis.h, 328
ORBITER_CW_GEAR
Atlantis.h, 328
ORBITER_DOCKPOS
Atlantis.h, 328
ORBITER_EMPTY_MASS
Atlantis.h, 328
ORBITER_FRCS_PROPELLANT_MASS
Atlantis.h, 328
ORBITER_MAIN_ISP0
Atlantis.h, 328
ORBITER_MAIN_ISP1
Atlantis.h, 328
ORBITER_MAIN_THRUST
Atlantis.h, 328
ORBITER_MAX_PROPELLANT_MASS
Atlantis.h, 328
ORBITER_MAX_THRUST
Atlantis.h, 328
ORBITER_MODULE
Atlantis.cpp, 306
Atlantis_SRB.cpp, 331
Atlantis_Tank.cpp, 332
AtlantisConfig.cpp, 338
CRT.cpp, 350
orbiter_ofs
Atlantis, 60
ORBITER_OMS_ISP0
Atlantis.h, 328
ORBITER_OMS_ISP1

Atlantis.h, 328
ORBITER_OMS_PITCH
Atlantis.h, 328
ORBITER_OMS_THRUST
Atlantis.h, 328
ORBITER_OMS_YAW
Atlantis.h, 328
ORBITER_PITCH_TORQUE
Atlantis.h, 328
ORBITER_RCS_ISP0
Atlantis.h, 328
ORBITER_RCS_ISP1
Atlantis.h, 328
ORBITER_RCS_THRUST
Atlantis.h, 328
ORBITER_ROLL_TORQUE
Atlantis.h, 328
ORBITER_YAW_TORQUE
Atlantis.h, 328
OrbiterA
Atlantis.cpp, 310
Orbiterb
Atlantis.cpp, 310
Orbiterc
Atlantis.cpp, 310
OrbiterS
Atlantis.cpp, 310
OutA
hydraulics::HydActuator, 162
OutB
hydraulics::HydActuator, 162
OutC
hydraulics::HydActuator, 162
OutD
hydraulics::HydActuator, 162
outPort
vc::StdSwitch2, 291
outPortA
vc::StdSwitch3, 293
outPortB
vc::StdSwitch3, 293
outY
gnc::AccAssembly, 30
outZ
gnc::AccAssembly, 30
ovcConsumeBufferedKey
SSU_FSS_RSS.cpp, 396
ovcExit
Atlantis.cpp, 307
Atlantis_SRB.cpp, 331
Atlantis_Tank.cpp, 332
MLP.cpp, 448
SSU_FSS_RSS.cpp, 396
ovcInit
Atlantis.cpp, 307
Atlantis_SRB.cpp, 331
Atlantis_Tank.cpp, 332
MLP.cpp, 448
SSU_FSS_RSS.cpp, 396
ovcLoadStateEx
SSU_FSS_RSS.cpp, 396
ovcSaveState
SSU_FSS_RSS.cpp, 396
ovcSetClassCaps
SSU_FSS_RSS.cpp, 396
ovcSetState
Atlantis_Tank.cpp, 332
ovcTimestep
SSU_FSS_RSS.cpp, 396
own_aid
vc::BasicPanel, 96

P
Atlantis, 66
FLOWSTATE, 154
p
Atlantis, 66
p_a
vc::BasicVCComponent, 109
p_b
vc::BasicVCComponent, 109
p_c
vc::BasicVCComponent, 109
p_d
vc::BasicVCComponent, 109
pA7A8Panel
Atlantis, 58
pADIAttitude
vc::PanelF6, 242
vc::PanelF8, 249
pADIError
vc::PanelF6, 242
vc::PanelF8, 249
pADIRate
vc::PanelF6, 242
vc::PanelF8, 249
pADPS
Atlantis, 60
Paint
vc::MDU, 195
PaintEdgeMenu
vc::MDU, 195
PaintMarkings
Atlantis, 53
pAirData
vc::PanelF6, 242
vc::PanelF8, 249
PanelA4, 222

anim_VC_A4, 224
Atlantis, 57, 224
DefineAnimations, 223
DefineVCAnimations, 223
old_switch_state, 224
PanelA4, 223
ParseScenarioLine, 224
RegisterVC, 223
SaveState, 224
sMETDays, 224
sMETHours, 224
sMETMinutes, 224
sMETSeconds, 224
Step, 224
sTimerMinutes, 224
sTimerSeconds, 224
sts, 224
switch_state, 224
switch_timer, 224
UpdateMesh, 223
UpdateVC, 223
VCMouseEvent, 223
VCRedrawEvent, 223
panela4
Atlantis, 60
PanelA4.cpp
ActionString, 467
g_hc, 467
g_Param, 467
PanelA7A8ODS
vc::PanelA7A8ODS, 227
PanelA8, 231
anim_VC_A8, 233
Atlantis, 57, 233
CRT, 233
DefineVCAnimations, 232
PanelA8, 232
ParseScenarioLine, 232
PayloadBayOp, 269
RegisterVC, 232
SaveState, 233
Step, 232
sts, 233
switch_state, 233
tkbk_state, 233
UpdateMesh, 232
UpdateVC, 232
VCDrawTalkback, 233
VCMouseEvent, 232
VCRedrawEvent, 232
panela8
Atlantis, 60
PanelA8.cpp
ActionString, 469
g_hc, 469
g_Param, 469
PanelC2, 234
~PanelC2, 235
anim_VC_C2, 236
anim_VC_C2Wheel, 236
Atlantis, 57, 236
CreateVisual, 235
CRT, 236
DefineAnimations, 235
DefineVCAnimations, 235
digit_texture, 236
EventTimerControl, 236
EventTimerMode, 236
EventTimerWheelRevs, 236
old_switch_state, 236
oldwheelnumber, 236
PanelC2, 235
ParseScenarioLine, 235
RegisterVC, 235
SaveState, 236
Step, 235
sts, 236
switch_state, 236
switch_timer, 236
tgtwheel_state, 236
UpdateMesh, 235
UpdateVC, 235
VCMouseEvent, 235
VCRedrawEvent, 235
wheel_tex, 236
wheelnumber, 236
wheelState, 236
panelc2
Atlantis, 60
PanelC2.cpp
ActionString, 472
EVTTMR_WHEELMAX, 472
EVTTMR_WINDOW_GROUPS, 472
g_hc, 472
g_Param, 472
PanelC2.h
REF_C2S1, 474
REF_C2S10, 474
REF_C2S11, 474
REF_C2S12, 474
REF_C2S2, 474
REF_C2S3, 474
REF_C2S4, 474
REF_C2S5, 474
REF_C2S6, 474
REF_C2S7, 474
REF_C2S8, 474
REF_C2S9, 474

PanelC3, 237
Air_Data_Probe, 239
Air_Data_Stow, 239
AirDataProbe, 239
AirDataProbeControl, 239
anim_VC_C3, 239
Atlantis, 57, 239
CheckOMSArm, 238
CheckProbesDeployed, 238
DefineAnimations, 238
DefineVCAnimations, 238
EngControl, 239
OMS_Eng, 239
PanelC3, 238
ParseScenarioLine, 238
RegisterVC, 238
SaveState, 238
Step, 238
sts, 239
UpdateMesh, 238
UpdateVC, 238
VCMouseEvent, 238
VCRedrawEvent, 238
PanelC3.cpp
ActionString, 475
g_hc, 475
g_Param, 475
PanelC3.h
REF_C3S1, 477
REF_C3S19, 477
REF_C3S2, 477
REF_C3S20, 477
REF_C3S8, 477
REF_C3S9, 477
PanelF6
vc::PanelF6, 241
PanelF7, 245
Atlantis, 246
DefineAnimations, 246
DefineVCAnimations, 246
PanelF7, 246
ParseScenarioLine, 246
RegisterVC, 246
SaveState, 246
Step, 246
sTimerMinutes, 246
sTimerSeconds, 246
sts, 246
UpdateMesh, 246
UpdateVC, 246
vc::PanelF7, 244
VCMouseEvent, 246
VCRedrawEvent, 246
PanelF7.cpp
ActionString, 478
g_hc, 478
g_Param, 478
PanelF8
vc::PanelF8, 248
PanelGroup
vc::PanelGroup, 251
PanelO3, 252
anim_VC_O3, 254
Atlantis, 57, 254
DefineAnimations, 253
DefineVCAnimations, 253
old_switch_state, 254
PanelO3, 253
ParseScenarioLine, 253
RegisterVC, 253
SaveState, 254
sMETDays, 254
sMETHours, 254
sMETMinutes, 254
sMETSeconds, 254
sOMSFuel, 254
Step, 253
sTimerMinutes, 254
sTimerSeconds, 254
sts, 254
switch_state, 254
switch_timer, 254
UpdateMesh, 253
UpdateVC, 253
VCMouseEvent, 253
VCRedrawEvent, 253
panelo3
Atlantis, 60
PanelO3.cpp
ActionString, 482
g_hc, 482
g_Param, 482
PanelO3.h
NUM_O3_ANIMATIONS, 484
NUM_O3_SWITCHES, 484
REF_O3S12, 484
SWITCH_O3S12, 484
PanelR2, 255
anim_VC_R2, 260
APU_AUTO_SHUT_DOWN, 260
APU_CNTLR_PWR, 260
APU_CONTROL, 260
APU_FUEL, 260
APU_FUEL_TK_VLV, 260
APU_READY, 260
APU_Speed, 260
APU_SPEED_SELECT, 260
APU_STATE, 260

