JFT-145-1-DL.SDD

Rev B, 30-Sept-94

# SOFTWARE DESIGN DOCUMENT

# FOR THE

# DIS LIBRARY (DL) CSCI

# OF THE

# ADA DISTRIBUTED INTERACTIVE SIMULATION (ADIS) SUPPORT SYSTEM


CONTRACT NO.  N00421-92-D-0028

CDRL SEQUENCE NO:  A008


Prepared for:

Naval Air Warfare Center, Aircraft Division (NAWCAD)
Systems Engineering Test Directorate (SETD)
Manned Flight Simulator (MFS)

Prepared by:


J. F. Taylor, Inc.
Rt. 235 and Maple Rd.
Lexington Park, MD  20653


| Authenticated by: | Approved by: |
| --- | --- |
| (Contracting Agency) | (Contractor) |
| (Date) | (Date) |

**Table of Contents**


| Paragraph | Title Page |
|---|---|
| µ1 | Scope....1 |
| 1.1 | Identification....1 |
| 1.2 | System Overview....1 |
| 1.3 | Document Overview....2 |
| 2 | Applicable Documents....3 |
| 2.1 | Government Documents....3 |
| 2.2 | Non-Government Documents....3 |
| 3 | Preliminary Design....4 |
| 3.1 | DL CSCI Overview....4 |
| 3.1.1 | DL CSCI Architecture....5 |
| 3.1.2 | System States and Modes....10 |
| 3.1.3 | Memory and Processing Time Allocation....10 |
| 3.2 | DL CSCI Design Description....10 |
| 3.2.1 | Conversion Algorithms CSC (DL-CSC-1)....11 |
| 3.2.1.1 | Coordinate System Conversion CSC (DL-CSC-1.1)....11 |
| 3.2.1.2 | Orientation System Conversion CSC (DL-CSC-1.2)....13 |
| 3.2.2 | Dead-Reckoning Algorithms CSC (DL-CSC-2)....15 |
| 3.2.3 | Calculation Algorithms CSC (DL-CSC-3)....17 |
| 3.2.4 | Filtering Algorithms CSC (DL-CSC-4)....19 |
| 3.2.4.1 | Filtering Routines CSC (DL-CSC-4.1)....22 |
| 3.2.4.2 | Filter List By Distance CSC (DL-CSC-4.2)....22 |
| 3.2.4.3 | Filter List By Orientation CSC (DL-CSC-4.3)....22 |
| 3.2.4.4 | Filter List By Velocity CSC (DL-CSC-4.4)....22 |
| 3.2.5 | Hashing Algorithms CSC (DL-CSC-5)....22 |
| 3.2.6 | Smoothing Algorithms CSC (DL-CSC-6)....23 |
| 3.2.7 | Sorting Algorithms CSC (DL-CSC-7)....26 |
| 3.2.7.1 | Sort Routines CSC (DL-CSC-7.1)....26 |
| 3.2.7.2 | Sort By Distance CSC (DL-CSC-7.2)....28 |
| 3.2.7.3 | Sort By Velocity CSC (DL-CSC-7.3)....28 |
| 4 | Detailed Design....29 |
| 4.1 | Conversion Algorithms CSC (DL-CSC-1)....29 |
| 4.1.1 | Coordinate System Conversion CSC (DL-CSC-1.1)....29 |
| 4.1.1.1 | Geocentric-Geodetic Conversion CSU (DL-CSU-1.1.1)....33 |
| 4.1.1.1.1 | Geocentric-Geodetic Conversion CSU Design Specification/Constraints....33 |
| 4.1.1.1.2 | Geocentric-Geodetic Conversion CSU Design....33 |
| 4.1.1.1.2.1 | Geocentric-Geodetic Conversion CSU Input/Output Data Elements ....33 |
| 4.1.1.1.2.2 | Geocentric-Geodetic Conversion CSU Local Data Elements....33 |
| 4.1.1.1.2.3 | Geocentric-Geodetic Conversion CSU Global Data Elements....34 |
| 4.1.1.1.2.4 | Geocentric-Geodetic Conversion CSU Local and Shared Data |

Structures....34
4.1.1.1.2.5 Geocentric-Geodetic Conversion CSU Interrupts and Signals....34
4.1.1.1.2.6 Geocentric-Geodetic Conversion CSU Error Handling....34
4.1.1.1.2.7 Geocentric-Geodetic Conversion CSU Use of Other Elements....34
4.1.1.1.2.8 Geocentric-Geodetic Conversion CSU Logic Flow....34
4.1.1.1.2.9 Geocentric-Geodetic Conversion CSU Algorithms....36
4.1.1.1.2.10 Geocentric-Geodetic Conversion CSU Limitations....36
4.1.1.2 Geocentric-UTM Conversion CSU (DL-CSU-1.1.2)....36
4.1.1.2.1 Geocentric-UTM Conversion CSU Design Specification/Constraints ....36
4.1.1.2.2 Geocentric-UTM Conversion CSU Design....36
4.1.1.2.2.1 Geocentric-UTM Conversion CSU Input/Output Data Elements....36
4.1.1.2.2.2 Geocentric-UTM Conversion CSU Local Data Elements....37
4.1.1.2.2.3 Geocentric-UTM Conversion CSU Global Data Elements....37
4.1.1.2.2.4 Geocentric-UTM Conversion CSU Local and Shared Data Structures ....37
4.1.1.2.2.5 Geocentric-UTM Conversion CSU Interrupts and Signals....37
4.1.1.2.2.6 Geocentric-UTM Conversion CSU Error Handling....37
4.1.1.2.2.7 Geocentric-UTM Conversion CSU Use of Other Elements....37
4.1.1.2.2.8 Geocentric-UTM Conversion CSU Logic Flow....37
4.1.1.2.2.9 Geocentric-UTM Conversion CSU Algorithms....39
4.1.1.2.2.10 Geocentric-UTM Conversion CSU Limitations....39
4.1.1.3 Geocentric-Local Conversion CSU (DL-CSU-1.1.3)....39
4.1.1.3.1 Geocentric-Local Conversion CSU Design Specification/Constraints ....39
4.1.1.3.2 Geocentric-Local Conversion CSU Design....39
4.1.1.3.2.1 Geocentric-Local Conversion CSU Input/Output Data Elements....39
4.1.1.3.2.2 Geocentric-Local Conversion CSU Local Data Elements....40
4.1.1.3.2.3 Geocentric-Local Conversion CSU Global Data Elements....40
4.1.1.3.2.4 Geocentric-Local Conversion CSU Local and Shared Data Structures ....40
4.1.1.3.2.5 Geocentric-Local Conversion CSU Interrupts and Signals....40
4.1.1.3.2.6 Geocentric-Local Conversion CSU Error Handling....40
4.1.1.3.2.7 Geocentric-Local Conversion CSU Use of Other Elements....41
4.1.1.3.2.8 Geocentric-Local Conversion CSU Logic Flow....41
4.1.1.3.2.9 Geocentric-Local Conversion CSU Algorithms....43
4.1.1.3.2.10 Geocentric-Local Conversion CSU Limitations....43
4.1.1.4 Geocentric-Entity Conversion CSU (DL-CSU-1.1.4)....43
4.1.1.4.1 Geocentric-Entity Conversion CSU Design Specification/Constraints ....43
4.1.1.4.2 Geocentric-Entity Conversion CSU Design....43
4.1.1.4.2.1 Geocentric-Entity Conversion CSU Input/Output Data Elements....43
4.1.1.4.2.2 Geocentric-Entity Conversion CSU Local Data Elements....44
4.1.1.4.2.3 Geocentric-Entity Conversion CSU Global Data Elements....44
4.1.1.4.2.4 Geocentric-Entity Conversion CSU Local and Shared Data Structures....44

4.1.1.4.2.5 Geocentric-Entity Conversion CSU Interrupts and Signals.............44
4.1.1.4.2.6 Geocentric-Entity Conversion CSU Error Handling........................45
4.1.1.4.2.7 Geocentric-Entity Conversion CSU Use of Other Elements...........45
4.1.1.4.2.8 Geocentric-Entity Conversion CSU Logic Flow.............................45
4.1.1.4.2.9 Geocentric-Entity Conversion CSU Algorithms.............................47
4.1.1.4.2.10 Geocentric-Entity Conversion CSU Limitations.............................47
4.1.1.5 Geodetic-Geocentric Conversion CSU (DL-CSU-1.1.5).................47
4.1.1.5.1 Geodetic-Geocentric Conversion CSU Design Specification/Constraints...........................................................47
4.1.1.5.2 Geodetic-Geocentric Conversion CSU Design................................47
4.1.1.5.2.1 Geodetic-Geocentric Conversion CSU Input/Output Data Elements ........................................................................................... 47
4.1.1.5.2.2 Geodetic-Geocentric Conversion CSU Local Data Elements..........48
4.1.1.5.2.3 Geodetic-Geocentric Conversion CSU Global Data Elements.........48
4.1.1.5.2.4 Geodetic-Geocentric Conversion CSU Local and Shared Data Structures.......................................................................48
4.1.1.5.2.5 Geodetic-Geocentric Conversion CSU Interrupts and Signals.........48
4.1.1.5.2.6 Geodetic-Geocentric Conversion CSU Error Handling...................48
4.1.1.5.2.7 Geodetic-Geocentric Conversion CSU Use of Other Elements.......48
4.1.1.5.2.8 Geodetic-Geocentric Conversion CSU Logic Flow.........................48
4.1.1.5.2.9 Geodetic-Geocentric Conversion CSU Algorithms..........................50
4.1.1.5.2.10 Geodetic-Geocentric Conversion CSU Limitations..........................50
4.1.1.6 UTM-Geocentric Conversion CSU (DL-CSU-1.1.6).......................50
4.1.1.6.1 UTM-Geocentric Conversion CSU Design Specification/Constraints ........................................................................................... 50
4.1.1.6.2 UTM-Geocentric Conversion CSU Design.....................................50
4.1.1.6.2.1 UTM-Geocentric Conversion CSU Input/Output Data Elements....50
4.1.1.6.2.2 UTM-Geocentric Conversion CSU Local Data Elements................51
4.1.1.6.2.3 UTM-Geocentric Conversion CSU Global Data Elements..............51
4.1.1.6.2.4 UTM-Geocentric Conversion CSU Local and Shared Data Structures ........................................................................................... 51
4.1.1.6.2.5 UTM-Geocentric Conversion CSU Interrupts and Signals..............51
4.1.1.6.2.6 UTM-Geocentric Conversion CSU Error Handling.........................51
4.1.1.6.2.7 UTM-Geocentric Conversion CSU Use of Other Elements.............51
4.1.1.6.2.8 UTM-Geocentric Conversion CSU Logic Flow..............................51
4.1.1.6.2.9 UTM-Geocentric Conversion CSU Algorithms..............................53
4.1.1.6.2.10 UTM-Geocentric Conversion CSU Limitations..............................53
4.1.1.7 Local-Geocentric Conversion CSU (DL-CSU-1.1.7).......................53
4.1.1.7.1 Local-Geocentric Conversion CSU Design Specification/Constraints ........................................................................................... 53
4.1.1.7.2 Local-Geocentric Conversion CSU Design.....................................53
4.1.1.7.2.1 Local-Geocentric Conversion CSU Input/Output Data Elements....53
4.1.1.7.2.2 Local-Geocentric Conversion CSU Local Data Elements................54
4.1.1.7.2.3 Local-Geocentric Conversion CSU Global Data Elements..............54
4.1.1.7.2.4 Local-Geocentric Conversion CSU Local and Shared Data Structures ........................................................................................... 54

4.1.1.7.2.5 Local-Geocentric Conversion CSU Interrupts and Signals..............54
4.1.1.7.2.6 Local-Geocentric Conversion CSU Error Handling.........................54
4.1.1.7.2.7 Local-Geocentric Conversion CSU Use of Other Elements.............55
4.1.1.7.2.8 Local-Geocentric Conversion CSU Logic Flow..............................55
4.1.1.7.2.9 Local-Geocentric Conversion CSU Algorithms..............................57
4.1.1.7.2.10 Local-Geocentric Conversion CSU Limitations..............................57
4.1.1.8 Entity-Geocentric Conversion CSU (DL-CSU-1.1.8)......................57
4.1.1.8.1 Entity-Geocentric Conversion CSU Design Specification/Constraints ..........................................................................................57
4.1.1.8.2 Entity-Geocentric Conversion CSU Design....................................57
4.1.1.8.2.1 Entity-Geocentric Conversion CSU Input/Output Data Elements....57
4.1.1.8.2.2 Entity-Geocentric Conversion CSU Local Data Elements...............58
4.1.1.8.2.3 Entity-Geocentric Conversion CSU Global Data Elements.............58
4.1.1.8.2.4 Entity-Geocentric Conversion CSU Local and Shared Data Structures.........................................................................58
4.1.1.8.2.5 Entity-Geocentric Conversion CSU Interrupts and Signals.............58
4.1.1.8.2.6 Entity-Geocentric Conversion CSU Error Handling........................59
4.1.1.8.2.7 Entity-Geocentric Conversion CSU Use of Other Elements............59
4.1.1.8.2.8 Entity-Geocentric Conversion CSU Logic Flow.............................59
4.1.1.8.2.9 Entity-Geocentric Conversion CSU Algorithms.............................61
4.1.1.8.2.10 Entity-Geocentric Conversion CSU Limitations.............................61
4.1.2 Orientation System Conversion CSC (DL-CSC-1.2).......................61
4.1.2.1 Euler-Orientation Conversion CSU (DL-CSU-1.1.1)......................63
4.1.2.1.1 Euler-Orientation Conversion CSU Design Specification/Constraints ..........................................................................................63
4.1.2.1.2 Euler-Orientation Conversion CSU Design....................................63
4.1.2.1.2.1 Euler-Orientation Conversion CSU Input/Output Data Elements....63
4.1.2.1.2.2 Euler-Orientation Conversion CSU Local Data Elements...............64
4.1.2.1.2.3 Euler-Orientation Conversion CSU Global Data Elements.............64
4.1.2.1.2.4 Euler-Orientation Conversion CSU Local and Shared Data Structures.........................................................................64
4.1.2.1.2.5 Euler-Orientation Conversion CSU Interrupts and Signals.............64
4.1.2.1.2.6 Euler-Orientation Conversion CSU Error Handling........................64
4.1.2.1.2.7 Euler-Orientation Conversion CSU Use of Other Elements............64
4.1.2.1.2.8 Euler-Orientation Conversion CSU Logic Flow.............................64
4.1.2.1.2.9 Euler-Orientation Conversion CSU Algorithms.............................66
4.1.2.1.2.10 Euler-Orientation Conversion CSU Limitations.............................66
4.1.2.2 Orientation-Euler Conversion CSU (DL-CSU-1.1.2)......................66
4.1.2.2.1 Orientation-Euler Conversion CSU Design Specification/Constraints ..........................................................................................66
4.1.2.2.2 Orientation-Euler Conversion CSU Design....................................66
4.1.2.2.2.1 Orientation-Euler Conversion CSU Input/Output Data Elements....66
4.1.2.2.2.2 Orientation-Euler Conversion CSU Local Data Elements...............67
4.1.2.2.2.3 Orientation-Euler Conversion CSU Global Data Elements.............67
4.1.2.2.2.4 Orientation-Euler Conversion CSU Local and Shared Data Structures.........................................................................67

| | | |
|---|---|---|
| 4.1.2.2.2.5 | Orientation-Euler Conversion CSU Interrupts and Signals | 67 |
| 4.1.2.2.2.6 | Orientation-Euler Conversion CSU Error Handling | 67 |
| 4.1.2.2.2.7 | Orientation-Euler Conversion CSU Use of Other Elements | 68 |
| 4.1.2.2.2.8 | Orientation-Euler Conversion CSU Logic Flow | 68 |
| 4.1.2.2.2.9 | Orientation-Euler Conversion CSU Algorithms | 70 |
| 4.1.2.2.2.10 | Orientation-Euler Conversion CSU Limitations | 70 |
| 4.1 | Dead-Reckoning Algorithms CSC (DL-CSC-2) | 70 |
| 4.2.1 | Update Dead Reckoned Position CSU (DL-CSU-2.1) | 74 |
| 4.2.1.1 | Update Dead Reckoned Position CSU Design Specification/Constraints | 74 |
| 4.2.1.2 | Update Dead Reckoned Position CSU Design | 74 |
| 4.2.1.2.1 | Update Dead Reckoned Position CSU Input/Output Data Elements | 74 |
| 4.2.1.2.2 | Update Dead Reckoned Position CSU Local Data Elements | 74 |
| 4.2.1.2.3 | Update Dead Reckoned Position CSU Global Data Elements | 75 |
| 4.2.1.2.4 | Update Dead Reckoned Position CSU Local and Shared Data Structures | 75 |
| 4.2.1.2.5 | Update Dead Reckoned Position CSU Interrupts and Signals | 75 |
| 4.2.1.2.6 | Update Dead Reckoned Position CSU Error Handling | 75 |
| 4.2.1.2.7 | Update Dead Reckoned Position CSU Use of Other Elements | 75 |
| 4.2.1.2.8 | Update Dead Reckoned Position CSU Logic Flow | 75 |
| 4.2.1.2.9 | Update Dead Reckoned Position CSU Algorithms | 77 |
| 4.2.1.2.10 | Update Dead Reckoned Position CSU Limitations | 77 |
| 4.2.2 | DRM Static CSU (DL-CSU-2.2) | 77 |
| 4.2.2.1 | DRM Static CSU Design Specification/Constraints | 77 |
| 4.2.2.2 | DRM Static CSU Design | 77 |
| 4.2.3 | DRM FPW CSU (DL-CSU-2.3) | 77 |
| 4.2.3.1 | DRM FPW CSU Design Specification/Constraints | 77 |
| 4.2.3.2 | DRM FPW CSU Design | 77 |
| 4.2.3.2.1 | DRM FPW CSU Input/Output Data Elements | 78 |
| 4.2.3.2.2 | DRM FPW CSU Local Data Elements | 78 |
| 4.2.3.2.3 | DRM FPW CSU Global Data Elements | 78 |
| 4.2.3.2.4 | DRM FPW CSU Local and Shared Data Structures | 78 |
| 4.2.3.2.5 | DRM FPW CSU Interrupts and Signals | 78 |
| 4.2.3.2.6 | DRM FPW CSU Error Handling | 78 |
| 4.2.3.2.7 | DRM FPW CSU Use of Other Elements | 79 |
| 4.2.3.2.8 | DRM FPW CSU Logic Flow | 79 |
| 4.2.3.2.9 | DRM FPW CSU Algorithms | 81 |
| 4.2.3.2.10 | DRM FPW CSU Limitations | 81 |
| 4.2.4 | DRM FVW CSU (DL-CSU-2.4) | 81 |
| 4.2.4.1 | DRM FVW CSU Design Specification/Constraints | 81 |
| 4.2.4.2 | DRM FVW CSU Design | 81 |
| 4.2.4.2.1 | DRM FVW CSU Input/Output Data Elements | 81 |
| 4.2.4.2.2 | DRM FVW CSU Local Data Elements | 82 |
| 4.2.4.2.3 | DRM FVW CSU Global Data Elements | 82 |
| 4.2.4.2.4 | DRM FVW CSU Local and Shared Data Structures | 82 |
| 4.2.4.2.5 | DRM FVW CSU Interrupts and Signals | 82 |

4.2.4.2.6 DRM FVW CSU Error Handling....83
4.2.4.2.7 DRM FVW CSU Use of Other Elements....83
4.2.4.2.8 DRM FVW CSU Logic Flow....83
4.2.4.2.9 DRM FVW CSU Algorithms....85
4.2.4.2.10 DRM FVW CSU Limitations....85
4.2.5 DRM RPW CSU (DL-CSU-2.5)....85
4.2.5.1 DRM RPW CSU Design Specification/Constraints....85
4.2.5.2 DRM RPW CSU Design....85
4.2.5.2.1 DRM RPW CSU Input/Output Data Elements....85
4.2.5.2.2 DRM RPW CSU Local Data Elements....86
4.2.5.2.3 DRM RPW CSU Global Data Elements....86
4.2.5.2.4 DRM RPW CSU Local and Shared Data Structures....86
4.2.5.2.5 DRM RPW CSU Interrupts and Signals....86
4.2.5.2.6 DRM RPW CSU Error Handling....86
4.2.5.2.7 DRM RPW CSU Use of Other Elements....87
4.2.5.2.8 DRM RPW CSU Logic Flow....87
4.2.5.2.9 DRM RPW CSU Algorithms....89
4.2.5.2.10 DRM RPW CSU Limitations....89
4.2.6 DRM RVW CSU (DL-CSU-2.6)....89
4.2.6.1 DRM RVW CSU Design Specification/Constraints....89
4.2.6.2 DRM RVW CSU Design....90
4.2.6.2.1 DRM RVW CSU Input/Output Data Elements....90
4.2.6.2.2 DRM RVW CSU Local Data Elements....90
4.2.6.2.3 DRM RVW CSU Global Data Elements....91
4.2.6.2.4 DRM RVW CSU Local and Shared Data Structures....91
4.2.6.2.5 DRM RVW CSU Interrupts and Signals....91
4.2.6.2.6 DRM RVW CSU Error Handling....91
4.2.6.2.7 DRM RVW CSU Use of Other Elements....91
4.2.6.2.8 DRM RVW CSU Logic Flow....91
4.2.6.2.9 DRM RVW CSU Algorithms....93
4.2.6.2.10 DRM RVW CSU Limitations....93
4.3 Calculation Algorithms CSC (DL-CSC-3)....93
4.3.1 Calculate Distance CSU (DL-CSU-3.1)....95
4.3.1.1 Calculate Distance CSU Design Specification/Constraints....95
4.3.1.2 Calculate Distance CSU Design....96
4.3.1.2.1 Calculate Distance CSU Input/Output Data Elements....96
4.3.1.2.2 Calculate Distance CSU Local Data Elements....96
4.3.1.2.3 Calculate Distance CSU Global Data Elements....97
4.3.1.2.4 Calculate Distance CSU Local and Shared Data Structures....97
4.3.1.2.5 Calculate Distance CSU Interrupts and Signals....97
4.3.1.2.6 Calculate Distance CSU Error Handling....97
4.3.1.2.7 Calculate Distance CSU Use of Other Elements....97
4.3.1.2.8 Calculate Distance CSU Logic Flow....97
4.3.1.2.9 Calculate Distance CSU Algorithms....98
4.3.1.2.10 Calculate Distance CSU Limitations....99
4.3.2 Calculate Azimuth CSU (DL-CSU-3.2)....99

4.3.2.1 Calculate Azimuth CSU Design Specification/Constraints..............99
4.3.2.2 Calculate Azimuth CSU Design......................................................99
4.3.2.2.1 Calculate Azimuth CSU Input/Output Data Elements......................99
4.3.2.2.2 Calculate Azimuth CSU Local Data Elements..............................100
4.3.2.2.3 Calculate Azimuth CSU Global Data Elements............................100
4.3.2.2.4 Calculate Azimuth CSU Local and Shared Data Structures...........100
4.3.2.2.5 Calculate Azimuth CSU Interrupts and Signals.............................101
4.3.2.2.6 Calculate Azimuth CSU Error Handling.......................................101
4.3.2.2.7 Calculate Azimuth CSU Use of Other Elements...........................101
4.3.2.2.8 Calculate Azimuth CSU Logic Flow............................................101
4.3.2.2.9 Calculate Azimuth CSU Algorithms............................................103
4.3.2.2.10 Calculate Azimuth CSU Limitations............................................104
4.3.3 Calculate Velocity CSU (DL-CSU-3.3).......................................104
4.3.3.1 Calculate Velocity CSU Design Specification/Constraints...........104
4.3.3.2 Calculate Velocity CSU Design...................................................105
4.3.3.2.1 Calculate Velocity CSU Input/Output Data Elements...................105
4.3.3.2.2 Calculate Velocity CSU Local Data Elements..............................105
4.3.3.2.3 Calculate Velocity CSU Global Data Elements............................105
4.3.3.2.4 Calculate Velocity CSU Local and Shared Data Structures...........105
4.3.3.2.5 Calculate Velocity CSU Interrupts and Signals.............................105
4.3.3.2.6 Calculate Velocity CSU Error Handling.......................................106
4.3.3.2.7 Calculate Velocity CSU Use of Other Elements...........................106
4.3.3.2.8 Calculate Velocity CSU Logic Flow............................................106
4.3.3.2.9 Calculate Velocity CSU Algorithms............................................108
4.3.3.2.10 Calculate Velocity CSU Limitations............................................108
4.3.4 Calculate Elevation CSU (DL-CSU-3.4)......................................108
4.3.4.1 Calculate Elevation CSU Design Specification/Constraints..........108
4.3.4.2 Calculate Elevation CSU Design..................................................109
4.3.4.2.1 Calculate Elevation CSU Input/Output Data Elements..................109
4.3.4.2.2 Calculate Elevation CSU Local Data Elements.............................109
4.3.4.2.3 Calculate Elevation CSU Global Data Elements...........................110
4.3.4.2.4 Calculate Elevation CSU Local and Shared Data Structures..........110
4.3.4.2.5 Calculate Elevation CSU Interrupts and Signals...........................110
4.3.4.2.6 Calculate Elevation CSU Error Handling.....................................110
4.3.4.2.7 Calculate Elevation CSU Use of Other Elements..........................110
4.3.4.2.8 Calculate Elevation CSU Logic Flow...........................................110
4.3.4.2.9 Calculate Elevation CSU Algorithms...........................................112
4.3.4.2.10 Calculate Elevation CSU Limitations...........................................113
4.3.5 Calculate Azimuth and Elevation CSU (DL-CSU-3.5).................113
4.3.5.1 Calculate Azimuth and Elevation CSU Design Specification/Constraints..........................................................113
4.3.5.2 Calculate Azimuth and Elevation CSU Design.............................114
4.3.5.2.1 Calculate Azimuth and Elevation CSU Input/Output Data Elements ..........................................................................................114
4.3.5.2.2 Calculate Azimuth and Elevation CSU Local Data Elements........115
4.3.5.2.3 Calculate Azimuth and Elevation CSU Global Data Elements......115