Atlantis, 57, 260
bHydraulicPressure, 260
BOILER_CNTLR, 260
BOILER_CNTLRHTR, 260
BOILER_N2_SUPPLY, 260
CENTERLINE_LATCH, 260
CenterlineLatches, 260
CheckAPUReadytoStart, 258
CheckAPUShutdown, 258
CheckMPSArmed, 258
CRT, 260
DefineAnimations, 257
DefineVCAnimations, 257
Fuel_Press, 260
HE_ISOLATION_A, 260
HE_ISOLATION_B, 260
HYD_CIRC_PUMP, 260
HYD_MAIN_PUMP_PRESS, 260
Hydraulic_Press, 260
LDoorLatches, 260
LEFT_DOOR, 260
LEFT_LATCH, 260
LETUmbDoorStatus, 260
MPS_ENGINE_POWER, 260
MPS_STATE, 260
PanelR2, 257
ParseScenarioLine, 258
RDoorLatches, 260
RegisterVC, 257
RETUmbDoorStatus, 260
RIGHT_DOOR, 260
RIGHT_LATCH, 260
SaveState, 258
SetETUmbDoorAction, 258
Step, 257
sts, 260
tkbk_state, 260
UpdateMesh, 257
UpdateVC, 257
VCDrawTalkback, 258
VCMouseEvent, 257
VCRedrawEvent, 257
PanelR2.cpp
ActionString, 485
g_hc, 485
g_Param, 485
panels
vc::PanelGroup, 251
pARLKTNLFanA
vc::PanelA7A8ODS, 229
pARLKTNLFanB
vc::PanelA7A8ODS, 229
ParseLine
KuBandSystem, 181
OMSSubsystem, 221
ParseScenarioLine
GearOp, 159
PanelA4, 224
PanelA8, 232
PanelC2, 235
PanelC3, 238
PanelF7, 246
PanelO3, 253
PanelR2, 258
PayloadBayOp, 267
SubsystemDirector, 296
PASSTRAJ
CRT, 128
PAYCAM_DlgProc
Atlantis, 57
Atlantis.cpp, 307
PayloadBayOp, 262
anim_VC_R13L, 269
Atlantis, 57, 269
AutoDoorSequenceControl, 266
BayDoor, 269
BayDoorOp, 269
BayDoorStatus, 269
BD_DISABLE, 263
BD_ENABLE, 263
BDO_CLOSE, 263
BDO_OPEN, 263
BDO_STOP, 263
CLBayDoorLatch, 269
DefineAnimations, 265
DlgProc, 268
GetDoorAction, 266
GetRadiatorAction, 266
hDlg, 269
KU_DEPLOY, 264
KU_DIRECT_OFF, 264
KU_DIRECT_ON, 264
KU_GND, 264
KU_STOW, 264
KuAntennaStatus, 269
KuCtrl, 269
KuDirectCtrl, 269
LC_LATCH, 264
LC_OFF, 264
LC_RELEASE, 264
MechPwr, 269
MP_OFF, 263
MP_ON, 263
OpenDialog, 265
PanelA8, 269
ParseScenarioLine, 267
PayloadBayOp, 265
PlOp_DlgProc, 269

RadiatorCtrl, 269
RadiatorStatus, 269
RadLatchCtrl, 269
RadLatchStatus, 269
RC_DEPLOY, 264
RC_OFF, 264
RC_STOW, 264
RegisterVC, 265
RevertKuAntennaAction, 267
SaveState, 268
SetDoorAction, 266
SetKuAntennaAction, 267
SetRadiatorAction, 267
SetRadLatchAction, 267
Step, 265
sts, 269
tkbk_state, 269
UpdateDialog, 268
UpdateVC, 265
VCDrawTalkback, 269
VCMouseEvent, 265
VCRedrawEvent, 265
pBundle
discsignals::DiscretePort, 145
pBus
dps::BIU, 111
pCDR1
vc::PanelF6, 242
pCDR2
vc::PanelF6, 242
PCfromOSFStoSTS
mps::SSME, 278
PCfromPCTtoPSI
mps::SSME, 278
PCfromSTStoOSFS
mps::SSME, 278
pCommModeHandler
Atlantis, 60
pControlPanelPowerA
vc::PanelA7A8ODS, 229
pControlPanelPowerB
vc::PanelA7A8ODS, 229
pControlPanelPowerC
vc::PanelA7A8ODS, 229
pCRT1
vc::PanelF7, 244
pCRT2
vc::PanelF7, 244
pCRT3
vc::PanelF7, 244
PD1
mps, 23
PD2
mps, 23
PD3
mps, 23
PDRS
CRT, 127
PEG7
Atlantis, 66
pExtAirlock
Atlantis, 60
pFltCntlrPower
vc::PanelF6, 242
vc::PanelF8, 249
pgAftMSS
Atlantis, 60
pgAftOOS
Atlantis, 60
pgAftPSS
Atlantis, 60
pgCenter
Atlantis, 60
pgForward
Atlantis, 60
pgLeft
Atlantis, 60
pgOverhead
Atlantis, 60
pgRight
Atlantis, 60
ph_frcs
Atlantis, 62
ph_lrcs
Atlantis, 62
ph_main
Atlantis_SRB, 69
ph_oms
Atlantis, 62
ph_rrcs
Atlantis, 62
ph_srb
Atlantis, 62
ph_tank
Atlantis, 62
PHASE1
eps, 16
PHASE2
eps, 16
PHASE3
eps, 16
phBSM
Atlantis_SRB, 69
phET
mps::SSME, 283
phi
Atlantis, 66
pHSISelectChannel

vc::PanelF6, 242
vc::PanelF8, 249
pHSISelectMode
vc::PanelF6, 242
vc::PanelF8, 249
pHSISelectSource
vc::PanelF6, 242
vc::PanelF8, 249
pIDP
Atlantis, 60
dps::IDPSoftware, 171
PITCH
Atlantis.h, 322
Pitch
Atlantis, 66
RefPoints, 271
pitch
Atlantis, 66
PitchActive
Atlantis, 60
PitchOff
Atlantis.cpp, 310
PitchYawRoll
Atlantis, 66
pKeyboardA
dps::IDP, 168
pKeyboardB
dps::IDP, 168
PL
dps, 15
pl_mass
Atlantis, 66
pl_ops
dps::IDP, 168
pl_specdisp
dps::IDP, 168
PlaybackEvent
SubsystemDirector, 296
PlBayOp.cpp
ActionString, 488
g_hc, 488
g_Param, 488
PlOp_DlgProc, 488
sts_dlg, 488
pLightsAirlock14
vc::PanelA7A8ODS, 229
pLightsAirlock23
vc::PanelA7A8ODS, 229
pLightsCLVestPORT
vc::PanelA7A8ODS, 229
pLightsCLVestSTBD
vc::PanelA7A8ODS, 229
pLightsDockingAFT
vc::PanelA7A8ODS, 229
pLightsDockingFWD
vc::PanelA7A8ODS, 229
plop
Atlantis, 60
PlOp_DlgProc
PayloadBayOp, 269
PlBayOp.cpp, 488
PLTKeyboard
Atlantis, 60
PMDU
vc, 25
pMemory
dps::RAM16, 270
pMFD1
vc::PanelF7, 244
pMFD2
vc::PanelF7, 244
PMI
Atlantis, 66
CRT, 130
pmodules
dps::MDM, 191
pMTU
Atlantis, 60
pOMS
Atlantis, 60
pos
BasicValve, 101
mps::SSME, 283
ValveTypeBool, 298
POS_HDP
Atlantis.h, 328
POS_MPS_SMOKE
MLP.h, 449
POS_SRB_SMOKE
MLP.h, 449
posCCV
mps::_DATA_TABLE, 28
posFPOV
mps::_DATA_TABLE, 28
PosLimit0
MechActuator, 201
PosLimit1
MechActuator, 201
posMFV
mps::_DATA_TABLE, 28
posMoon
Atlantis, 66
posMOV
mps::_DATA_TABLE, 28
posOPOV
mps::_DATA_TABLE, 28
posPD1
mps::MPS, 212

posPD2
mps::MPS, 212
posPD3
mps::MPS, 212
POSTSHUTDOWN_OXIDIZERDUMP
mps, 23
POSTSHUTDOWN_STANDBY
mps, 22
POSTSHUTDOWN_TERMINATESEQUENCE
mps, 23
PostStep
CommModeHandler, 117
KuBandSystem, 181
SubsystemDirector, 296
powerlevel
mps::_DATA_TABLE, 28
pParent
vc::BasicVCComponent, 109
pPLT1
vc::PanelF8, 249
pPLT2
vc::PanelF8, 249
pPSUPowerMNA
vc::PanelA7A8ODS, 229
pPSUPowerMNB
vc::PanelA7A8ODS, 229
pPyroPowerMNA
vc::PanelA7A8ODS, 229
pPyroPowerMNC
vc::PanelA7A8ODS, 229
pPyrosAp
vc::PanelA7A8ODS, 229
pPyrosBp
vc::PanelA7A8ODS, 229
pPyrosCp
vc::PanelA7A8ODS, 229
pRAMBuffer
dps::BIU, 111
pRdrAltm
vc::PanelF6, 242
vc::PanelF8, 249
PreStep
KuBandSystem, 181
OMSPod, 218
OMSSubsystem, 221
SubsystemDirector, 296
PRI_ATT_DB
DAPConfig, 134
PRI_COMP
DAPConfig, 134
PRI_P_OPTION
DAPConfig, 134
PRI_RATE_DB
DAPConfig, 134
PRI_ROT_PLS
DAPConfig, 134
PRI_ROT_RATE
DAPConfig, 134
PRI_TRAN_PLS
DAPConfig, 134
PRI_Y_OPTION
DAPConfig, 134
prim_idp
vc::MDU, 198
PrintToBuffer
vc::MDU, 197
PRO
KeyboardInput, 179
PROBE_AXIS
Atlantis.h, 328
PROBEL_REF
Atlantis.h, 328
PROBER_REF
Atlantis.h, 328
Projection
Atlantis, 55
ProjectMouse
vc::BasicVCComponent, 106
PROM_STANDBY
mps, 23
Propagate
KuBandSystem, 181
OMSPod, 218
OMSSubsystem, 221
pshSlag1
Atlantis, 60
pshSlag2
Atlantis, 60
pshSlag3
Atlantis, 60
PSN4time
mps::SSME, 283
pSSME
Atlantis, 60
pSTS
CommModeHandler, 119
psts
SubsystemDirector, 296
vc::BasicPanel, 96
psubsystems
Atlantis, 60
pswitchpull
vc::LockableLever, 183
pswitchrot
vc::LockableLever, 183
vc::StandardSwitch, 289
pSys1Vent
vc::PanelA7A8ODS, 229