4.3.5.2.4 Calculate Azimuth and Elevation CSU Local and Shared Data Structures....115
4.3.5.2.5 Calculate Azimuth and Elevation CSU Interrupts and Signals......115
4.3.5.2.6 Calculate Azimuth and Elevation CSU Error Handling................115
4.3.5.2.7 Calculate Azimuth and Elevation CSU Use of Other Elements.....115
4.3.5.2.8 Calculate Azimuth and Elevation CSU Logic Flow......................116
4.3.5.2.9 Calculate Azimuth and Elevation CSU Algorithms......................117
4.3.5.2.10 Calculate Azimuth and Elevation CSU Limitations......................117
4.4 Filtering Algorithms CSC (DL-CSC-4)........................................117
4.4.1 Filtering Routines CSC (DL-CSC-4.1)..........................................117
4.4.1.1 Filter Distance CSU......................................................................120
4.4.1.1.1 Filter Distance CSU Design Specification/Constraints.................120
4.4.1.1.2 Filter Distance CSU Design..........................................................120
4.4.1.1.2.1 Filter Distance CSU Input/Output Data Elements.........................120
4.3.2.1.2.2 Filter Distance CSU Local Data Elements....................................121
4.4.1.1.2.3 Filter Distance CSU Global Data Elements..................................121
4.4.1.1.2.4 Filter Distance CSU Local and Shared Data Structures................121
4.4.1.1.2.5 Filter Distance CSU Interrupts and Signals..................................121
4.4.1.1.2.6 Filter Distance CSU Error Handling.............................................121
4.4.1.1.2.7 Filter Distance CSU Use of Other Elements.................................122
4.4.1.1.2.8 Filter Distance CSU Logic Flow...................................................122
4.4.1.1.2.9 Filter Distance CSU Algorithms...................................................124
4.4.1.1.2.10 Filter Distance CSU Limitations...................................................124
4.4.1.2 Filter Azimuth CSU.......................................................................125
4.4.1.2.1 Filter Azimuth CSU Design Specification/Constraints.................125
4.4.1.2.2 Filter Azimuth CSU Design..........................................................125
4.4.1.2.2.1 Filter Azimuth CSU Input/Output Data Elements.........................125
4.4.1.2.2.2 Filter Azimuth CSU Local Data Elements....................................126
4.4.1.2.2.3 Filter Azimuth CSU Global Data Elements..................................126
4.4.1.2.2.4 Filter Azimuth CSU Local and Shared Data Structures................126
4.4.1.2.2.5 Filter Azimuth CSU Interrupts and Signals..................................126
4.4.1.2.2.6 Filter Azimuth CSU Error Handling.............................................127
4.4.1.2.2.7 Filter Azimuth CSU Use of Other Elements.................................127
4.4.1.2.2.8 Filter Azimuth CSU Logic Flow...................................................127
4.4.1.2.2.9 Filter Azimuth CSU Algorithms...................................................130
4.4.1.2.2.10 Filter Azimuth CSU Limitations...................................................130
4.4.1.3 Filter Minimum Velocity CSU.....................................................130
4.4.1.3.1 Filter Minimum Velocity CSU Design Specification/Constraints..130
4.4.1.3.2 Filter Minimum Velocity CSU Design.........................................131
4.4.1.3.2.1 Filter Minimum Velocity CSU Input/Output Data Elements.........131
4.4.1.3.2.2 Filter Minimum Velocity CSU Local Data Elements....................131
4.4.1.3.2.3 Filter Minimum Velocity CSU Global Data Elements..................132
4.4.1.3.2.4 Filter Minimum Velocity CSU Local and Shared Data Structures.132
4.4.1.3.2.5 Filter Minimum Velocity CSU Interrupts and Signals..................132
4.4.1.3.2.6 Filter Minimum Velocity CSU Error Handling.............................132
4.4.1.3.2.7 Filter Minimum Velocity CSU Use of Other Elements.................132

4.4.1.3.2.8 Filter Minimum Velocity CSU Logic Flow....132
4.4.1.3.2.9 Filter Minimum Velocity CSU Algorithms....134
4.4.1.3.2.10 Filter Minimum Velocity CSU Limitations....134
4.4.1.4 Filter Maximum Velocity CSU....134
4.4.1.4.1 Filter Maximum Velocity CSU Design Specification/Constraints. 135
4.4.1.4.2 Filter Maximum Velocity CSU Design....135
4.4.1.4.2.1 Filter Maximum Velocity CSU Input/Output Data Elements....135
4.4.1.4.2.2 Filter Maximum Velocity CSU Local Data Elements....136
4.4.1.4.2.3 Filter Maximum Velocity CSU Global Data Elements....136
4.4.1.4.2.4 Filter Maximum Velocity CSU Local and Shared Data Structures 136
4.4.1.4.2.5 Filter Maximum Velocity CSU Interrupts and Signals....136
4.4.1.4.2.6 Filter Maximum Velocity CSU Error Handling....136
4.4.1.4.2.7 Filter Maximum Velocity CSU Use of Other Elements....137
4.4.1.4.2.8 Filter Maximum Velocity CSU Logic Flow....137
4.4.1.4.2.9 Filter Maximum Velocity CSU Algorithms....139
4.4.1.4.2.10 Filter Maximum Velocity CSU Limitations....139
4.4.1.5 Filter Elevation CSU....140
4.4.1.5.1 Filter Elevation CSU Design Specification/Constraints....140
4.4.1.5.2 Filter Elevation CSU Design....140
4.4.1.5.2.1 Filter Elevation CSU Input/Output Data Elements....140
4.4.1.5.2.2 Filter Elevation CSU Local Data Elements....141
4.4.1.5.2.3 Filter Elevation CSU Global Data Elements....141
4.4.1.5.2.4 Filter Elevation CSU Local and Shared Data Structures....141
4.4.1.5.2.5 Filter Elevation CSU Interrupts and Signals....141
4.4.1.5.2.6 Filter Elevation CSU Error Handling....142
4.4.1.5.2.7 Filter Elevation CSU Use of Other Elements....142
4.4.1.5.2.8 Filter Elevation CSU Logic Flow....142
4.4.1.5.2.9 Filter Elevation CSU Algorithms....145
4.4.1.5.2.10 Filter Elevation CSU Limitations....145
4.4.1.6 Filter Azimuth and Elevation CSU....145
4.4.1.6.1 Filter Azimuth and Elevation CSU Design Specification/Constraints ....145
4.4.1.6.2 Filter Azimuth and Elevation CSU Design....146
4.4.1.6.2.1 Filter Azimuth and Elevation CSU Input/Output Data Elements...146
4.4.1.6.2.2 Filter Azimuth and Elevation CSU Local Data Elements....147
4.4.1.6.2.3 Filter Azimuth and Elevation CSU Global Data Elements....147
4.4.1.6.2.4 Filter Azimuth and Elevation CSU Local and Shared Data Structures ....148
4.4.1.6.2.5 Filter Azimuth and Elevation CSU Interrupts and Signals....148
4.4.1.6.2.6 Filter Azimuth and Elevation CSU Error Handling....148
4.4.1.6.2.7 Filter Azimuth and Elevation CSU Use of Other Elements....148
4.4.1.6.2.8 Filter Azimuth and Elevation CSU Logic Flow....148
4.4.1.6.2.9 Filter Azimuth and Elevation CSU Algorithms....151
4.4.1.6.2.10 Filter Azimuth and Elevation CSU Limitations....151
4.4.2 Filter List By Distance CSC (DL-CSC-4.2)....151
4.4.2.1 Filter Entity State Distance CSU....154

4.4.2.1.1 Filter Entity State Distance CSU Design Specification/Constraints ....154
4.4.2.1.2 Filter Entity State Distance CSU Design....154
4.4.2.1.2.1 Filter Entity State Distance CSU Input/Output Data Elements....154
4.4.2.1.2.2 Filter Entity State Distance CSU Local Data Elements....155
4.4.2.1.2.3 Filter Entity State Distance CSU Global Data Elements....155
4.4.2.1.2.4 Filter Entity State Distance CSU Local and Shared Data Structures ....155
4.4.2.1.2.5 Filter Entity State Distance CSU Interrupts and Signals....156
4.4.2.1.2.6 Filter Entity State Distance CSU Error Handling....156
4.4.2.1.2.7 Filter Entity State Distance CSU Use of Other Elements....156
4.4.2.1.2.8 Filter Entity State Distance CSU Logic Flow....156
4.4.2.1.2.9 Filter Entity State Distance CSU Algorithms....158
4.4.2.1.2.10 Filter Entity State Distance CSU Limitations....158
4.4.2.2 Filter Fire Distance CSU....158
4.4.2.2.1 Filter Fire Distance CSU Design Specification/Constraints....158
4.4.2.2.2 Filter Fire Distance CSU Design....158
4.4.2.2.2.1 Filter Fire Distance CSU Input/Output Data Elements....158
4.4.2.2.2.2 Filter Fire Distance CSU Local Data Elements....159
4.4.2.2.2.3 Filter Fire Distance CSU Global Data Elements....159
4.4.2.2.2.4 Filter Fire Distance CSU Local and Shared Data Structures....159
4.4.2.2.2.5 Filter Fire Distance CSU Interrupts and Signals....159
4.4.2.2.2.6 Filter Fire Distance CSU Error Handling....160
4.4.2.2.2.7 Filter Fire Distance CSU Use of Other Elements....160
4.4.2.2.2.8 Filter Fire Distance CSU Logic Flow....160
4.4.2.2.2.9 Filter Fire Distance CSU Algorithms....161
4.4.2.2.2.10 Filter Fire Distance CSU Limitations....162
4.4.2.3 Filter Detonation Distance CSU....162
4.4.2.3.1 Filter Detonation Distance CSU Design Specification/Constraints162
4.4.2.3.2 Filter Detonation Distance CSU Design....162
4.4.2.3.2.1 Filter Detonation Distance CSU Input/Output Data Elements....162
4.4.2.3.2.2 Filter Detonation Distance CSU Local Data Elements....163
4.4.2.3.2.3 Filter Detonation Distance CSU Global Data Elements....163
4.4.2.3.2.4 Filter Detonation Distance CSU Local and Shared Data Structures ....163
4.4.2.3.2.5 Filter Detonation Distance CSU Interrupts and Signals....164
4.4.2.3.2.6 Filter Detonation Distance CSU Error Handling....164
4.4.2.3.2.7 Filter Detonation Distance CSU Use of Other Elements....164
4.4.2.3.2.8 Filter Detonation Distance CSU Logic Flow....164
4.4.2.3.2.9 Filter Detonation Distance CSU Algorithms....166
4.4.2.3.2.10 Filter Detonation Distance CSU Limitations....166
4.4.2.4 Filter Laser Distance CSU....166
4.4.2.4.1 Filter Laser Distance CSU Design Specification/Constraints....166
4.4.2.4.2 Filter Laser Distance CSU Design....166
4.4.2.4.2.1 Filter Laser Distance CSU Input/Output Data Elements....166
4.4.2.4.2.2 Filter Laser Distance CSU Local Data Elements....167

4.4.2.4.2.3 Filter Laser Distance CSU Global Data Elements..........................167
4.4.2.4.2.4 Filter Laser Distance CSU Local and Shared Data Structures.......167
4.4.2.4.2.5 Filter Laser Distance CSU Interrupts and Signals..........................167
4.4.2.4.2.6 Filter Laser Distance CSU Error Handling....................................168
4.4.2.4.2.7 Filter Laser Distance CSU Use of Other Elements.........................168
4.4.2.4.2.8 Filter Laser Distance CSU Logic Flow..........................................168
4.4.2.4.2.9 Filter Laser Distance CSU Algorithms..........................................170
4.4.2.4.2.10 Filter Laser Distance CSU Limitations..........................................170
4.4.2.5 Filter Transmitter Distance CSU..................................................170
4.4.2.5.1 Filter Transmitter Distance CSU Design Specification/Constraints ..................................................................................................170
4.4.2.5.2 Filter Transmitter Distance CSU Design.......................................170
4.4.2.5.2.1 Filter Transmitter Distance CSU Input/Output Data Elements......170
4.4.2.5.2.2 Filter Transmitter Distance CSU Local Data Elements..................171
4.4.2.5.2.3 Filter Transmitter Distance CSU Global Data Elements................171
4.4.2.5.2.4 Filter Transmitter Distance CSU Local and Shared Data Structures ..................................................................................................171
4.4.2.5.2.5 Filter Transmitter Distance CSU Interrupts and Signals................172
4.4.2.5.2.6 Filter Transmitter Distance CSU Error Handling..........................172
4.4.2.5.2.7 Filter Transmitter Distance CSU Use of Other Elements..............172
4.4.2.5.2.8 Filter Transmitter Distance CSU Logic Flow................................172
4.4.2.5.2.10 Filter Transmitter Distance CSU Limitations................................174
4.4.3 Filter List By Orientation CSC (DL-CSC-4.3)..............................174
4.4.3.1 Filter Entity State Azimuth CSU..................................................177
4.4.3.1.1 Filter Entity State Azimuth CSU Design Specification/Constraints ..................................................................................................177
4.4.3.1.2 Filter Entity State Azimuth CSU Design.......................................177
4.4.3.1.2.1 Filter Entity State Azimuth CSU Input/Output Data Elements......177
4.4.3.1.2.2 Filter Entity State Azimuth CSU Local Data Elements..................178
4.4.3.1.2.3 Filter Entity State Azimuth CSU Global Data Elements................178
4.4.3.1.2.4 Filter Entity State Azimuth CSU Local and Shared Data Structures ..................................................................................................178
4.4.3.1.2.5 Filter Entity State Azimuth CSU Interrupts and Signals................178
4.4.3.1.2.6 Filter Entity State Azimuth CSU Error Handling..........................178
4.4.3.1.2.7 Filter Entity State Azimuth CSU Use of Other Elements..............179
4.4.3.1.2.8 Filter Entity State Azimuth CSU Logic Flow................................179
4.4.3.1.2.9 Filter Entity State Azimuth CSU Algorithms................................181
4.4.3.1.2.10 Filter Entity State Azimuth CSU Limitations................................181
4.4.3.2 Filter Fire Azimuth CSU.............................................................181
4.4.3.2.1 Filter Fire Azimuth Design Specification/Constraints...................181
4.4.3.2.2 Filter Fire Azimuth Design..........................................................181
4.4.3.2.2.1 Filter Fire Azimuth Input/Output Data Elements..........................181
4.4.3.2.2.2 Filter Fire Azimuth Local Data Elements......................................182
4.4.3.2.2.3 Filter Fire Azimuth Global Data Elements....................................182
4.4.3.2.2.4 Filter Fire Azimuth Local and Shared Data Structures..................182
4.4.3.2.2.5 Filter Fire Azimuth Interrupts and Signals....................................182