pSys1VentIsol
vc::PanelA7A8ODS, 229
pSys2Vent
vc::PanelA7A8ODS, 229
pSys2VentIsol
vc::PanelA7A8ODS, 229
pSystemPowerMNA
vc::PanelA7A8ODS, 229
pSystemPowerMNB
vc::PanelA7A8ODS, 229
pszCameraLabelBuffer
Atlantis, 66
ptrPV1
mps::MPS, 212
ptrPV10
mps::MPS, 212
ptrPV11
mps::MPS, 212
ptrPV12
mps::MPS, 212
ptrPV13
mps::MPS, 212
ptrPV17
mps::MPS, 212
ptrPV18
mps::MPS, 212
ptrPV19
mps::MPS, 212
ptrPV2
mps::MPS, 212
ptrPV20
mps::MPS, 212
ptrPV21
mps::MPS, 212
ptrPV22
mps::MPS, 212
ptrPV3
mps::MPS, 212
ptrPV4
mps::MPS, 212
ptrPV5
mps::MPS, 212
ptrPV6
mps::MPS, 212
ptrPV7
mps::MPS, 212
ptrPV8
mps::MPS, 212
ptrPV9
mps::MPS, 212
PV1
mps, 23
PV10
mps, 23
PV11
mps, 23
PV12
mps, 23
PV13
mps, 23
PV17
mps, 23
PV18
mps, 23
PV19
mps, 23
PV2
mps, 23
PV20
mps, 23
PV21
mps, 23
PV22
mps, 23
PV3
mps, 23
PV4
mps, 23
PV5
mps, 23
PV6
mps, 23
PV7
mps, 23
PV8
mps, 23
PV9
mps, 23

r
Atlantis, 66
r2d2
Atlantis, 60
RAD_OPERATING_SPEED
Atlantis.h, 328
RadiatorCtrl
PayloadBayOp, 269
RadiatorStatus
PayloadBayOp, 269
Radius
Atlantis, 66
RADLATCH_OPERATING_SPEED
Atlantis.h, 328
RadLatchCtrl
PayloadBayOp, 269
RadLatchStatus
PayloadBayOp, 269
RAM16

dps::RAM16, 270
range
Atlantis, 55
RateCommand
Atlantis, 55
RateDeadband
Atlantis, 66
rbar
Atlantis, 66
RC_DEPLOY
PayloadBayOp, 264
RC_OFF
PayloadBayOp, 264
RC_STOW
PayloadBayOp, 264
RCS_F1D
Atlantis.h, 328
RCS_F1D_DIR
Atlantis.h, 328
RCS_F1D_OFS
Atlantis.h, 328
RCS_F1F
Atlantis.h, 328
RCS_F1F_DIR
Atlantis.h, 328
RCS_F1F_OFS
Atlantis.h, 328
RCS_F1L
Atlantis.h, 328
RCS_F1L_DIR
Atlantis.h, 328
RCS_F1L_OFS
Atlantis.h, 328
RCS_F1U
Atlantis.h, 328
RCS_F1U_DIR
Atlantis.h, 328
RCS_F1U_OFS
Atlantis.h, 328
RCS_F2D
Atlantis.h, 328
RCS_F2D_DIR
Atlantis.h, 328
RCS_F2D_OFS
Atlantis.h, 328
RCS_F2F
Atlantis.h, 328
RCS_F2F_DIR
Atlantis.h, 328
RCS_F2F_OFS
Atlantis.h, 328
RCS_F2R
Atlantis.h, 328
RCS_F2R_DIR
Atlantis.h, 328
RCS_F2R_OFS
Atlantis.h, 328
RCS_F2U
Atlantis.h, 328
RCS_F2U_DIR
Atlantis.h, 328
RCS_F2U_OFS
Atlantis.h, 328
RCS_F3D
Atlantis.h, 328
RCS_F3D_DIR
Atlantis.h, 328
RCS_F3D_OFS
Atlantis.h, 328
RCS_F3F
Atlantis.h, 328
RCS_F3F_DIR
Atlantis.h, 328
RCS_F3F_OFS
Atlantis.h, 328
RCS_F3L
Atlantis.h, 328
RCS_F3L_DIR
Atlantis.h, 328
RCS_F3L_OFS
Atlantis.h, 328
RCS_F3U
Atlantis.h, 328
RCS_F3U_DIR
Atlantis.h, 328
RCS_F3U_OFS
Atlantis.h, 328
RCS_F4D
Atlantis.h, 328
RCS_F4D_DIR
Atlantis.h, 328
RCS_F4D_OFS
Atlantis.h, 328
RCS_F4R
Atlantis.h, 328
RCS_F4R_DIR
Atlantis.h, 328
RCS_F4R_OFS
Atlantis.h, 328
RCS_F5L
Atlantis.h, 328
RCS_F5R
Atlantis.h, 328
RCS_L1A
Atlantis.h, 328
RCS_L1L
Atlantis.h, 328
RCS_L1U

Atlantis.h, 328
RCS_L2D
Atlantis.h, 328
RCS_L2L
Atlantis.h, 328
RCS_L2U
Atlantis.h, 328
RCS_L3A
Atlantis.h, 328
RCS_L3D
Atlantis.h, 328
RCS_L3L
Atlantis.h, 328
RCS_L4D
Atlantis.h, 328
RCS_L4L
Atlantis.h, 328
RCS_L4U
Atlantis.h, 328
RCS_L5D
Atlantis.h, 328
RCS_L5R
Atlantis.h, 328
RCS_R1A
Atlantis.h, 328
RCS_R1R
Atlantis.h, 328
RCS_R1U
Atlantis.h, 328
RCS_R2D
Atlantis.h, 328
RCS_R2R
Atlantis.h, 328
RCS_R2U
Atlantis.h, 328
RCS_R3A
Atlantis.h, 328
RCS_R3D
Atlantis.h, 328
RCS_R3R
Atlantis.h, 328
RCS_R4D
Atlantis.h, 328
RCS_R4R
Atlantis.h, 328
RCS_R4U
Atlantis.h, 328
RCS_R5D
Atlantis.h, 328
RCS_R5R
Atlantis.h, 328
RCSEnabled
Atlantis, 66
rdoor_drag
Atlantis, 60
RDoorLatches
PanelR2, 260
Re
Atlantis, 66
ReadImage
dps::RAM16, 270
ReadStatus
CRT, 126
Realize
AtlantisSubsystem, 79
eva_docking::BasicExternalAirlock, 86
ISubsystem, 175
mps::SSME, 281
vc::BasicPanel, 94
vc::BasicVCComponent, 106
vc::PanelA7A8ODS, 228
vc::PanelGroup, 251
vc::StandardSwitch, 289
RealizeMFD
vc::MDU, 196
RecallCriticalFormat
CRT, 128
RecallStatus
CRT, 126
RED
CRT.cpp, 350
ref
gnc::AccAssembly, 30
REF_C2S1
PanelC2.h, 474
REF_C2S10
PanelC2.h, 474
REF_C2S11
PanelC2.h, 474
REF_C2S12
PanelC2.h, 474
REF_C2S2
PanelC2.h, 474
REF_C2S3
PanelC2.h, 474
REF_C2S4
PanelC2.h, 474
REF_C2S5
PanelC2.h, 474
REF_C2S6
PanelC2.h, 474
REF_C2S7
PanelC2.h, 474
REF_C2S8
PanelC2.h, 474
REF_C2S9
PanelC2.h, 474
REF_C3S1

PanelC3.h, 477
REF_C3S19
PanelC3.h, 477
REF_C3S2
PanelC3.h, 477
REF_C3S20
PanelC3.h, 477
REF_C3S8
PanelC3.h, 477
REF_C3S9
PanelC3.h, 477
REF_O3S12
PanelO3.h, 484
reference
vc::BasicVCComponent, 109
RefPoints, 271
Pitch, 271
Roll, 271
Yaw, 271
RegisterComponent
AtlantisSubsystem, 79
ISubsystem, 175
vc::BasicVCComponent, 106
RegisterMDU
Atlantis, 53
RegisterMFDContext
vc::MDU, 194
RegisterVC
GearOp, 158
Keyboard, 178
PanelA4, 223
PanelA8, 232
PanelC2, 235
PanelC3, 238
PanelF7, 246
PanelO3, 253
PanelR2, 257
PayloadBayOp, 265
vc::BasicPanel, 94
vc::BasicVCComponent, 106
vc::MDU, 196
vc::PanelA7A8ODS, 228
vc::PanelF6, 241
vc::PanelF7, 244
vc::PanelF8, 248
vc::PanelGroup, 251
RemoveBundle
discsignals::DiscreteBundleManager, 143
render_cockpit
Atlantis, 66
ReplaceComponent
vc::BasicPanel, 94
REQD_ATT
Atlantis, 66
ReqdRates
Atlantis, 66
RequestCMD
mps::EIU, 148
RequestDATA
mps::EIU, 148
reset_mmu
Atlantis, 66
reset_sat
Atlantis, 66
ResetEventTimer
dps::MasterTimingUnit, 187
ResetLine
discsignals::DiscOutPort, 139
ResetMET
dps::MasterTimingUnit, 187
resource.h
IDB_7SEGDIGITS, 493
IDB_CLOCKDIGITS, 493
IDB_DEUCHARACTERS, 493
IDB_TKBKLABEL, 493
IDC_BUTTON1, 493
IDC_BUTTON10, 493
IDC_BUTTON14, 493
IDC_BUTTON15, 493
IDC_BUTTON16, 493
IDC_BUTTON17, 493
IDC_BUTTON2, 493
IDC_BUTTON5, 493
IDC_BUTTON6, 493
IDC_BUTTON9, 493
IDC_CAM_BL, 493
IDC_CAM_BR, 493
IDC_CAM_DOWN, 493
IDC_CAM_FL, 493
IDC_CAM_FR, 493
IDC_CAM_HIGHRATE, 493
IDC_CAM_LEFT, 493
IDC_CAM_LOWRATE, 493
IDC_CAM_RIGHT, 493
IDC_CAM_ROT, 493
IDC_CAM_SEL, 493
IDC_CAM_UP, 493
IDC_DEG01, 493
IDC_DEG1, 493
IDC_DEG10, 493
IDC_ELBOW_PITCHDOWN, 493
IDC_ELBOW_PITCHUP, 493
IDC_GRAPPLE, 493
IDC_KU, 493
IDC_KU_DIRECT, 493
IDC_KU_TLKBK, 493
IDC_LATCHA, 493
IDC_LATCHB, 493

IDC_LATCHP_TLKBK, 493
IDC_LATCHS_TLKBK, 493
IDC_MECH1, 493
IDC_MECH2, 493
IDC_PAYCAM, 493
IDC_PAYLOAD, 493
IDC_PLBAYOP, 493
IDC_PLBD, 493
IDC_PLBD1, 493
IDC_PLBD2, 493
IDC_PLBD_TLKBK, 493
IDC_RADA, 493
IDC_RADB, 493
IDC_RADP_TLKBK, 493
IDC_RADS_TLKBK, 493
IDC_RELEASE, 493
IDC_RMSOP, 493
IDC_SHOULDER_PITCHDOWN, 493
IDC_SHOULDER_PITCHUP, 493
IDC_SHOULDER_YAWLEFT, 493
IDC_SHOULDER_YAWRIGHT, 493
IDC_SHOWGRAPPLE, 493
IDC_STOW, 493
IDC_TRANS_MX, 493
IDC_TRANS_MY, 493
IDC_TRANS_MZ, 493
IDC_TRANS_PX, 493
IDC_TRANS_PY, 493
IDC_TRANS_PZ, 493
IDC_WRIST_PITCHDOWN, 493
IDC_WRIST_PITCHUP, 493
IDC_WRIST_ROLLLEFT, 493
IDC_WRIST_ROLLRIGHT, 493
IDC_WRIST_YAWLEFT, 493
IDC_WRIST_YAWRIGHT, 493
IDD_CTRL, 493
IDD_PAYLOADCAM, 493
IDD_PLBAY, 493
IDD_RMS, 493
IDI_DOWN, 493
IDI_LEFT, 493
IDI_RIGHT, 493
IDI_RLEFT, 493
IDI_RRIGHT, 493
IDI_UP, 493
RETUmbDoorStatus
PanelR2, 260
RevertGVA
SSU_FSS_RSS, 285
RevertKuAntennaAction
PayloadBayOp, 267
RevertLandingGear
GearOp, 159
RevertOAA
SSU_FSS_RSS, 285
RevertRSS
SSU_FSS_RSS, 285
RevertSpeedbrake
Atlantis, 53
rh
Atlantis, 66
rh0
Atlantis, 66
right_deploy
AirDataProbeSystem, 34
RIGHT_DOOR
PanelR2, 260
RIGHT_LATCH
PanelR2, 260
right_mode
AirDataProbeSystem, 34
right_state
AirDataProbeSystem, 34
right_stow_state
AirDataProbeSystem, 34
rightPod
OMSSubsystem, 221
Rigidize
Atlantis, 66
rmh
Atlantis, 66
RMS
Atlantis, 66
rms_anim
Atlantis, 66
rms_attach
Atlantis, 60
RMS_DlgProc
Atlantis, 57
Atlantis.cpp, 307
rms_rollout_anim
Atlantis, 66
RMS_SEL
CRT, 130
RMSRollout
Atlantis, 66
ROLL
Atlantis.h, 323
Roll
Atlantis, 66
RefPoints, 271
roll
Atlantis, 66
RollActive
Atlantis, 60
RollOff
Atlantis.cpp, 310
RollToHeadsUp

Atlantis, 66
ROT
Atlantis, 66
RotateVector
Atlantis, 55
RotateVectorPYR
Atlantis, 55
RotateVectorX
Atlantis, 55
RotateVectorY
Atlantis, 55
RotateVectorZ
Atlantis, 55
RotRate
Atlantis, 66
RSS_OPERATING_SPEED
SSU_FSS_RSS.h, 397
RSS_proc
SSU_FSS_RSS, 285
RSS_status
SSU_FSS_RSS, 285
Run
dps::IDPSoftware, 171
RussianCB
vc::RussianCB, 273
rv
Atlantis, 66

sat_anim
Atlantis, 66
sat_attach
Atlantis, 60
sat_ref
Atlantis, 66
SatGrappled
Atlantis, 55
SatStowed
Atlantis, 55
SaveMission
mission::Mission, 204
saveprm
CRT, 130
SaveState
GearOp, 159
KuBandSystem, 181
OMSSubsystem, 221
PanelA4, 224
PanelA8, 233
PanelC2, 236
PanelC3, 238
PanelF7, 246
PanelO3, 254
PanelR2, 258
PayloadBayOp, 268
SSU_FSS_RSS, 285
SubsystemDirector, 296
SCU_PROM
dps::MDM, 191
sDigit
vc::EventTimer, 151
sec_idp
vc::MDU, 198
SeparateBoosters
Atlantis, 53
SeparateMMU
Atlantis, 53
SeparateTank
Atlantis, 53
SetAID
vc::BasicPanel, 94
SetAirDataProbeDeployment
Atlantis, 53
SetAnimation
vc::BasicVCComponent, 104
SetAnimationArm
Atlantis, 53
SetAnimationCameras
Atlantis, 56
SetAnimationIKArm
Atlantis, 53
SetBayDoorLatchPosition
Atlantis, 53
SetBayDoorPosition
Atlantis, 53
SetClassCaps
OMSSubsystem, 221
SSU_FSS_RSS, 285
SubsystemDirector, 296
SetCOTime
mps::SSME, 278
SetDeployMode
AirDataProbeSystem, 33
SetDirection
vc::BasicVCComponent, 106
SetDiscrete
discsignals::DiscreteBundle, 141
SetDisplayTitle
CRT, 126
SetDoorAction
PayloadBayOp, 266
SetETUmbDoorAction
PanelR2, 258
SetETUmbDoorPosition
Atlantis, 53
SetEventTimer
dps::MasterTimingUnit, 187
SetEventTimerMode
dps::MasterTimingUnit, 187

SetGearParameters
GearOp, 159
SetHasOwnVCMesh
vc::BasicPanel, 93
SetILoads
Atlantis, 55
SetInitialAnimState
vc::BasicVCComponent, 105
SetKuAntennaAction
PayloadBayOp, 267
SetKuAntennaPosition
Atlantis, 53
SetKuGimbalAngles
Atlantis, 53
SetLastCreatedMFD
Atlantis, 53
SetLaunchConfiguration
Atlantis, 53
SetLine
discsignals::DiscOutPort, 138
SetLogMode
discsignals::DiscreteBundle, 140
SetMotorParams
MechActuator, 201
SetMouseRegion
vc::BasicVCComponent, 105
SetN2TankFillState
OMSPod, 218
SetObjectAnim
MechActuator, 201
SetObjectParams
MechActuator, 201
SetOMSThrusterPos
OMSPod, 218
SetOrbiterConfiguration
Atlantis, 53
SetOrbiterTankConfiguration
Atlantis, 53
SetOrientation
vc::BasicSwitch, 98
SetParentPanel
vc::BasicVCComponent, 105
SetPostLaunchConfiguration
Atlantis, 53
SetPrimaryIDP
vc::MDU, 195
SetRadiatorAction
PayloadBayOp, 267
SetRadiatorPosition
Atlantis, 53
SetRadLatchAction
PayloadBayOp, 267
SetRadLatchPosition
Atlantis, 53
SetRates
Atlantis, 55
SetReference
vc::BasicVCComponent, 106
SetRefTime
Atlantis_SRB, 68
SetSecondaryIDP
vc::MDU, 195
SetSlew
KuBandSystem, 181
SetSpeedbrake
Atlantis, 53
SetWord
dps::RAM16, 270
sEventMinutes
dps::MasterTimingUnit, 189
sEventSeconds
dps::MasterTimingUnit, 189
sGMTDays
dps::MasterTimingUnit, 189
sGMTHours
dps::MasterTimingUnit, 189
sGMTMillis
dps::MasterTimingUnit, 189
sGMTMinutes
dps::MasterTimingUnit, 189
sGMTSeconds
dps::MasterTimingUnit, 189
shBackGround
vc::MDU, 198
shLabelTex
vc::MDU, 198
shoulder_brace
Atlantis, 66
SHOULDER_BRACE_SPEED
Atlantis.h, 328
shoulder_max
Atlantis, 66
shoulder_min
Atlantis, 66
shoulder_neutral
Atlantis, 66
shoulder_pos
Atlantis, 66
shoulder_range
Atlantis, 66
ShowMidDeck
Atlantis, 53
ShowODS
Atlantis, 53
Shutdown
mps::SSME, 277
SHUTDOWN_FAILSAFEPNEUMATIC
mps, 22