4.4.3.2.2.6 Filter Fire Azimuth Error Handling....182
4.4.3.2.2.7 Filter Fire Azimuth Use of Other Elements....183
4.4.3.2.2.8 Filter Fire Azimuth Logic Flow....183
4.4.3.2.2.9 Filter Fire Azimuth Algorithms....185
4.4.3.2.2.10 Filter Fire Azimuth Limitations....185
4.4.3.3 Filter Detonation Azimuth CSU....185
4.4.3.3.1 Filter Detonation Azimuth Design Specification/Constraints....185
4.4.3.3.2 Filter Detonation Azimuth Design....185
4.4.3.3.2.1 Filter Detonation Azimuth Input/Output Data Elements....185
4.4.3.3.2.2 Filter Detonation Azimuth Local Data Elements....186
4.4.3.3.2.3 Filter Detonation Azimuth Global Data Elements....186
4.4.3.3.2.4 Filter Detonation Azimuth Local and Shared Data Structures....186
4.4.3.3.2.5 Filter Detonation Azimuth Interrupts and Signals....186
4.4.3.3.2.6 Filter Detonation Azimuth Error Handling....186
4.4.3.3.2.7 Filter Detonation Azimuth Use of Other Elements....187
4.4.3.3.2.8 Filter Detonation Azimuth Logic Flow....187
4.4.3.3.2.9 Filter Detonation Azimuth Algorithms....189
4.4.3.3.2.10 Filter Detonation Azimuth Limitations....189
4.4.3.4 Filter Laser Azimuth CSU....189
4.4.3.4.1 Filter Laser Azimuth Design Specification/Constraints....189
4.4.3.4.2 Filter Laser Azimuth Design....189
4.4.3.4.2.1 Filter Laser Azimuth Input/Output Data Elements....189
4.4.3.4.2.2 Filter Laser Azimuth Local Data Elements....190
4.4.3.4.2.3 Filter Laser Azimuth Global Data Elements....190
4.4.3.4.2.4 Filter Laser Azimuth Local and Shared Data Structures....190
4.4.3.4.2.5 Filter Laser Azimuth Interrupts and Signals....190
4.4.3.4.2.6 Filter Laser Azimuth Error Handling....190
4.4.3.4.2.7 Filter Laser Azimuth Use of Other Elements....191
4.4.3.4.2.8 Filter Laser Azimuth Logic Flow....191
4.4.3.4.2.9 Filter Laser Azimuth Algorithms....193
4.4.3.4.2.10 Filter Laser Azimuth Limitations....193
4.4.3.5 Filter Transmitter Azimuth CSU....193
4.4.3.5.1 Filter Transmitter Azimuth Design Specification/Constraints....193
4.4.3.5.2 Filter Transmitter Azimuth Design....193
4.4.3.5.2.1 Filter Transmitter Azimuth Input/Output Data Elements....193
4.4.3.5.2.2 Filter Transmitter Azimuth Local Data Elements....194
4.4.3.5.2.3 Filter Transmitter Azimuth Global Data Elements....194
4.4.3.5.2.4 Filter Transmitter Azimuth Local and Shared Data Structures....194
4.4.3.5.2.5 Filter Transmitter Azimuth Interrupts and Signals....194
4.4.3.5.2.6 Filter Transmitter Azimuth Error Handling....194
4.4.3.5.2.7 Filter Transmitter Azimuth Use of Other Elements....195
4.4.3.5.2.8 Filter Transmitter Azimuth Logic Flow....195
4.4.3.5.2.9 Filter Transmitter Azimuth Algorithms....197
4.4.3.5.2.10 Filter Transmitter Azimuth Limitations....197
4.4.3.6 Filter Entity State Elevation CSU....197
4.4.3.6.1 Filter Entity State Elevation CSU Design Specification/Constraints

..................................................................................197
4.4.3.6.2 Filter Entity State Elevation CSU Design......................................197
4.4.3.6.2.1 Filter Entity State Elevation CSU Input/Output Data Elements.....197
4.4.3.6.2.2 Filter Entity State Elevation CSU Local Data Elements................198
4.4.3.6.2.3 Filter Entity State Elevation CSU Global Data Elements...............198
4.4.3.6.2.4 Filter Entity State Elevation CSU Local and Shared Data Structures ..................................................................................198
4.4.3.6.2.5 Filter Entity State Elevation CSU Interrupts and Signals...............198
4.4.3.6.2.6 Filter Entity State Elevation CSU Error Handling..........................198
4.4.3.6.2.7 Filter Entity State Elevation CSU Use of Other Elements.............199
4.4.3.6.2.8 Filter Entity State Elevation CSU Logic Flow...............................199
4.4.3.6.2.9 Filter Entity State Elevation CSU Algorithms................................200
4.4.3.6.2.10 Filter Entity State Elevation CSU Limitations................................201
4.4.3.7 Filter Entity State Azimuth and Elevation CSU.............................201
4.4.3.7.1 Filter Entity State Azimuth and Elevation CSU Design Specification/Constraints..........................................................201
4.4.3.7.2 Filter Entity State Azimuth and Elevation CSU Design.................201
4.4.3.7.2.1 Filter Entity State Azimuth and Elevation CSU Input/Output Data Elements..........................................................................201
4.4.3.7.2.2 Filter Entity State Azimuth and Elevation CSU Local Data Elements ..................................................................................202
4.4.3.7.2.3 Filter Entity State Azimuth and Elevation CSU Global Data Elements..........................................................................202
4.4.3.7.2.4 Filter Entity State Azimuth and Elevation CSU Local and Shared Data Structures..........................................................................202
4.4.3.7.2.5 Filter Entity State Azimuth and Elevation CSU Interrupts and Signals..........................................................................203
4.4.3.7.2.6 Filter Entity State Azimuth and Elevation CSU Error Handling....203
4.4.3.7.2.7 Filter Entity State Azimuth and Elevation CSU Use of Other Elements..........................................................................203
4.4.3.7.2.8 Filter Entity State Azimuth and Elevation CSU Logic Flow..........203
4.4.3.7.2.9 Filter Entity State Azimuth and Elevation CSU Algorithms..........204
4.4.3.7.2.10 Filter Entity State Azimuth and Elevation CSU Limitations..........205
4.4.4 Filter List By Velocity CSC (DL-CSC-4.4)..................................205
4.4.4.1 Filter Entity State Minimum Velocity CSU..................................206
4.4.4.1.1 Filter Entity State Minimum Velocity CSU Design Specification/Constraints..........................................................206
4.4.4.1.2 Filter Entity State Minimum Velocity CSU Design.......................207
4.4.4.1.2.1 Filter Entity State Minimum Velocity CSU Input/Output Data Elements..........................................................................207
4.4.4.1.2.2 Filter Entity State Minimum Velocity CSU Local Data Elements. 207
4.4.4.1.2.3 Filter Entity State Minimum Velocity CSU Global Data Elements ..................................................................................207
4.4.4.1.2.4 Filter Entity State Minimum Velocity CSU Local and Shared Data Structures..........................................................................207
4.4.4.1.2.5 Filter Entity State Minimum Velocity CSU Interrupts and Signals

..................................................................................... 208
4.4.4.1.2.6 Filter Entity State Minimum Velocity CSU Error Handling..........208
4.4.4.1.2.7 Filter Entity State Minimum Velocity CSU Use of Other Elements
..................................................................................... 208
4.4.4.1.2.8 Filter Entity State Minimum Velocity CSU Logic Flow................208
4.4.4.1.2.9 Filter Entity State Minimum Velocity CSU Algorithms................210
4.4.4.1.2.10 Filter Entity State Minimum Velocity CSU Limitations................210
4.4.4.2 Filter Entity State Maximum Velocity CSU..................................210
4.4.4.2.1 Filter Entity State Maximum Velocity CSU Design Specification/Constraints..........................................................210
4.4.4.2.2 Filter Entity State Maximum Velocity CSU Design......................210
4.4.4.2.2.1 Filter Entity State Maximum Velocity CSU Input/Output Data Elements....................................................................................210
4.4.4.2.2.2 Filter Entity State Maximum Velocity CSU Local Data Elements 211
4.4.4.2.2.3 Filter Entity State Maximum Velocity CSU Global Data Elements
..................................................................................... 211
4.4.4.2.2.4 Filter Entity State Maximum Velocity CSU Local and Shared Data Structures....................................................................................211
4.4.4.2.2.5 Filter Entity State Maximum Velocity CSU Interrupts and Signals
..................................................................................... 211
4.4.4.2.2.6 Filter Entity State Maximum Velocity CSU Error Handling..........211
4.4.4.2.2.7 Filter Entity State Maximum Velocity CSU Use of Other Elements
..................................................................................... 212
4.4.4.2.2.8 Filter Entity State Maximum Velocity CSU Logic Flow...............212
4.4.4.2.2.9 Filter Entity State Maximum Velocity CSU Algorithms...............214
4.4.4.2.2.10 Filter Entity State Maximum Velocity CSU Limitations...............214
4.5 Hashing Algorithms CSC (DL-CSC-5)........................................214
4.5.1 Initialize Hash Table CSU...........................................................216
4.5.1.1 Initialize Hash Table Design Specification/Constraints.................217
4.5.1.2 Initialize Hash Table Design........................................................217
4.5.1.2.1 Initialize Hash Table Input/Output Data Elements........................217
4.5.1.2.2 Initialize Hash Table Local Data Elements...................................217
4.5.1.2.3 Initialize Hash Table Global Data Elements.................................218
4.5.1.2.4 Initialize Hash Table Local and Shared Data Structures................218
4.5.1.2.5 Initialize Hash Table Interrupts and Signals.................................218
4.5.1.2.6 Initialize Hash Table Error Handling...........................................218
4.5.1.2.7 Initialize Hash Table Use of Other Elements................................218
4.5.1.2.8 Initialize Hash Table Logic Flow.................................................218
4.5.1.2.9 Initialize Hash Table Algorithms..................................................222
4.5.1.2.10 Initialize Hash Table Limitations.................................................222
4.5.2 Calculate Hash Table Address CSU.............................................222
4.5.2.1 Calculate Hash Table Address Design Specification/Constraints. .222
4.5.2.2 Calculate Hash Table Address Design..........................................222
4.5.2.2.1 Calculate Hash Table Address Input/Output Data Elements..........223
4.5.2.2.2 Calculate Hash Table Address Local Data Elements.....................223
4.5.2.2.3 Calculate Hash Table Address Global Data Elements...................223

| | | |
|---|---|---|
| 4.5.2.2.4 | Calculate Hash Table Address Local and Shared Data Structures. | 223 |
| 4.5.2.2.5 | Calculate Hash Table Address Interrupts and Signals | 224 |
| 4.5.2.2.6 | Calculate Hash Table Address Error Handling | 224 |
| 4.5.2.2.7 | Calculate Hash Table Address Use of Other Elements | 224 |
| 4.5.2.2.8 | Calculate Hash Table Address Logic Flow | 224 |
| 4.5.2.2.9 | Calculate Hash Table Address Algorithms | 225 |
| 4.5.2.2.10 | Calculate Hash Table Address Limitations | 226 |
| 4.5.3 | Get Item CSU | 226 |
| 4.5.3.1 | Get Item Design Specification/Constraints | 226 |
| 4.5.3.2 | Get Item Design | 226 |
| 4.5.3.2.1 | Get Item Input/Output Data Elements | 227 |
| 4.5.3.2.2 | Get Item Local Data Elements | 227 |
| 4.5.3.2.3 | Get Item Global Data Elements | 228 |
| 4.5.3.2.4 | Get Item Local and Shared Data Structures | 228 |
| 4.5.3.2.5 | Get Item Interrupts and Signals | 228 |
| 4.5.3.2.6 | Get Item Error Handling | 228 |
| 4.5.3.2.7 | Get Item Use of Other Elements | 228 |
| 4.5.3.2.8 | Get Item Logic Flow | 229 |
| 4.5.3.2.9 | Get Item Algorithms | 232 |
| 4.5.3.2.10 | Get Item Limitations | 232 |
| 4.5.4 | Put Item CSU | 232 |
| 4.5.4.1 | Put Item Design Specification/Constraints | 232 |
| 4.5.4.2 | Put Item Design | 232 |
| 4.5.4.2.1 | Put Item Input/Output Data Elements | 233 |
| 4.5.4.2.2 | Put Item Local Data Elements | 233 |
| 4.5.4.2.3 | Put Item Global Data Elements | 233 |
| 4.5.4.2.4 | Put Item Local and Shared Data Structures | 233 |
| 4.5.4.2.5 | Put Item Interrupts and Signals | 234 |
| 4.5.4.2.6 | Put Item Error Handling | 234 |
| 4.5.4.2.8 | Put Item Use of Other Elements | 234 |
| 4.5.4.2.8 | Put Item Logic Flow | 234 |
| 4.5.4.2.9 | Put Item Algorithms | 237 |
| 4.5.4.2.10 | Put Item Limitations | 237 |
| 4.6 | Smoothing Algorithms CSC (DL-CSC-6) | 237 |
| 4.6.1 | Smooth Entity CSU (DL-CSU-6.1) | 239 |
| 4.6.1.1 | Smooth Entity CSU Design Specification/Constraints | 240 |
| 4.6.1.2 | Smooth Entity CSU Design | 240 |
| 4.6.1.2.1 | Smooth Entity CSU Input/Output Data Elements | 240 |
| 4.6.1.2.2 | Smooth Entity CSU Local Data Elements | 240 |
| 4.6.1.2.3 | Smooth Entity CSU Global Data Elements | 241 |
| 4.6.1.2.4 | Smooth Entity CSU Local and Shared Data Structures | 241 |
| 4.6.1.2.5 | Smooth Entity CSU Interrupts and Signals | 241 |
| 4.6.1.2.6 | Smooth Entity CSU Error Handling | 241 |
| 4.6.1.2.7 | Smooth Entity CSU Use of Other Elements | 241 |
| 4.6.1.2.8 | Smooth Entity CSU Logic Flow | 242 |
| 4.6.1.2.9 | Smooth Entity CSU Algorithms | 245 |