SHUTDOWN_PROPELLANTVALVESCLOSED
mps, 22
SHUTDOWN_THROTTLINGTOZEROTHRUST
mps, 22
ShutdownEnable
mps::_DATA_TABLE, 28
ShuttleBus
dps::ShuttleBus, 274
ShuttleBus.h
AFX_SHUTTLEBUS_H__5E15D9D6_-08AF_49BD_96FB_2835793F608E__-INCLUDED_, 381
SidDay
Atlantis, 66
SignalGSEBreakHDP
Atlantis, 53
SignalGSEStart
Atlantis, 53
MLP, 206
Simtime
CRT, 130
Simtime_last
CRT, 130
slag1
Atlantis, 60
Atlantis_SRB, 69
SLAG1_RATE
Atlantis.h, 328
slag2
Atlantis, 60
Atlantis_SRB, 69
SLAG2_RATE
Atlantis.h, 328
slag3
Atlantis, 60
Atlantis_SRB, 69
SLAG3_RATE
Atlantis.h, 328
SLAG_DIR
Atlantis.h, 328
SLAG_SPREAD
Atlantis.h, 328
SM
dps, 15
sm_ops
dps::IDP, 168
sm_specdisp
dps::IDP, 168
sMETDays
dps::MasterTimingUnit, 189
PanelA4, 224
PanelO3, 254
sMETHours
dps::MasterTimingUnit, 189
PanelA4, 224
PanelO3, 254
sMETMillis
dps::MasterTimingUnit, 189
sMETMinutes
dps::MasterTimingUnit, 189
PanelA4, 224
PanelO3, 254
sMETSeconds
dps::MasterTimingUnit, 189
PanelA4, 224
PanelO3, 254
SMOps
Atlantis, 66
snLines
discsignals::DiscreteBundle, 141
sOMSFuel
PanelO3, 254
sp_angle
Atlantis, 66
sp_null
Atlantis, 66
spdb_proc
Atlantis, 58
spdb_status
Atlantis, 57
spdb_tgt
Atlantis, 60
SPEC
KeyboardInput, 179
spec
CRT, 130
CRT::SavePrm, 132
SPEEDBRAKE_OPERATING_SPEED
Atlantis.h, 328
SPI
CRT, 126
srb_contrail
Common.cpp, 346
SRB_CUTOUT_TIME
Atlantis.h, 328
SRB_EMPTY_MASS
Atlantis.h, 328
srb_exhaust
Common.cpp, 346
srb_id1
Atlantis, 58
srb_id2
Atlantis, 58
SRB_ISP0
Atlantis.h, 328
SRB_ISP1
Atlantis.h, 328
SRB_MAX_PROPELLANT_MASS

Atlantis.h, 328
SRB_nt
Atlantis.h, 328
SRB_Prop
Common.cpp, 346
SRB_SEPARATION_TIME
Atlantis.h, 328
srb_separation_time
Atlantis_SRB, 69
SRB_Seq
Common.cpp, 346
srb_slag1
Common.cpp, 347
srb_slag2
Common.cpp, 347
srb_slag3
Common.cpp, 347
SRB_STABILISATION_TIME
Atlantis.h, 328
SRB_THRUST
Atlantis.h, 328
SRB_Thrust
Common.cpp, 347
sScratchPadLength
dps::IDP, 168
SSME
mps::SSME, 277
SSME_DATA_TABLE
mps::SSME, 283
SSME_RATED_THRUST
Atlantis.h, 328
SSMEL_GOX_REF
Atlantis.h, 328
SSMEL_GOX_REF1
Atlantis, 66
SSMEL_REF
Atlantis.h, 328
SSMER_GOX_REF
Atlantis.h, 328
SSMER_GOX_REF1
Atlantis, 66
SSMER_REF
Atlantis.h, 328
SSMET_GOX_REF
Atlantis.h, 328
SSMET_GOX_REF1
Atlantis, 66
SSMET_REF
Atlantis.h, 328
SSU_FSS_RSS
DOOR_CLOSED, 284
DOOR_CLOSING, 284
DOOR_OPEN, 284
DOOR_OPENING, 284
SSU_FSS_RSS, 284
anim_GVA, 285
anim_OAA, 285
anim_RSS, 285
DefineAnimations, 285
DoorStatus, 284
GVA_proc, 285
GVA_status, 285
LoadState, 285
OAA_proc, 285
OAA_status, 285
RevertGVA, 285
RevertOAA, 285
RevertRSS, 285
RSS_proc, 285
RSS_status, 285
SaveState, 285
SetClassCaps, 285
SSU_FSS_RSS, 285
SSU_FSS_RSS, 285
Timestep, 285
SSU_FSS_RSS.cpp
ovcConsumeBufferedKey, 396
ovcExit, 396
ovcInit, 396
ovcLoadStateEx, 396
ovcSaveState, 396
ovcSetClassCaps, 396
ovcTimestep, 396
STRICT, 396
SSU_FSS_RSS.h
GVA_OPERATING_SPEED, 397
OAA_OPERATING_SPEED, 397
RSS_OPERATING_SPEED, 397
SSU_FSS_RSS_EMPTY_MASS, 397
SSU_FSS_RSS_SIZE, 397
SSU_FSS_RSS_EMPTY_MASS
SSU_FSS_RSS.h, 397
SSU_FSS_RSS_SIZE
SSU_FSS_RSS.h, 397
stage1guidance
Atlantis, 66
stage1guidance_size
Atlantis, 66
StandardSwitch
vc::StandardSwitch, 288
StandardSwitch.h
AFX_STANDARDSWITCH_H__-
A87D233D_E5CF_4343_90D8_-
BF65F16C4DB4__INCLUDED_, 523
Start
Stopwatch, 294
START_TIME
Atlantis, 66

StartEnable
mps::_DATA_TABLE, 28
StartEventTimer
dps::MasterTimingUnit, 187
STARTMAINSTAGE_ELECTRICALLOCKUP
mps, 22
STARTMAINSTAGE_FIXEDDENSITY
mps, 22
STARTMAINSTAGE_HYDRAULICLOCKUP
mps, 22
STARTMAINSTAGE_-
MAINSTAGEPHASENORMALCONTROL
mps, 22
STARTMAINSTAGE_-
STARTPHASESTARTINITIATION
mps, 22
STARTMAINSTAGE_-
STARTPHASETHRUSTBUILDUP
mps, 22
STARTMAINSTAGE_THRUSTLIMITING
mps, 22
StartMET
dps::MasterTimingUnit, 187
STARTPREPARATION_ENGINEREADY
mps, 22
STARTPREPARATION_PURGESEQUENCE1
mps, 22
STARTPREPARATION_PURGESEQUENCE2
mps, 22
STARTPREPARATION_PURGESEQUENCE3
mps, 22
STARTPREPARATION_PURGESEQUENCE4
mps, 22
STATE_DESTROYED
Atlantis.h, 328
STATE_ORBITER
Atlantis.h, 328
STATE_PRELAUNCH
Atlantis.h, 328
STATE_STAGE1
Atlantis.h, 328
STATE_STAGE2
Atlantis.h, 328
STATUS
mps::_DATA_TABLE, 28
Status
Atlantis, 66
CRT, 130
status
Atlantis, 57
STD_CLASS_NAME
CRT.h, 353
StdAfx.h
AFX_STDAFX_H__C669E737_B06A_-
482C_B99B_41C360D81979__-
INCLUDED_, 405
StdSwitch2
vc::StdSwitch2, 291
StdSwitch3
vc::StdSwitch3, 293
SteerGimbal
Atlantis, 55
Step
GearOp, 158
PanelA4, 224
PanelA8, 232
PanelC2, 235
PanelC3, 238
PanelF7, 246
PanelO3, 253
PanelR2, 257
PayloadBayOp, 265
sTimerMinutes
PanelA4, 224
PanelF7, 246
PanelO3, 254
sTimerSeconds
PanelA4, 224
PanelF7, 246
PanelO3, 254
Stop
Stopwatch, 294
StopAllManifolds
Atlantis, 53
StopEventTimer
dps::MasterTimingUnit, 187
Stopwatch, 294
~Stopwatch, 294
liCurrent, 294
liFrequency, 294
liLast, 294
Start, 294
Stop, 294
Stopwatch, 294
StoreStatus
CRT, 126
STOW_ENABLE_STATE
AirDataProbeSystem, 32
strCompA2G
CommModeHandler, 119
strDecompA2G
CommModeHandler, 119
strDecompG2A
CommModeHandler, 119
STRICT
Atlantis.cpp, 306
AtlantisConfig.cpp, 338

CRT.cpp, 350
SSU_FSS_RSS.cpp, 396
strMissionName
mission::Mission, 204
strNegAnswer_Comp
CommModeHandler, 119
strNegAnswer_Decomp
CommModeHandler, 119
strOrbiter
mission::Mission, 204
strPosAnswer_Comp
CommModeHandler, 119
strPosAnswer_Decomp
CommModeHandler, 119
STS
AtlantisSubsystem, 78
SubsystemDirector, 295
vc::BasicPanel, 93
vc::BasicVCComponent, 104
sts
CRT, 130
GearOp, 160
Keyboard, 178
PanelA4, 224
PanelA8, 233
PanelC2, 236
PanelC3, 239
PanelF7, 246
PanelO3, 254
PanelR2, 260
PayloadBayOp, 269
vc::BasicVCComponent, 109
sts_dlg
PlBayOp.cpp, 488
SubsystemDirector, 295
~SubsystemDirector, 295
AddSubsystem, 295
BundleManager, 296
ParseScenarioLine, 296
PlaybackEvent, 296
PostStep, 296
PreStep, 296
psts, 296
SaveState, 296
SetClassCaps, 296
STS, 295
SubsystemDirector, 295
subsystems, 296
WriteLog, 296
subsystems
SubsystemDirector, 296
SW_NOT_PUSH
GearOp, 157
SW_PUSH
GearOp, 157
SWC_CLOSE
GearOp, 157
SWC_OPEN
GearOp, 157
SWITCH1
Atlantis.h, 322
SWITCH10
Atlantis.h, 322
SWITCH11
Atlantis.h, 322
SWITCH12
Atlantis.h, 322
SWITCH13
Atlantis.h, 322
SWITCH14
Atlantis.h, 322
SWITCH15
Atlantis.h, 322
SWITCH16
Atlantis.h, 322
SWITCH17
Atlantis.h, 322
SWITCH18
Atlantis.h, 322
SWITCH19
Atlantis.h, 322
SWITCH2
Atlantis.h, 322
SWITCH20
Atlantis.h, 322
SWITCH3
Atlantis.h, 322
SWITCH4
Atlantis.h, 322
SWITCH5
Atlantis.h, 322
SWITCH6
Atlantis.h, 322
SWITCH7
Atlantis.h, 322
SWITCH8
Atlantis.h, 322
SWITCH9
Atlantis.h, 322
SWITCH_HOLD_TIME
Atlantis.h, 328
SWITCH_O3S12
PanelO3.h, 484
switch_state
PanelA4, 224
PanelA8, 233
PanelC2, 236
PanelO3, 254

switch_timer
PanelA4, 224
PanelC2, 236
PanelO3, 254
SwitchCover
GearOp, 160
SwitchMFDMode
vc::MDU, 194
SwitchPush
GearOp, 160
sy_angle
Atlantis, 66

T
Atlantis, 66
FLOWSTATE, 154
t0
Atlantis, 58
Atlantis_SRB, 69
vc::MDU, 198
tableterp
Atlantis.cpp, 307
GPC.cpp, 401
TailThrusters
Atlantis, 66
TANK_EMPTY_MASS
Atlantis.h, 328
TANK_MAX_PROPELLANT_MASS
Atlantis.h, 328
TAp
Atlantis, 66
target_pitch
Atlantis, 66
target_roll
Atlantis, 66
target_yaw
Atlantis, 66
TargetAttM50
Atlantis, 66
TargetAttOrbiter
Atlantis, 66
tau
Atlantis, 66
TEcc
Atlantis, 66
TEX_A1SHADOW_VC
meshres_vc_additions.h, 446
TEX_A8BOTTOM_VC
meshres_vc_additions.h, 446
TEX_A8TOP_VC
meshres_vc_additions.h, 446
TEX_AFTTIMER_VC
meshres_vc_additions.h, 446
TEX_ATLANTIS
meshres_vc_additions.h, 446
TEX_CLOCKNUMS_VC
meshres_vc_additions.h, 446
TEX_COAS_VC
meshres_vc_additions.h, 446
TEX_DIGITS_VC
meshres_vc_additions.h, 446
TEX_FPANELS_VC
meshres_vc_additions.h, 446
TEX_GPC_VC
meshres_vc_additions.h, 446
tex_hires_disabled
AtlantisConfig.cpp, 338
tex_hires_enabled
AtlantisConfig.cpp, 338
TEX_LABEL_VC
meshres_vc_additions.h, 446
TEX_PANELA1L_VC
meshres_vc_additions.h, 446
TEX_PANELA1R_VC
meshres_vc_additions.h, 446
TEX_PANELA1U_VC
meshres_vc_additions.h, 446
TEX_PANELC2_VC
meshres_vc_additions.h, 446
TEX_PANELC3_VC
meshres_vc_additions.h, 446
TEX_PANELF6_VC
meshres_vc_additions.h, 446
TEX_PANELF8_VC
meshres_vc_additions.h, 446
TEX_PANELO1_VC
meshres_vc_additions.h, 446
TEX_PANELO3_VC
meshres_vc_additions.h, 446
TEX_PANELR2_VC
meshres_vc_additions.h, 446
TEX_R13L_VC
meshres_vc_additions.h, 446
TEX_R1L1_VC
meshres_vc_additions.h, 446
tex_rcs
Atlantis, 62
TEX_STSVC01_VC
meshres_vc_additions.h, 446
TEX_STSVC02_VC
meshres_vc_additions.h, 446
TEX_STSVC03_VC
meshres_vc_additions.h, 446
TEX_TALKBACK_VC
meshres_vc_additions.h, 446
TEX_TIMER_VC
meshres_vc_additions.h, 446
TexEnableHires

AtlantisConfig, 75
TexHiresEnabled
AtlantisConfig, 74
textBuffer
vc::MDU, 198
TGO
CRT, 130
TGT_ID
Atlantis, 66
TgtAlt
Atlantis, 66
TgtFPA
Atlantis, 66
TgtInc
Atlantis, 66
TgtLAN
Atlantis, 66
TgtRad
Atlantis, 66
TgtSpd
Atlantis, 66
tgtwheel_state
PanelC2, 236
th_att_lin
Atlantis, 62
th_att_rcs
Atlantis, 62
th_bolt
Atlantis_SRB, 69
th_main
Atlantis, 62
Atlantis_SRB, 69
th_oms
Atlantis, 62
th_srb
Atlantis, 62
th_ssme_gox
Atlantis, 62
th_ssme_loxdump
Atlantis, 62
thBSM
Atlantis_SRB, 69
THeading
Atlantis, 66
theta
Atlantis, 66
thetah
Atlantis, 66
ThetaT
Atlantis, 66
thetaT
Atlantis, 66
thg_main
Atlantis, 66
thg_retro
Atlantis, 66
thg_srb
Atlantis, 66
ThisVessel
Atlantis, 60
thManFRCS1
Atlantis, 63
thManFRCS2
Atlantis, 63
thManFRCS3
Atlantis, 63
thManFRCS4
Atlantis, 63
thManFRCS5
Atlantis, 63
thManLRCS1
Atlantis, 63
thManLRCS2
Atlantis, 63
thManLRCS3
Atlantis, 63
thManLRCS4
Atlantis, 63
thManLRCS5
Atlantis, 63
thManRRCS1
Atlantis, 63
thManRRCS2
Atlantis, 64
thManRRCS3
Atlantis, 64
thManRRCS4
Atlantis, 64
thManRRCS5
Atlantis, 64
ThrAngleP
Atlantis, 66
ThrAngleY
Atlantis, 66
Throttle
Atlantis, 55
Throttle_Bucket
Atlantis, 66
Throttling
mps::SSME, 278
Thrust
Atlantis, 66
Thrusters
Atlantis, 66
thSSME
mps::SSME, 283
TIG
Atlantis, 66

tig
Atlantis, 66
timeCO
mps::_DATA_TABLE, 28
timeESC
mps::_DATA_TABLE, 28
TIMER_AFT
dps, 15
TIMER_FORWARD
dps, 15
Timestep
SSU_FSS_RSS, 285
timeTAG
mps::_DATA_TABLE, 28
tkbk_label
GDIParams, 155
tkbk_state
GearOp, 160
PanelA8, 233
PanelR2, 260
PayloadBayOp, 269
TLastAB
Atlantis, 66
TLastMajorCycle
Atlantis, 66
TMajorCycle
Atlantis.h, 328
tMECO
Atlantis, 66
tmestp
BasicValve, 101
ValveTypeBool, 298
ToDeg
Atlantis, 55
CRT, 126
ToggleArrest
Atlantis, 53
ToggleCoordinateDisplayMode
vc::BasicPanel, 96
vc::PanelGroup, 251
ToggleGrapple
Atlantis, 53
ToggleVCMode
Atlantis, 53
ToRad
Atlantis, 55
CRT, 126
Torque
Atlantis, 66
TPe
Atlantis, 66
TPEGStop
Atlantis.h, 328
tpir
Atlantis.cpp, 307
GPC.cpp, 402
TrackRates
Atlantis, 66
TriggerHDP
MLP, 206
TriggerLiftOff
Atlantis, 57
Trim
Atlantis, 66
trimExt
Atlantis.cpp, 310
TRK
Atlantis, 66
TrkRate
Atlantis, 66
TRKROT_OPTION
Atlantis, 66
tSRBSep
Atlantis, 66
TTrA
Atlantis, 66
TV_ROLL
Atlantis, 66