4.6.1.2.10 Smooth Entity CSU Limitations....245
4.6.2 Rate Limiter CSU (DL-CSU-6.2)....246
4.6.2.1 Rate Limiter CSU Design Specification/Constraints....246
4.6.2.2 Rate Limiter CSU Design....246
4.6.2.2.1 Rate Limiter CSU Input/Output Data Elements....246
4.6.2.2.2 Rate Limiter CSU Local Data Elements....247
4.6.2.2.3 Rate Limiter CSU Global Data Elements....247
4.6.2.2.4 Rate Limiter CSU Local and Shared Data Structures....248
4.6.2.2.5 Rate Limiter CSU Interrupts and Signals....248
4.6.2.2.6 Rate Limiter CSU Error Handling....248
4.6.2.2.7 Rate Limiter CSU Use of Other Elements....248
4.6.2.2.8 Rate Limiter CSU Logic Flow....248
4.6.2.2.9 Rate Limiter CSU Algorithms....252
4.6.2.2.10 Rate Limiter CSU Limitations....253
4.6.3 Alpha-Beta Filter CSU (DL-CSU-6.3)....253
4.6.3.1 Alpha-Beta Filter CSU Design Specification/Constraints....254
4.6.3.2 Alpha-Beta Filter CSU Design....254
4.6.3.2.1 Alpha-Beta Filter CSU Input/Output Data Elements....254
4.6.3.2.2 Alpha-Beta Filter CSU Local Data Elements....255
4.6.3.2.3 Alpha-Beta Filter CSU Global Data Elements....255
4.6.3.2.4 Alpha-Beta Filter CSU Local and Shared Data Structures....255
4.6.3.2.5 Alpha-Beta Filter CSU Interrupts and Signals....255
4.6.3.2.6 Alpha-Beta Filter CSU Error Handling....255
4.6.3.2.7 Alpha-Beta Filter CSU Use of Other Elements....255
4.6.3.2.8 Alpha-Beta Filter CSU Logic Flow....255
4.6.3.2.9 Alpha-Beta Filter CSU Algorithms....257
4.6.3.2.10 Alpha-Beta Filter CSU Limitations....257
4.6.4 Set Alpha-Beta Filter Coefficient CSU (DL-CSU-6.4)....257
4.6.4.1 Set Alpha-Beta Filter Coefficient CSU Design Specification/Constraints....258
4.6.4.2 Set Alpha-Beta Filter Coefficient CSU Design....258
4.6.4.2.1 Set Alpha-Beta Filter Coefficient CSU Input/Output Data Elements....258
4.6.4.2.2 Set Alpha-Beta Filter Coefficient CSU Local Data Elements....259
4.6.4.2.3 Set Alpha-Beta Filter Coefficient CSU Global Data Elements....259
4.6.4.2.4 Set Alpha-Beta Filter Coefficient CSU Local and Shared Data Structures....260
4.6.4.2.5 Set Alpha-Beta Filter Coefficient CSU Interrupts and Signals....260
4.6.4.2.6 Set Alpha-Beta Filter Coefficient CSU Error Handling....260
4.6.4.2.7 Set Alpha-Beta Filter Coefficient CSU Use of Other Elements....260
4.6.4.2.8 Set Alpha-Beta Filter Coefficient CSU Logic Flow....260
4.6.4.2.9 Set Alpha-Beta Filter Coefficient CSU Algorithms....261
4.6.4.2.10 Set Alpha-Beta Filter Coefficient CSU Limitations....262
4.6.5 Get Alpha-Beta Filter Coefficient CSU (DL-CSU-6.5)....262
4.6.5.1 Get Alpha-Beta Filter Coefficient CSU Design Specification/Constraints....262

4.6.5.2 Get Alpha-Beta Filter Coefficient CSU Design....262
4.6.5.2.1 Get Alpha-Beta Filter Coefficient CSU Input/Output Data Elements....262
4.6.5.2.2 Get Alpha-Beta Filter Coefficient CSU Local Data Elements....263
4.6.5.2.3 Get Alpha-Beta Filter Coefficient CSU Global Data Elements....263
4.6.5.2.4 Get Alpha-Beta Filter Coefficient CSU Local and Shared Data Structures....263
4.6.5.2.5 Get Alpha-Beta Filter Coefficient CSU Interrupts and Signals....263
4.6.5.2.6 Get Alpha-Beta Filter Coefficient CSU Error Handling....263
4.6.5.2.7 Get Alpha-Beta Filter Coefficient CSU Use of Other Elements....264
4.6.5.2.8 Get Alpha-Beta Filter Coefficient CSU Logic Flow....264
4.6.5.2.9 Get Alpha-Beta Filter Coefficient CSU Algorithms....265
4.6.5.2.10 Get Alpha-Beta Filter Coefficient CSU Limitations....265
4.7 Sorting Algorithms CSC (DL-CSC-7)....265
4.7.1 Sort Routines CSC (DL-CSC-4.7.1)....265
4.7.1.1 Sort List CSU (DL-CSU-4.7.1.1)....268
4.7.1.1.1 Sort List CSU Design Specification/Constraints....268
4.7.1.1.2 Sort List CSU Design....268
4.7.1.1.2.1 Sort List CSU Input/Output Data Elements....268
4.7.1.1.2.2 Sort List CSU Local Data Elements....269
4.7.1.1.2.3 Sort List CSU Global Data Elements....269
4.7.1.1.2.4 Sort List CSU Local and Shared Data Structures....269
4.7.1.1.2.5 Sort List CSU Interrupts and Signals....269
4.7.1.1.2.6 Sort List CSU Error Handling....270
4.7.1.1.2.7 Sort List CSU Use of Other Elements....270
4.7.1.1.2.8 Sort List CSU Logic Flow....270
4.7.1.1.2.9 Sort List CSU Algorithms....274
4.7.1.1.2.10 Sort List CSU Limitations....274
4.7.2 Sort By Distance CSC (DL-CSC-4.7.2)....275
4.7.2.1 Sort Entity State Distance CSU (DL-CSU-7.2.1)....278
4.7.2.1.1 Sort Entity State Distance CSU Design Specification/Constraints.278
4.7.2.1.2 Sort Entity State Distance CSU Design....278
4.7.2.1.2.1 Sort Entity State Distance CSU Input/Output Data Elements....278
4.7.2.1.2.2 Sort Entity State Distance CSU Local Data Elements....279
4.7.2.1.2.3 Sort Entity State Distance CSU Global Data Elements....280
4.7.2.1.2.4 Sort Entity State Distance CSU Local and Shared Data Structures280
4.7.2.1.2.5 Sort Entity State Distance CSU Interrupts and Signals....280
4.7.2.1.2.6 Sort Entity State Distance CSU Error Handling....281
4.7.2.1.2.7 Sort Entity State Distance CSU Use of Other Elements....281
4.7.2.1.2.8 Sort Entity State Distance CSU Logic Flow....281
4.7.2.1.2.9 Sort Entity State Distance CSU Algorithms....286
4.7.2.1.2.10 Sort Entity State Distance CSU Limitations....287
4.7.2.2 Sort Fire Distance CSU (DL-CSU-7.2.2)....287
4.7.2.2.1 Sort Fire Distance CSU Design Specification/Constraints....287
4.7.2.2.2 Sort Fire Distance CSU Design....287
4.7.2.2.2.1 Sort Fire Distance CSU Input/Output Data Elements....288

4.7.2.2.2.2 Sort Fire Distance CSU Local Data Elements....288
4.7.2.2.2.3 Sort Fire Distance CSU Global Data Elements....289
4.7.2.2.2.4 Sort Fire Distance CSU Local and Shared Data Structures....289
4.7.2.2.2.5 Sort Fire Distance CSU Interrupts and Signals....290
4.7.2.2.2.6 Sort Fire Distance CSU Error Handling....290
4.7.2.2.2.7 Sort Fire Distance CSU Use of Other Elements....290
4.7.2.2.2.8 Sort Fire Distance CSU Logic Flow....290
4.7.2.2.2.9 Sort Fire Distance CSU Algorithms....296
4.7.2.2.2.10 Sort Fire Distance CSU Limitations....296
4.7.2.3 Sort Detonation Distance CSU (DL-CSU-7.2.3)....296
4.7.2.3.1 Sort Detonation Distance CSU Design Specification/Constraints..296
4.7.2.3.2 Sort Detonation Distance CSU Design....297
4.7.2.3.2.1 Sort Detonation Distance CSU Input/Output Data Elements....297
4.7.2.3.2.2 Sort Detonation Distance CSU Local Data Elements....297
4.7.2.3.2.3 Sort Detonation Distance CSU Global Data Elements....299
4.7.2.3.2.4 Sort Detonation Distance CSU Local and Shared Data Structures.299
4.7.2.3.2.5 Sort Detonation Distance CSU Interrupts and Signals....299
4.7.2.3.2.6 Sort Detonation Distance CSU Error Handling....299
4.7.2.3.2.7 Sort Detonation Distance CSU Use of Other Elements....300
4.7.2.3.2.8 Sort Detonation Distance CSU Logic Flow....300
4.7.2.3.2.9 Sort Detonation Distance CSU Algorithms....305
4.7.2.3.2.10 Sort Detonation Distance CSU Limitations....306
4.7.2.4 Sort Laser Distance CSU (DL-CSU-7.2.4)....306
4.7.2.4.1 Sort Laser Distance CSU Design Specification/Constraints....306
4.7.2.4.2 Sort Laser Distance CSU Design....306
4.7.2.4.2.1 Sort Laser Distance CSU Input/Output Data Elements....307
4.7.2.4.2.2 Sort Laser Distance CSU Local Data Elements....307
4.7.2.4.2.3 Sort Laser Distance CSU Global Data Elements....308
4.7.2.4.2.4 Sort Laser Distance CSU Local and Shared Data Structures....308
4.7.2.4.2.5 Sort Laser Distance CSU Interrupts and Signals....309
4.7.2.4.2.6 Sort Laser Distance CSU Error Handling....309
4.7.2.4.2.7 Sort Laser Distance CSU Use of Other Elements....309
4.7.2.4.2.8 Sort Laser Distance CSU Logic Flow....309
4.7.2.4.2.9 Sort Laser Distance CSU Algorithms....314
4.7.2.4.2.10 Sort Laser Distance CSU Limitations....315
4.7.2.5 Sort Transmitter Distance CSU (DL-CSU-7.2.5)....315
4.7.2.5.1 Sort Transmitter Distance CSU Design Specification/Constraints.315
4.7.2.5.2 Sort Transmitter Distance CSU Design....315
4.7.2.5.2.1 Sort Transmitter Distance CSU Input/Output Data Elements....316
4.7.2.5.2.2 Sort Transmitter Distance CSU Local Data Elements....316
4.7.2.5.2.3 Sort Transmitter Distance CSU Global Data Elements....317
4.7.2.5.2.4 Sort Transmitter Distance CSU Local and Shared Data Structures317
4.7.2.5.2.5 Sort Transmitter Distance CSU Interrupts and Signals....318
4.7.2.5.2.6 Sort Transmitter Distance CSU Error Handling....318
4.7.2.5.2.7 Sort Transmitter Distance CSU Use of Other Elements....318
4.7.2.5.2.8 Sort Transmitter Distance CSU Logic Flow....318

| | | |
|---|---|---|
| 4.7.2.5.2.9 | Sort Transmitter Distance CSU Algorithms | 323 |
| 4.7.2.5.2.10 | Sort Transmitter Distance CSU Limitations | 324 |
| 4.7.3 | Sort By Velocity CSC (DL-CSC-4.7.3) | 324 |
| 4.7.3.1 | Sort Entity State Velocity CSU (DL-CSU-7.3.1) | 326 |
| 4.7.3.1.1 | Sort Entity State Velocity CSU Design Specification/Constraints | 326 |
| 4.7.3.1.2 | Sort Entity State Velocity CSU Design | 326 |
| 4.7.3.1.2.1 | Sort Entity State Velocity CSU Input/Output Data Elements | 326 |
| 4.7.3.1.2.2 | Sort Entity State Velocity CSU Local Data Elements | 327 |
| 4.7.3.1.2.3 | Sort Entity State Velocity CSU Global Data Elements | 328 |
| 4.7.3.1.2.4 | Sort Entity State Velocity CSU Local and Shared Data Structures | 328 |
| 4.7.3.1.2.5 | Sort Entity State Velocity CSU Interrupts and Signals | 329 |
| 4.7.3.1.2.6 | Sort Entity State Velocity CSU Error Handling | 329 |
| 4.7.3.1.2.7 | Sort Entity State Velocity CSU Use of Other Elements | 329 |
| 4.7.3.1.2.8 | Sort Entity State Velocity CSU Logic Flow | 329 |
| 4.7.3.1.2.9 | Sort Entity State Velocity CSU Algorithms | 335 |
| 4.7.3.1.2.10 | Sort Entity State Velocity CSU Limitations | 335 |
| 5 | CSCI Data | 336 |
| 6 | DL CSCI Data Files | 338 |
| 7 | Notes | 338 |
| 7.1 | Acronyms and Abbreviations | 338 |