UMBDOOR_AXIS
Atlantis.h, 328
UMBDOORL_REF
Atlantis.h, 328
UMBDOORR_REF
Atlantis.h, 328
UNIVPTG
CRT, 126
Update
CRT, 124
UpdateDAP
Atlantis, 55
CRT, 127
UpdateDialog
PayloadBayOp, 268
UpdateMesh
Atlantis, 53
GearOp, 158
PanelA4, 223
PanelA8, 232
PanelC2, 235
PanelC3, 238
PanelF7, 246
PanelO3, 253
PanelR2, 257
UpdateMPMMicroswitches
Atlantis, 55
UpdateMRLMicroswitches
Atlantis, 55

UpdateRMSAngles
Atlantis, 53
UpdateRMSPositions
Atlantis, 53
UpdateSSMEGimbalAnimations
Atlantis, 53
UpdateStatus
CRT, 130
UpdateTextBuffer
vc::MDU, 197
UpdateVC
GearOp, 158
PanelA4, 223
PanelA8, 232
PanelC2, 235
PanelC3, 238
PanelF7, 246
PanelO3, 253
PanelR2, 257
PayloadBayOp, 265
usCurrentPlayerChar
Atlantis, 62
usCurrentPosition
vc::BasicSwitch, 99
usDISP
dps::IDP, 168
usDisp
CRT, 130
usGPCDriver
CRT, 130
dps::IDP, 168
usGPCMode
KuBandSystem, 181
usIDPID
dps::IDP, 168
usLastMDUID
Atlantis, 66
usLine
discsignals::DiscretePort, 145
usMDU
CRT, 130
usMDUID
vc::MDU, 198
usMode
vc::EventTimer, 151
usNumPositions
vc::BasicSwitch, 99
usOPS
dps::IDP, 168
usPageNumber
CRT, 130
usSelectedFC
dps::IDP, 168
usSPEC
dps::IDP, 168

v
Atlantis, 66
ValveTypeBool, 297
~ValveTypeBool, 297
_backdoor, 298
Close, 297
GetPos, 298
Open, 297
pos, 298
tmestp, 298
ValveTypeBool, 297
vlv, 298
VARSTATE_HIGH
Atlantis.h, 328
VARSTATE_LOW
Atlantis.h, 328
VARSTATE_MISSING
Atlantis.h, 328
VARSTATE_OFFSCALE_HIGH
Atlantis.h, 328
VARSTATE_OFFSCALE_LOW
Atlantis.h, 328
VARSTATE_OK
Atlantis.h, 328
vc, 24
BitmapLocation, 25
MDUID_AFD, 25
MDUID_CDR1, 25
MDUID_CDR2, 25
MDUID_CRT1, 25
MDUID_CRT2, 25
MDUID_CRT3, 25
MDUID_CRT4, 25
MDUID_MFD1, 25
MDUID_MFD2, 25
MDUID_PLT1, 25
MDUID_PLT2, 25
PMDU, 25
vc::BasicLight, 89
~BasicLight, 90
BasicLight, 90
Connect, 90
input, 90
vc::BasicPanel, 91
~BasicPanel, 92
Add, 94
AddAIDToMouseEventList, 93
AddMeshes, 94
availableForMouse, 96
BasicPanel, 92
bCoordinateDisplayMode, 96
bHasOwnVCMesh, 96

components, 96
CreateSwitch2, 93
CreateSwitch3, 93
CreateSwitch3H, 93
DefinePanelMouseArea, 93
DefineSwitches, 93
DefineVC, 94
DefineVCAnimations, 93
DeleteAllComponents, 93
DisableCoordinateDisplayMode, 96
DistributeMouseEvent, 93
EnableCoordinateDisplayMode, 96
FindComponent, 93
GetIdentifier, 96
GetQualifiedIdentifier, 96
GetVCMeshIndex, 94
HasOwnVCMesh, 96
HasSwitch, 96
local_vcidx, 96
name, 96
OnParseLine, 96
OnPostStep, 96
OnPreStep, 95
OnPropagate, 96
OnSaveState, 96
OnVCMouseEvent, 94
OnVCRedrawEvent, 95
own_aid, 96
psts, 96
Realize, 94
RegisterVC, 94
ReplaceComponent, 94
SetAID, 94
SetHasOwnVCMesh, 93
STS, 93
ToggleCoordinateDisplayMode, 96
vc::BasicSwitch, 97
~BasicSwitch, 98
BasicSwitch, 98
bOrientation, 99
DefineVCAnimations, 98
GetLabel, 98
labels, 99
OnMouseEvent, 98
OnPositionChange, 99
OnPositionDown, 99
OnPositionUp, 98
OnSaveState, 99
SetOrientation, 98
usCurrentPosition, 99
usNumPositions, 99
vc::BasicVCComponent, 102
~BasicVCComponent, 104
AddAIDToRedrawEventList, 105
availableForRendering, 109
BasicVCComponent, 104
bHasAnimations, 109
bHasDirection, 109
bHasReference, 109
cache_lastx, 109
cache_lasty, 109
cache_mx, 109
cache_my, 109
CountComponents, 107
DefineVCAnimations, 105
DelComponent, 106
dir, 109
fInitialAnimState, 109
GetDirection, 106
GetIdentifier, 108
GetQualifiedIdentifier, 107
GetReference, 106
GetState, 105
ident, 109
InitialAnimState, 104
IsOwnRegion, 106
IsPointOver, 106
OnDumpToLog, 107
OnFail, 108
OnMouseEvent, 105
OnParseLine, 107
OnPlaybackEvent, 107
OnPostStep, 107
OnPreStep, 107
OnPropagate, 107
OnSaveState, 107
OnVCRedrawEvent, 107
p_a, 109
p_b, 109
p_c, 109
p_d, 109
pParent, 109
ProjectMouse, 106
Realize, 106
reference, 109
RegisterComponent, 106
RegisterVC, 106
SetAnimation, 104
SetDirection, 106
SetInitialAnimState, 105
SetMouseRegion, 105
SetParentPanel, 105
SetReference, 106
STS, 104
sts, 109
VerifyAnimations, 105
WriteScenarioInt, 105
WriteScenarioString, 105

WriteScenarioVector, 105
vc::EventTimer, 150
~EventTimer, 151
dilModeA, 151
dilModeB, 151
dilReset, 151
dilSet, 151
dilStart, 151
dilTest, 151
EventTimer, 151
fEventTime, 151
fRunning, 151
sDigit, 151
usMode, 151
vc::Indicator7Seg, 172
~Indicator7Seg, 173
Indicator7Seg, 173
vc::LockableLever, 182
~LockableLever, 183
anim_switch, 183
bIsPulled, 183
fInitialMousePos, 183
grpIndex, 183
input, 183
LockableLever, 183
pswitchpull, 183
pswitchrot, 183
vc::MDU, 192
~MDU, 194
Atlantis, 57
bInverseX, 198
bIsConnectedToCRTMFD, 198
bPower, 198
bUseCRTMFD, 198
bUseSecondaryPort, 198
ConnectToCRTMFD, 197
counting, 198
DefineRegionAID, 196
DefineVCAnimations, 195
DefineVCGroup, 196
DefineVCTexture, 196
fBrightness, 198
GetDrivingIDP, 197
GetEdgekeyMenu, 197
GetEdgeKeyMenuLabel, 195
GetPortConfig, 197
GetSelectedPort, 197
GetViewAngle, 197
IsCRTBufferEnabled, 196
MDU, 194
MDU_ADI, 194
MDU_ADI_AVVI, 194
MDU_COMP_ADI_HSI, 194
MDU_CONFIGSTATUS, 194
MDU_CST, 194
MDU_DPS, 194
MDU_FAULTSUMM, 194
MDU_HSI_AMI, 194
MDU_HYD_APU, 194
MDU_MEMORYMGMT, 194
MDU_OMS_MPS, 194
MDU_PORT_SELECT, 194
MDU_SPI, 194
MDU_MODE, 194
mfdspec, 198
myGroup, 198
OnMouseEvent, 195
Paint, 195
PaintEdgeMenu, 195
prim_idp, 198
PrintToBuffer, 197
RealizeMFD, 196
RegisterMFDContext, 194
RegisterVC, 196
sec_idp, 198
SetPrimaryIDP, 195
SetSecondaryIDP, 195
shBackGround, 198
shLabelTex, 198
SwitchMFDMode, 194
t0, 198
textBuffer, 198
UpdateTextBuffer, 197
usMDUID, 198
vc::PanelA7A8ODS, 225
~PanelA7A8ODS, 227
AddMeshes, 227
DefineVC, 227
DefineVCAnimations, 227
GetVCMeshIndex, 228
midx_odspanel, 229
PanelA7A8ODS, 227
pARLKTNLFanA, 229
pARLKTNLFanB, 229
pControlPanelPowerA, 229
pControlPanelPowerB, 229
pControlPanelPowerC, 229
pLightsAirlock14, 229
pLightsAirlock23, 229
pLightsCLVestPORT, 229
pLightsCLVestSTBD, 229
pLightsDockingAFT, 229
pLightsDockingFWD, 229
pPSUPowerMNA, 229
pPSUPowerMNB, 229
pPyroPowerMNA, 229
pPyroPowerMNC, 229
pPyrosAp, 229