## Figures

Figure Title
Page


µ3.1-1 System Architecture : DL To Other CSCIs........................................5
3.1.1-1 DL CSCI Hierarchy Chart........................................6
3.1.1-2 Conversion Algorithms CSC Hierarchy Chart........................................6
3.1.1-3 Filtering Algorithms CSC Hierarchy Chart........................................7
3.1.1-4 Sorting Algorithms CSC Hierarchy Chart........................................7
3.1.1-5 DL CSCI CSC-to-CSC Relationships........................................8
3.2.1.1-1 Coordinate System Conversion CSC Control Flow........................................12
3.2.1.1-2 Coordinate System Conversion CSC Data Flow........................................12
3.2.1.2-1 Orientation System Conversion CSC Control Flow........................................14
3.2.1.2-2 Orientation System Conversion CSC Data Flow........................................14
3.2.2-1 Dead-Reckoning Algorithms CSC Control Flow........................................16
3.2.2-2 Dead-Reckoning Algorithms CSC Data Flow........................................16
3.2.3-1 Calculation Algorithms CSC Control Flow........................................18
3.2.3-2 Calculation Algorithms CSC Data Flow........................................19
3.2.4-1 Filtering Algorithms CSC Control Flow........................................20
3.2.4-2 Filtering Algorithms CSC Data Flow........................................21
3.2.5-1 Hashing Algorithms CSC Control Flow........................................23
3.2.5-2 Hashing Algorithms CSC Data Flow........................................23
3.2.6-1 Smoothing Algorithms CSC Control Flow........................................25
3.2.6-2 Smoothing Algorithms CSC Data Flow........................................25
3.2.7.1-1 Sorting Algorithms CSC Control Flow........................................26
3.2.7.1-2 Sorting Algorithms Data Flow CSC........................................27
4.1.1-1 Coordinate System Conversion CSC Computer Software Units.....30
4.1.1-2 Coordinate System Conversion CSC Execution Control Flow.......31
4.1.1-3 Coordinate System Conversion CSC Data Flow........................................32
4.1.1.1.2.8-1 Geocentric-Geodetic Conversion CSU Flowchart........................................35
4.1.1.2.2.8-1 Geocentric-UTM Conversion CSU Flowchart........................................38
4.1.1.3.2.8-1 Geocentric-Local Conversion CSU Flowchart........................................42
4.1.1.4.2.8-1 Geocentric-Entity Conversion CSU Flowchart........................................46
4.1.1.5.2.8-1 Geodetic-Geocentric Conversion CSU Flowchart........................................49
4.1.1.6.2.8-1 UTM-Geocentric Conversion CSU Flowchart........................................52
4.1.1.7.2.8-1 Local-Geocentric Conversion CSU Flowchart........................................56
4.1.1.8.2.8-1 Entity-Geocentric Conversion CSU Flowchart........................................60
4.1.2-1 Orientation System Conversion CSC Computer Software Units….61
4.1.2-2 Orientation System Conversion CSC Execution Control Flow.......62
4.1.2-3 Orientation System Conversion CSC Data Flow........................................62
4.1.2.1.2.8-1 Euler-Orientation Conversion CSU Flowchart........................................65
4.1.2.2.2.8-1 Orientation-Euler Conversion CSU Flowchart........................................69
4.2-1 Dead-Reckoning Algorithms CSC Computer Software Units..........71
4.2-2 Dead-Reckoning Algorithms CSC Execution Control Flow............72
4.2-3 Dead-Reckoning Algorithms CSC Data Flow........................................73

| | | |
|---|---|---|
| 4.2.1.2.8-1 | Update Dead Reckoned Position CSU Flowchart | 76 |
| 4.2.3.2.8-1 | DRM FPW CSU Flowchart | 80 |
| 4.2.4.2.8-1 | DRM FVW CSU Flowchart | 84 |
| 4.2.5.2.8-1 | DRM RPW CSU Flowchart | 88 |
| 4.2.6.2.8-1 | DRM RVW CSU Flowchart | 92 |
| 4.3-1 | Calculation Algorithms CSC Computer Software Units | 93 |
| 4.3-2 | Calculation Algorithms CSC Execution Control | 94 |
| 4.3-3 | Calculation Algorithms CSC Data Flow | 95 |
| 4.3.1.2.8-1 | Calculate Distance CSU Flowchart | 98 |
| 4.3.2.2.8-1 | Calculate Azimuth CSU Flowchart | 102 |
| 4.3.2.2.9-1 | Example Azimuth Calculation | 104 |
| 4.3.3.2.8-1 | Calculate Velocity CSU Flowchart | 107 |
| 4.3.4.2.8-1 | Calculate Elevation CSU Flowchart | 111 |
| 4.3.4.2.9-1 | Example Elevation Calculation | 113 |
| 4.3.5.2.8-1 | Calculate Azimuth and Elevation CSU Flowchart | 116 |
| 4.4.1-1 | Filtering Routines CSC Computer Software Units | 118 |
| 4.4.1-2 | Filtering Routines CSC Execution Control | 118 |
| 4.4.1-3 | Filtering Routines CSC Data Flow | 119 |
| 4.4.1.1.2.8-1 | Filter Distance CSU Flowchart (Part 1) | 123 |
| 4.4.1.1.2.8-2 | Filter Distance CSU Flowchart (Part 2) | 124 |
| 4.4.1.2.2.8-1 | Filter Azimuth CSU Flowchart (Part 1) | 128 |
| 4.4.1.2.2.8-2 | Filter Azimuth CSU Flowchart (Part 2) | 129 |
| 4.4.1.3.2.8-1 | Filter Minimum Velocity CSU Flowchart (Part 1) | 133 |
| 4.4.1.3.2.8-2 | Filter Minimum Velocity CSU Flowchart (Part 2) | 134 |
| 4.4.1.4.2.8-1 | Filter Maximum Velocity CSU Flowchart (Part 1) | 138 |
| 4.4.1.4.2.8-2 | Filter Maximum Velocity CSU Flowchart (Part 2) | 139 |
| 4.4.1.5.2.8-1 | Filter Elevation CSU Flowchart (Part 1) | 143 |
| 4.4.1.5.2.8-2 | Filter Elevation CSU Flowchart (Part 2) | 144 |
| 4.4.1.6.2.8-1 | Filter Azimuth and Elevation CSU Flowchart (Part 1) | 149 |
| 4.4.1.6.2.8-2 | Filter Azimuth and Elevation CSU Flowchart (Part 2) | 150 |
| 4.4.2-1 | Filtering Routines CSC Computer Software Units | 152 |
| 4.4.2-2 | Filter List By Distance CSC Execution Control | 152 |
| 4.4.2-3 | Filter List By Distance CSC Data Flow | 153 |
| 4.4.2.1.2.8-1 | Filter Entity State Distance CSU Flowchart | 157 |
| 4.4.2.2.2.8-1 | Filter Fire Distance CSU Flowchart | 161 |
| 4.4.2.3.2.8-1 | Filter Detonation Distance CSU Flowchart | 165 |
| 4.4.2.4.2.8-1 | Filter Laser Distance CSU Flowchart | 169 |
| 4.4.2.5.2.8-1 | Filter Transmitter Distance CSU Flowchart | 173 |
| 4.4.3-1 | Filter List By Orientation CSC Computer Software Units | 174 |
| 4.4.3-2 | Filter List By Orientation CSC Execution Control | 175 |
| 4.4.3-3 | Filter List By Orientation CSC Data Flow | 176 |
| 4.4.3.1.2.8-1 | Filter Entity State Azimuth CSU Flowchart | 180 |
| 4.4.3.2.2.8-1 | Filter Fire Azimuth CSU Flowchart | 184 |
| 4.4.3.3.2.8-1 | Filter Detonation Azimuth CSU Flowchart | 188 |
| 4.4.3.4.2.8-1 | Filter Laser Azimuth CSU Flowchart | 192 |
| 4.4.3.5.2.8-1 | Filter Transmitter Azimuth CSU Flowchart | 196 |

4.4.3.6.2.8-1 Filter Entity State Elevation CSU Flowchart................................200
4.4.3.7.2.8-1 Filter Entity State Azimuth and Elevation CSU Flowchart............204
4.4.4-1 Filter List By Velocity CSC Computer Software Units.................205
4.4.4-2 Execution Control of the Filter List By Velocity CSC..................206
4.4.4-3 Filter List By Velocity CSC Data Flow........................................206
4.4.4.1.2.8-1 Filter Entity State Minimum Velocity CSU Flowchart.................209
4.4.4.2.2.8-1 Filter Entity State Maximum Velocity CSU Flowchart.................213
4.5-1 Hashing Algorithms CSC Computer Software Units.....................214
4.6-2 Smoothing Algorithms CSC Execution Control............................215
4.6-3 Smoothing Algorithms CSC Data Flow........................................216
4.5.1.2.8-1 Initialize Hash Table CSU Flowchart (Part 1)...............................219
4.5.1.2.8-2 Initialize Hash Table CSU Flowchart (Part 2)...............................220
4.5.1.2.8-3 Initialize Hash Table CSU Flowchart (Part 3)...............................221
4.5.2.2.8-1 Calculate Hash Table Address CSU Flowchart.............................225
4.5.3.2.8-1 Get Item CSU Flowchart (Part 1).................................................230
4.5.3.2.8-2 Get Item CSU Flowchart (Part 2).................................................231
4.5.4.2.8-1 Put Item CSU Flowchart (Part 1)..................................................235
4.5.4.2.8-1 Put Item CSU Flowchart (Part 2)..................................................236
4.6-1 Smoothing Algorithms CSC Computer Software Units................237
4.6-2 Smoothing Algorithms CSC Execution Control............................238
4.6-3 Smoothing Algorithms CSC Data Flow........................................239
4.6.1.2.8-1 Smooth Entity CSU Flowchart (Part 1).......................................243
4.6.1.2.8-2 Smooth Entity CSU Flowchart (Part 2)........................................244
4.6.1.2.8-3 Smooth Entity CSU Flowchart (Part 3)........................................245
4.6.2.2.8-1 Rate Limiter CSU Flowchart (Part 1)...........................................249
4.6.2.2.8-2 Rate Limiter CSU Flowchart (Part 2)...........................................250
4.6.2.2.8-3 Rate Limiter CSU Flowchart (Part 3)...........................................251
4.6.2.2.8-4 Rate Limiter CSU Flowchart (Part 4)...........................................252
4.6.3.2.8-1 Alpha-Beta Filter CSU Flowchart (Part 1)...................................256
4.6.4.2.8-1 Set Alpha-Beta Filter Coefficient CSU Flowchart.........................261
4.6.5.2.8-1 Get Alpha-Beta Filter Coefficient CSU Flowchart........................264
4.7.1-1 Sort Routines CSC Computer Software Units...............................266
4.7.1-2 Sort Routines CSC Execution Control..........................................266
4.7.1-3 Sort Routines CSC Data Flow......................................................267
4.7.1.1.2.8-1 Sort List CSU Flowchart (Part 1).................................................271
4.7.1.1.2.8-2 Sort List CSU Flowchart (Part 2)................................................272
4.7.1.1.2.8-3 Sort List CSU Flowchart (Part 3).................................................273
4.7.2-1 Sort By Distance CSC Computer Software Units..........................275
4.7.2-2 Sort By Distance CSC Execution Control.....................................276
4.7.2-3 Sort By Distance CSC Data Flow.................................................277
4.7.2.1.2.8-1 Sort Entity State Distance CSU Flowchart (Part 1)........................282
4.7.2.1.2.8-2 Sort Entity State Distance CSU Flowchart (Part 2)........................283
4.7.2.1.2.8-3 Sort Entity State Distance CSU Flowchart (Part 3)........................284
4.7.2.1.2.8-4 Sort Entity State Distance CSU Flowchart (Part 4)........................285
4.7.2.1.2.8-5 Sort Entity State Distance CSU Flowchart (Part 5)........................286
4.7.2.2.2.8-1 Sort Fire Distance CSU Flowchart (Part 1)...................................291

4.7.2.2.2.8-2 Sort Fire Distance CSU Flowchart (Part 2)....292
4.7.2.2.2.8-3 Sort Fire Distance CSU Flowchart (Part 3)....293
4.7.2.2.2.8-4 Sort Fire Distance CSU Flowchart (Part 4)....294
4.7.2.2.2.8-5 Sort Fire Distance CSU Flowchart (Part 5)....295
4.7.2.3.2.8-1 Sort Detonation Distance CSU Flowchart (Part 1)....301
4.7.2.3.2.8-2 Sort Detonation Distance CSU Flowchart (Part 2)....302
4.7.2.3.2.8-3 Sort Detonation Distance CSU Flowchart (Part 3)....303
4.7.2.3.2.8-4 Sort Detonation Distance CSU Flowchart (Part 4)....304
4.7.2.3.2.8-5 Sort Detonation Distance CSU Flowchart (Part 5)....305
4.7.2.4.2.8-1 Sort Laser Distance CSU Flowchart (Part 1)....310
4.7.2.4.2.8-2 Sort Laser Distance CSU Flowchart (Part 2)....311
4.7.2.4.2.8-3 Sort Laser Distance CSU Flowchart (Part 3)....312
4.7.2.4.2.8-4 Sort Laser Distance CSU Flowchart (Part 4)....313
4.7.2.4.2.8-5 Sort Laser Distance CSU Flowchart (Part 5)....314
4.7.2.5.2.8-1 Sort Transmitter Distance CSU Flowchart (Part 1)....319
4.7.2.5.2.8-2 Sort Transmitter Distance CSU Flowchart (Part 2)....320
4.7.2.5.2.8-3 Sort Transmitter Distance CSU Flowchart (Part 3)....321
4.7.2.5.2.8-4 Sort Transmitter Distance CSU Flowchart (Part 4)....322
4.7.2.5.2.8-5 Sort Transmitter Distance CSU Flowchart (Part 5)....323
4.7.3-1 Sort By Velocity CSC Computer Software Units....325
4.7.3-2 Sort By Velocity CSC Execution Control....325
4.7.3-3 Sort By Velocity CSC Data Flow....326
4.7.3.1.2.8-1 Sort Entity State Velocity CSU Flowchart (Part 1)....330
4.7.3.1.2.8-2 Sort Entity State Velocity CSU Flowchart (Part 2)....331
4.7.3.1.2.8-3 Sort Entity State Velocity CSU Flowchart (Part 3)....332
4.7.3.1.2.8-4 Sort Entity State Velocity CSU Flowchart (Part 4)....333
4.7.3.1.2.8-5 Sort Entity State Velocity CSU Flowchart (Part 5)....334

## Tables

| Table | Title Page |
|---|---|
| µ2.1-1 | Government Documents....3 |
| 2.2-1 | Non-Government Documents....3 |
| 3.1-1 | External Interfaces for the DL CSCI....4 |
| 3.1.1-1 | CSCs Defined for the DL CSCI....9 |
| 3.1.1-2 | CSC-to-CSC Interfaces for the DL CSCI....10 |
| 4.1.1.1.2.1-1 | Geocentric-Geodetic Conversion CSU I/O Data....33 |
| 4.1.1.2.2.1-1 | Geocentric-UTM Conversion CSU I/O Data....36 |
| 4.1.1.3.2.1-1 | Geocentric-Local Conversion CSU I/O Data....40 |
| 4.1.1.4.2.1-1 | Geocentric-Entity Conversion CSU I/O Data....44 |
| 4.1.1.5.2.1-1 | Geodetic-Geocentric Conversion CSU I/O Data....47 |
| 4.1.1.6.2.1-1 | UTM-Geocentric Conversion CSU I/O Data....50 |
| 4.1.1.7.2.1-1 | Local-Geocentric Conversion CSU I/O Data....54 |
| 4.1.1.8.2.1-1 | Entity-Geocentric Conversion CSU I/O Data....58 |
| 4.1.2.1.2.1-1 | Euler-Orientation Conversion CSU I/O Data....63 |
| 4.1.2.2.2.1-1 | Orientation-Euler Conversion CSU I/O Data....67 |
| 4.2.1.2.1-1 | Update Dead Reckoned Position CSU I/O Data....74 |
| 4.2.1.2.2-1 | Local Data Descriptions for the Update Dead Reckoned Position CSU....75 |
| 4.2.3.2.1-1 | DRM FPW CSU I/O Data....78 |
| 4.2.4.2.1-1 | DRM FVW CSU I/O Data....82 |
| 4.2.5.2.1-1 | DRM RPW CSU I/O Data....86 |
| 4.2.6.2.1-1 | DRM RVW CSU I/O Data....90 |
| 4.3.1.1-1 | Requirements Allocated to the Calculate Distance CSU....96 |
| 4.3.1.2.1-1 | Calculate Distance CSU I/O....96 |
| 4.3.2.1-1 | Requirements Allocated to the Calculate Azimuth CSU....99 |
| 4.3.2.2.1-1 | Calculate Azimuth CSU I/O Data....100 |
| 4.3.2.2.2-1 | Calculate Azimuth CSU Local Data Descriptions....100 |
| 4.3.3.1-1 | Requirements Allocated to the Calculate Velocity CSU....104 |
| 4.3.3.2.1-1 | Calculate Velocity CSU I/O Data....105 |
| 4.3.4.1-1 | Requirements Allocated to the Calculate Elevation CSU....108 |
| 4.3.4.2.1-1 | Calculate Elevation CSU I/O Data....109 |
| 4.3.4.2.2-1 | Calculate Elevation CSU Local Data Descriptions....110 |
| 4.3.5.1-1 | Requirements Allocated to the Calculate Azimuth and Elevation CSU....114 |
| 4.3.5.2.1-1 | Calculate Azimuth and Elevation CSU I/O Data....114 |
| 4.3.5.2.2-1 | Local Data Descriptions for the Calculate Azimuth and Elevation CSU....115 |
| 4.4.1.1.1-1 | Requirements Allocated to the Filter Distance CSU....120 |
| 4.4.1.1.2.1-1 | Filter Distance CSU I/O Data....121 |
| 4.4.1.2.1-1 | Requirements Allocated to the Filter Azimuth CSU....125 |
| 4.4.1.2.2.1-1 | Filter Azimuth CSU I/O Data....126 |

4.4.1.3.1-1 Requirements Allocated to the Filter Minimum Velocity CSU......131
4.4.1.3.2.1-1 Filter Minimum Velocity CSU I/O Data.......................................131
4.4.1.4.1-1 Requirements Allocated to the Filter Maximum Velocity CSU.....135
4.4.1.4.2.1-1 Filter Maximum Velocity CSU I/O Data......................................136
4.4.1.5.1-1 Requirements Allocated to the Filter Elevation CSU.....................140
4.4.1.5.2.1-1 Filter Elevation CSU I/O Data......................................................141
4.4.1.6.1-1 Requirements Allocated to the Filter Azimuth and Elevation CSU ........................................................................................... 146
4.4.1.6.2.1-1 Filter Azimuth and Elevation CSU I/O Data................................147
4.4.2.1.1-1 Requirements Allocated to the Filter Entity State Distance CSU...154
4.4.2.1.2.1-1 Filter Entity State Distance CSU I/O Data....................................155
4.4.2.2.1-1 Requirements Allocated to the Filter Fire Distance CSU...............158
4.4.2.2.2.1-1 I/O Filter Fire Distance CSU I/O Data..........................................159
4.4.2.3.1-1 Requirements Allocated to the Filter Detonation Distance CSU....162
4.4.2.3.2.1-1 Filter Detonation Distance CSU I/O Data.....................................163
4.4.2.4.1-1 Requirements Allocated to the Filter Laser Distance CSU............166
4.4.2.4.2.1-1 Filter Laser Distance CSU I/O Data.............................................167
4.4.2.5.1-1 Requirements Allocated to the Filter Transmitter Distance CSU...170
4.4.2.5.2.1-1 Filter Transmitter Distance CSU I/O Data....................................171
4.4.3.1.1-1 Requirements Allocated to the Filter Entity State Azimuth CSU...177
4.4.3.1.2.1-1 Filter Entity State Azimuth CSU I/O Data....................................178
4.4.3.2.1-1 Requirements Allocated to the Filter Fire Azimuth CSU...............181
4.4.3.2.2.1-1 Filter Fire Azimuth CSU I/O Data................................................182
4.4.3.3.1-1 Requirements Allocated to the Filter Detonation Azimuth CSU....185
4.4.3.3.2.1-1 Filter Detonation Azimuth CSU I/O Data.....................................186
4.4.3.4.1-1 Requirements Allocated to the Filter Laser Azimuth CSU............189
4.4.3.4.2.1-1 Filter Laser Azimuth CSU I/O Data.............................................190
4.4.3.5.1-1 Requirements Allocated to the Filter Transmitter Azimuth CSU...193
4.4.3.5.2.1-1 Filter Transmitter Azimuth CSU I/O Data....................................194
4.4.3.6.1-1 Requirements Allocated to the Filter Entity State Elevation CSU. 197
4.4.3.6.2.1-1 Filter Entity State Elevation CSU I/O Data...................................198
4.4.3.7.1-1 Requirements Allocated to the Filter Entity State Azimuth and Elevation CSU............................................................................201
4.4.3.7.2.1-1 Filter Entity State Azimuth and Elevation CSU I/O Data..............202
4.4.4.1.1-1 Requirements Allocated to the Filter Entity State Minimum Velocity CSU..........................................................................................207
4.4.4.1.2.1-1 Filter Entity State Minimum Velocity CSU I/O Data.....................207
4.4.4.2.1-1 Requirements Allocated to the Filter Entity State Maximum Velocity CSU..........................................................................................210
4.4.4.2.2.1-1 Filter Entity State Maximum Velocity CSU I/O Data....................211
4.5.1.2.1-1 Initialize Hash Table CSU I/O Data.............................................217
4.6.2.2.2-1 Initialize Hash Table CSU Local Data Descriptions......................218
4.5.2.2.1-1 Calculate Hash Table Address CSU I/O Data...............................223
4.5.3.2.1-1 Get Item CSU I/O Data................................................................227
4.7.1.1.2.2-1 Local Data Descriptions for the Sort List CSU..............................228
4.5.4.2.1-1 Put Item CSU I/O Data.................................................................233

| | | |
|---|---|---|
| 4.7.1.1.2.2-1 | Local Data Descriptions for the Sort List CSU | 234 |
| 4.6.1.1-1 | Requirements Allocated to the Smooth Entity CSU | 240 |
| 4.6.1.2.1-1 | Smooth Entity CSU I/O Data | 240 |
| 4.6.1.2.3-1 | Smooth Entity CSU Global Data Elements | 241 |
| 4.6.2.1-1 | Requirements Allocated to the Rate Limiter CSU | 246 |
| 4.6.2.2.1-1 | Rate Limiter CSU I/O Data | 247 |
| 4.6.2.2.2-1 | Rate Limiter CSU Local Data Descriptions | 247 |
| 4.6.3.1-1 | Requirements Allocated to the Alpha-Beta Filter CSU | 254 |
| 4.6.3.2.1-1 | Alpha-Beta Filter CSU I/O Data | 254 |
| 4.6.3.2.3-1 | Alpha-Beta Filter CSU Global Data Elements | 255 |
| 4.6.4.1-1 | Requirements Allocated to the Set Alpha-Beta Filter Coefficient CSU | 258 |
| 4.6.4.2.1-1 | Set Alpha-Beta Filter Coefficient CSU I/O Data | 258 |
| 4.6.4.2.3-1 | Set Alpha-Beta Filter Coefficient CSU Global Data Elements | 259 |
| 4.6.5.1-1 | Requirements Allocated to the Get Alpha-Beta Filter Coefficient CSU | 262 |
| 4.6.5.2.1-1 | Get Alpha-Beta Filter Coefficient CSU I/O Data | 263 |
| 4.6.5.2.3-1 | Get Alpha-Beta Filter Coefficient CSU Global Data Elements | 263 |
| 4.7.1.-1 | Sort Routines CSC Generic Data Types | 267 |
| 4.7.1-2 | Sort Routines CSC Generic Functions | 267 |
| 4.7.1.1.1-1 | Requirements Allocated to the Sort List CSU | 268 |
| 4.7.1.1.2.1-1 | Sort List CSU I/O Data | 269 |
| 4.7.1.1.2.2-1 | Sort List CSU Local Data Descriptions | 269 |
| 4.7.2.1.1-1 | Requirements Allocated to the Sort Entity State Distance CSU | 278 |
| 4.7.2.1.2.1-1 | Sort Entity State Distance CSU I/O Data | 278 |
| 4.7.2.1.2.2-1 | Sort Entity State Distance CSU Local Data Descriptions | 279 |
| 4.7.2.1.2.2-2 | Size, Units, and Range of Local Data for the Sort Entity State Distance CSU | 279 |
| 4.7.2.1.2.2-3 | Sort Entity State Distance CSU Local Function Description | 279 |
| 4.7.2.1.2.2-4 | Scope Level of Local Declarations for the Sort Entity State Distance CSU | 280 |
| 4.7.2.1.2.4-1 | Entity State Pointer Array Data Structure | 280 |
| 4.7.2.1.2.4-2 | Sort Record Data Structure | 280 |
| 4.7.2.2.1-1 | Requirements Allocated to the Sort Fire Distance CSU | 287 |
| 4.7.2.2.2.1-1 | Sort Fire Distance CSU I/O Data | 288 |
| 4.7.2.2.2.2-1 | Sort Fire Distance CSU Local Data Descriptions | 288 |
| 4.7.2.2.2.2-2 | Size, Units, and Range of Local Data for the Sort Fire Distance CSU | 289 |
| 4.7.2.2.2.2-3 | Sort Fire Distance CSU Local Function Description | 289 |
| 4.7.2.2.2.2-4 | Scope Level of Local Declarations for the Sort Fire Distance CSU | 289 |
| 4.7.2.2.2.4-1 | Fire Pointer Array Data Structure | 290 |
| 4.7.2.2.2.4-2 | Sort Record Data Structure | 290 |
| 4.7.2.3.1-1 | Requirements Allocated to the Sort Detonation Distance CSU | 296 |
| 4.7.2.3.2.1-1 | Sort Detonation Distance CSU I/O Data | 297 |
| 4.7.2.3.2.2-1 | Sort Detonation Distance CSU Local Data Descriptions | 298 |