pPyrosBp, 229
pPyrosCp, 229
pSys1Vent, 229
pSys1VentIsol, 229
pSys2Vent, 229
pSys2VentIsol, 229
pSystemPowerMNA, 229
pSystemPowerMNB, 229
Realize, 228
RegisterVC, 228
vc::PanelF6, 240
~PanelF6, 241
DefineVC, 241
pADIAttitude, 242
pADIError, 242
pADIRate, 242
pAirData, 242
PanelF6, 241
pCDR1, 242
pCDR2, 242
pFltCntlrPower, 242
pHSISelectChannel, 242
pHSISelectMode, 242
pHSISelectSource, 242
pRdrAltm, 242
RegisterVC, 241
vc::PanelF7, 243
~PanelF7, 244
DefineVC, 244
PanelF7, 244
pCRT1, 244
pCRT2, 244
pCRT3, 244
pMFD1, 244
pMFD2, 244
RegisterVC, 244
vc::PanelF8, 247
~PanelF8, 248
DefineVC, 248
pADIAttitude, 249
pADIError, 249
pADIRate, 249
pAirData, 249
PanelF8, 248
pFltCntlrPower, 249
pHSISelectChannel, 249
pHSISelectMode, 249
pHSISelectSource, 249
pPLT1, 249
pPLT2, 249
pRdrAltm, 249
RegisterVC, 248
vc::PanelGroup, 250
~PanelGroup, 251
AddPanel, 251
DefineVC, 251
DefineVCAnimations, 251
DisableCoordinateDisplayMode, 251
EnableCoordinateDisplayMode, 251
OnParseLine, 251
OnPostStep, 251
OnPreStep, 251
OnPropagate, 251
OnSaveState, 251
OnVCMouseEvent, 251
OnVCRedrawEvent, 251
PanelGroup, 251
panels, 251
Realize, 251
RegisterVC, 251
ToggleCoordinateDisplayMode, 251
vc::RussianCB, 272
~RussianCB, 273
DefineVCAnimations, 273
RussianCB, 273
vc::StandardSwitch, 287
~StandardSwitch, 288
anim_switch, 289
ConnectPort, 289
DefineSwitchGroup, 288
DefineVCAnimations, 288
grpIndex, 289
input, 289
OnPositionChange, 288
pswitchrot, 289
Realize, 289
StandardSwitch, 288
vc::StdSwitch2, 290
~StdSwitch2, 291
ConnectPort, 291
outPort, 291
StdSwitch2, 291
vc::StdSwitch3, 292
~StdSwitch3, 293
ConnectPort, 293
outPortA, 293
outPortB, 293
StdSwitch3, 293
vc::VCMOUSEEVENTPOINT, 299
x, 299
y, 299
VC_AFTPILOT
Atlantis.h, 328
VC_AFTWORKSTATION
Atlantis.h, 328
VC_CDR
Atlantis.h, 328
vc_defs.h

___SPACE_SHUTTLE_ULTRA_VC_-DEFINITIONS_H_INCLUDED___, 524
VC_DIR_AFTPILOT
Atlantis.h, 328
VC_DIR_AFTWORKSTATION
Atlantis.h, 330
VC_DIR_COAS
Atlantis.h, 330
VC_DIR_EXT_AL
Atlantis.h, 330
VC_DIR_MIDDECK
Atlantis.h, 330
VC_DIR_MS1
Atlantis.h, 330
VC_DIR_MS2
Atlantis.h, 330
VC_DIR_PORTSTATION
Atlantis.h, 330
VC_DIR_RMSSTATION
Atlantis.h, 330
VC_DIR_STBDSTATION
Atlantis.h, 330
VC_DOCKCAM
Atlantis.h, 330
VC_EXT_AL
Atlantis.h, 330
VC_LBL_AFTPILOT
Atlantis.h, 330
VC_LBL_AFTWORKSTATION
Atlantis.h, 330
VC_LBL_CDR
Atlantis.h, 330
VC_LBL_COAS
Atlantis.h, 330
VC_LBL_DOCKCAM
Atlantis.h, 330
VC_LBL_EXT_AL
Atlantis.h, 330
VC_LBL_LEECAM
Atlantis.h, 330
VC_LBL_MIDDECK
Atlantis.h, 330
VC_LBL_MS1
Atlantis.h, 330
VC_LBL_MS2
Atlantis.h, 330
VC_LBL_PLBCAMBL
Atlantis.h, 330
VC_LBL_PLBCAMBR
Atlantis.h, 330
VC_LBL_PLBCAMFL
Atlantis.h, 330
VC_LBL_PLBCAMFR
Atlantis.h, 330
VC_LBL_PLT
Atlantis.h, 330
VC_LBL_PORTSTATION
Atlantis.h, 330
VC_LBL_RMSSTATION
Atlantis.h, 330
VC_LBL_STBDSTATION
Atlantis.h, 330
VC_LEECAM
Atlantis.h, 330
VC_MIDDECK
Atlantis.h, 330
VC_MS1
Atlantis.h, 330
VC_MS2
Atlantis.h, 330
VC_MS3
Atlantis.h, 330
VC_MS4
Atlantis.h, 330
VC_MS5
Atlantis.h, 330
VC_OFSFWD_AFTPILOT
Atlantis.h, 330
VC_PLBCAMBL
Atlantis.h, 330
VC_PLBCAMBR
Atlantis.h, 330
VC_PLBCAMFL
Atlantis.h, 330
VC_PLBCAMFR
Atlantis.h, 330
VC_PLT
Atlantis.h, 330
VC_PORTSTATION
Atlantis.h, 330
VC_POS_AFTPILOT
Atlantis.h, 330
VC_POS_AFTWORKSTATION
Atlantis.h, 330
VC_POS_CDR
Atlantis.h, 330
VC_POS_COAS
Atlantis.h, 330
VC_POS_EXT_AL
Atlantis.h, 330
VC_POS_MIDDECK
Atlantis.h, 330
VC_POS_MS1
Atlantis.h, 330
VC_POS_MS2
Atlantis.h, 330
VC_POS_PLT

Atlantis.h, 330
VC_POS_PORTSTATION
Atlantis.h, 330
VC_POS_RMSSTATION
Atlantis.h, 330
VC_POS_STBDSTATION
Atlantis.h, 330
VC_RMSSTATION
Atlantis.h, 330
VC_STBDSTATION
Atlantis.h, 330
vcDeckMode
Atlantis, 58
VCDrawTalkback
GearOp, 160
PanelA8, 233
PanelR2, 258
PayloadBayOp, 269
VCM_AIRLOCK
Atlantis, 51
VCM_FLIGHTDECK
Atlantis, 51
VCM_MIDDECK
Atlantis, 51
VCMode
Atlantis, 57
VCMouseEvent
GearOp, 158
Keyboard, 178
PanelA4, 223
PanelA8, 232
PanelC2, 235
PanelC3, 238
PanelF7, 246
PanelO3, 253
PanelR2, 257
PayloadBayOp, 265
vcmsh_fname
AtlantisConfig.cpp, 338
VCRedrawEvent
GearOp, 158
PanelA4, 223
PanelA8, 232
PanelC2, 235
PanelC3, 238
PanelF7, 246
PanelO3, 253
PanelR2, 257
PayloadBayOp, 265
velMoon
Atlantis, 66
VerifyAnimations
vc::BasicVCComponent, 105
VERN_ATT_DB
DAPConfig, 134
VERN_CNTL_ACC
DAPConfig, 134
VERN_COMP
DAPConfig, 134
VERN_RATE_DB
DAPConfig, 134
VERN_ROT_PLS
DAPConfig, 134
VERN_ROT_RATE
DAPConfig, 134
vessel
CRT, 130
VGO
Atlantis, 66
vgoTot
Atlantis, 66
vh
Atlantis, 66
vis
Atlantis, 60
VLiftCoeff
Atlantis.cpp, 308
vlv
ValveTypeBool, 298
vr
Atlantis, 66
vtheta
Atlantis, 66
vv
Atlantis, 66
vY
gnc::AccAssembly, 30
vZ
gnc::AccAssembly, 30

wheel_tex
PanelC2, 236
wheelnumber
PanelC2, 236
wheelState
PanelC2, 236
WHITE
CRT.cpp, 350
width
CRT, 130
WingName
Atlantis, 57
wp_angle
Atlantis, 66
wp_wy
Atlantis, 66
wr_angle
Atlantis, 66

wrist_max
Atlantis, 66
wrist_min
Atlantis, 66
wrist_neutral
Atlantis, 66
wrist_range
Atlantis, 66
wrist_roll_max
Atlantis, 66
wrist_roll_min
Atlantis, 66
wrist_roll_neutral
Atlantis, 66
wrist_roll_range
Atlantis, 66
wrist_yaw_max
Atlantis, 66
wrist_yaw_min
Atlantis, 66
wrist_yaw_neutral
Atlantis, 66
wrist_yaw_range
Atlantis, 66
WriteImage
dps::RAM16, 270
WriteLog
SubsystemDirector, 296
WriteScenarioInt
vc::BasicVCComponent, 105
WriteScenarioString
vc::BasicVCComponent, 105
WriteScenarioVector
vc::BasicVCComponent, 105
WriteStatus
CRT, 125
WT
Atlantis, 66
wy_angle
Atlantis, 66
wy_ee
Atlantis, 66

x
vc::VCMOUSEEVENTPOINT, 299

Y
Atlantis, 66
y
vc::VCMOUSEEVENTPOINT, 299
YAW
Atlantis.h, 322
Yaw
Atlantis, 66
RefPoints, 271
yaw
Atlantis, 66
YawActive
Atlantis, 60
YawOff
Atlantis.cpp, 310
YELLOW
CRT.cpp, 350