4.7.2.3.2.2-2 Size, Units, and Range of Local Data for the Sort Detonation Distance CSU....298
4.7.2.3.2.2-3 Sort Detonation Distance CSU Local Function Description....298
4.7.2.3.2.2-4 Scope Level of Local Declarations for the Sort Detonation Distance CSU....299
4.7.2.3.2.4-1 Detonation Pointer Array Data Structure....299
4.7.2.3.2.4-2 Sort Record Data Structure....299
4.7.2.4.1-1 Requirements Allocated to the Sort Laser Distance CSU....306
4.7.2.4.2.1-1 Sort Laser Distance CSU I/O Data....307
4.7.2.4.2.2-1 Sort Laser Distance CSU Local Data Descriptions....307
4.7.2.4.2.2-2 Size, Units, and Range of Local Data for the Sort Laser Distance CSU....308
4.7.2.4.2.2-3 Sort Laser Distance CSU Local Function Description....308
4.7.2.4.2.2-4 Scope Level of Local Declarations for the Sort Laser Distance CSU....308
4.7.2.4.2.4-1 Laser Pointer Array Data Structure....309
4.7.2.4.2.4-2 Sort Record Data Structure....309
4.7.2.5.1-1 Requirements Allocated to the Sort Transmitter Distance CSU....315
4.7.2.5.2.1-1 Sort Transmitter Distance CSU I/O Data....316
4.7.2.5.2.2-1 Sort Transmitter Distance CSU Local Data Descriptions....316
4.7.2.5.2.2-2 Size, Units, and Range of Local Data for the Sort Transmitter Distance CSU....317
4.7.2.5.2.2-3 Sort Transmitter Distance CSU Local Function Description....317
4.7.2.5.2.2-4 Scope Level of Local Declarations for the Sort Transmitter Distance CSU....317
4.7.2.5.2.4-1 Transmitter Pointer Array Data Structure....318
4.7.2.5.2.4-2 Sort Record Data Structure....318
4.7.3.1.1-1 Requirements Allocated to the Sort Entity State Velocity CSU....326
4.7.3.1.2.1-1 Sort Entity State Velocity CSU I/O Data....327
4.7.3.1.2.2-1 Sort Entity State Velocity CSU Local Data Descriptions....327
4.7.3.1.2.2-2 Size, Units, and Range of Local Data for the Sort Entity State Velocity CSU....328
4.7.3.1.2.2-3 Sort Entity State Velocity CSU Local Function Description....328
4.7.3.1.2.2-4 Scope Level of Local Declarations for the Sort Entity State Velocity CSU....328
4.7.3.1.2.4-1 Entity State Pointer Array Data Structure....329
4.7.3.1.2.4-2 Sort Record Data Structure....329
5-1 Global Data Element Descriptions....336
5-2 Global Data Element Definitions....337
8.3-1 Acronyms and Abbreviations....338

## 1 Scopetc "1 Scope"§

The following paragraphs will identify the CSCI, provide an overview of the system, and describe the purpose and contents of this document.

### 1.1 Identificationtc "1.1 Identification"§

This Software Design Document (SDD) describes the complete design for the CSCI identified as DIS Library (DL), CSCI 2, of the Ada Distributed Interactive Simulation (ADIS) project. The requirements from which this CSCI design was derived may be found in the DL Software Requirements Specification (SRS) and the DL Interface Requirements Specification (IRS).

### 1.2 System Overviewtc "1.2 System Overview"§

The Naval Air Warfare Center, Aircraft Division (NAWCAD) Flight Test and Engineering Group (FTEG) develops and maintains a state-of-the-art high-fidelity flight test simulation facility, the Manned Flight Simulator (MFS). This facility supports a number of Department of the Navy (DON) programs and is a key element of the Air Combat Environment Test and Evaluation Facility (ACETEF). The MFS has worked extensively with integration of a new standard in inter-simulation communications, the Distributed Interactive Simulation (DIS) standard, which allows the MFS to communicate with other simulation facilities.

DIS is a time and space coherent synthetic representation of world environments designed for linking the interactive, free play activities of people in operational exercises. The synthetic environment is created through real-time exchange of data units between distributed, computationally autonomous simulation applications in the form of simulations, simulators, and instrumented equipment interconnected through standard computer communicative services. The computational simulation entities may be present in one location or may be distributed geographically.

The basic architecture concepts of DIS are an extension of the Simulator Networking (SIMNET) program developed by Defense Advanced Research Project Agency (DARPA). The basic architecture concepts for DIS are:

1. No central computer controls the entire simulation exercise.
2. Autonomous simulation applications are responsible for maintaining the state of one or more simulation entities.
3. A standard protocol is used for communicating "ground truth" data.
4. Changes in the state of an entity are communicated by simulation applications.
5. Perception of events or other entities is determined by the receiving application.
6. Dead-reckoning algorithms are used to reduce communications processing.

In supporting the implementation of these basic concepts, the DIS Library (DL) CSCI will provide a set of data processing DIS-related routines that have a broad range of functionality such as: the capability to prioritize or refine a list or stream of Protocol Data Units (a.k.a.  PDUs, which represent sets of related simulation data describing an entity or event) that are received from the DIS network, calculating dead-reckoning and smoothing entity update positions and coordinate transformations.  Each routine in the library is independent and has a specific purpose.  The units may be used in any desired combination, however the resulting data reduction and associated processing time may vary with different usage's.

## 1.3  Document Overviewtc "1.3	Document Overview"§

The purpose of this document is to describe each of the Computer Software Components (CSCs) of the DL CSCI.  Each CSC is organized into sub-level CSCs, Computer Software Units (CSUs), or a combination of both.  This SDD provides the preliminary as well as the detailed design.

## 2 Applicable Documentstc "2 Applicable Documents"§

The following subparagraphs list the government and non-government documents which were utilized in creating the DL SDD.

### 2.1 Government Documentstc "2.1 Government Documents"§

The following documents of the exact issue shown form a part of this document to the extent specified herein. In the event of conflict between the documents referenced herein and the contents of this document, the contents of this document shall be considered a superseding requirement.

**Table 2.1-1**
**Government Documents**tc " 2.1-1 Government Documents" \f t§

| **Document Number** | **Title** |
|---|---|
| DI-MCCR-80012A | Software Design Document |
| DOD-STD-2167A | Military Standard Defense System Software Development |

Copies of specifications, standards, drawings, and publications required by suppliers in connection with specified procurement functions should be obtained from the contracting agency or as directed by the contracting officer.

### 2.2 Non-Government Documentstc "2.2 Non-Government Documents"§

The following documents of the exact issue shown form a part of this document to the extent specified herein. In the event of conflict between the documents referenced herein and the contents of this document, the contents of this document shall be considered a superseding requirement.

**Table 2.2-1**
**Non-Government Documents**tc " 2.2-1 Non-Government Documents" \f t§

| Document Number | Title | Source |
|---|---|---|
| JFT-145-1-DL.IRS-A | DL Interface Requirements Specification | J.F. Taylor, Inc. Lexington Park, MD. |
| JFT-145-1-DL.SRS-A | DL Software Requirements Specification | J.F. Taylor, Inc. Lexington Park, MD. |

## 3 Preliminary Designtc "3 Preliminary Design"§

The following paragraphs will provide an overview of the DL CSCI of the ADIS system as well as a detailed description of each CSC within the CSCI. The overview will discuss the CSCI architecture and the available system states. The overview will be followed by the CSC design descriptions.

### 3.1 DL CSCI Overviewtc "3.1 DL CSCI Overview"§

The DIS Library (DL) CSCI consists of a set of commonly used, non-trivial data processing units which perform such useful operations as coordinate system conversion, orientation conversion, dead-reckoning, calculations, filtering, smoothing, hashing and sorting. The library may be combined and linked with other software configuration items to support DIS-related processing units.

The operations of the DL CSCI are implemented via a hierarchical set of software components and units; the DL CSCI is divided into 7 subsections corresponding to each of the above operations, and each subsection is further divided into the appropriate computer software components and computer software units which comprise the individual activities. The external interfaces for the DL CSCI are summarized in Table 3.1-1. The system architecture diagram in Figure 3.1-1 shows the relationships between the DL CSCI and other CSCIs in the system.

**Table 3.1-1**
**External Interfaces For The DL CSCI**tc " 3.1-1 External Interfaces for the DL CSCI" \f t§

| Project Unique Identifier | Interface Name | IRS Reference |
|---|---|---|
| DL-EI-1 | DIS Library Interface | Para 1.2.1 |

**Figure 3.1-1**
**System Architecture : DL To Other CSCIs**tc "3.1-1 System Architecture : DL To Other CSCIs" \f f§

### 3.1.1 DL CSCI Architecturetc "3.1.1 DL CSCI Architecture"§

The hierarchy chart in Figure 3.1.1-1 describes the internal organization structure for the DL CSCI; specifically, the diagram illustrates the top-level CSCs which comprise the DIS Library. Figures 3.1.1-2 through 3.1.1-4 depicts the sub-level CSCs which are formed within the various CSCs. Figure 3.1.1-5 shows the top-level CSC-to-CSC relationships within the DL CSCI and shows the external data flow. Table 3.1.1-1 summarizes and states the purpose of each CSC and sub-level CSC within the DL CSCI. Table 3.1.1-2 defines the top-level CSC-to-CSC interfaces.

**Figure 3.1.1-1**
**DL CSCI Hierarchy Chart**tc "3.1.1-1	DL CSCI Hierarchy Chart" \f f§

**Figure 3.1.1-2**
**Conversion Algorithms CSC Hierarchy Chart**tc "3.1.1-2	Conversion Algorithms CSC Hierarchy Chart" \f f§

µ §

**Figure 3.1.1-3**
**Filtering Algorithms CSC Hierarchy Chart** tc "3.1.1-3 Filtering Algorithms CSC Hierarchy Chart" \f f§

µ §

**Figure 3.1.1-4**
**Sorting Algorithms CSC Hierarchy Chart**tc "3.1.1-4 Sorting Algorithms CSC Hierarchy Chart" \f f§

**Figure 3.1.1-5**
**DL CSCI CSC-to-CSC Relationships**tc "3.1.1-5 DL CSCI CSC-to-CSC Relationships " \f f§

**Table 3.1.1-1**
**CSCs Defined for the DL CSCI**tc " 3.1.1-1 CSCs Defined for the DL CSCI" \f t§

| CSC Name | Identifier | Purpose |
|---|---|---|
| Conversion Algorithms | DL-CSC-1 | To provide support for converting between various coordinate and orientation systems. |
| Coordinate System Conversion | DL-CSC-1.1 | To provide routines to convert between geocentric and UTM, local and entity coordinate systems. |
| Orientation System Conversion | DL-CSC-1.2 | To provide routines to convert between Euler Angles and Orientation Angles. |
| Dead-Reckoning Algorithms | DL-CSC-2 | To provide support for dead-reckoning entity position updates. |
| Calculation Algorithms | DL-CSC-3 | To produce specific quantities such as distance, azimuth, elevation, and velocity which can be used to filter PDUs or for some user-specific purpose. The calculations are performed using position and velocity data from input PDUs. |
| Filtering Algorithms | DL-CSC-4 | To accept or reject PDUs based on a comparison of data elements within the PDUs against threshold values. |
| Filtering Routines | DL-CSC-4.1 | To accept or reject PDUs based on such criteria as distance, azimuth, elevation, and velocity. |
| Filter List By Distance | DL-CSC-4.2 | To accept or reject PDUs within a list of PDUs based on distance. The position within each PDU in the list is measured from a reference position. A distance which exceeds the threshold value causes the PDU to be removed from the list |
| Filter List By Orientation | DL-CSC-4.3 | To accept or reject PDUs within a list of PDUs based on azimuth, elevation, or both. Each PDU in the list is considered relative to the entity coordinate system of a reference Entity State PDU. Those PDUs in the list which do not fall within the specified thresholds for azimuth or elevation are removed from the list. |
| Filter List By Velocity | DL-CSC-4.4 | To accept or reject PDUs within a list of PDUs based on velocity. The velocity vector within each PDU in the list is used to calculate a velocity magnitude. A magnitude which exceeds the threshold value causes the PDU to be removed from the list |
| Hashing Algorithms | DL-CSC-5 | To provide a mechanism for allocating and managing the storage and retrieval of Entity State PDU information that is required for entity position updates. |
| Smoothing Algorithms | DL-CSC-6 | To compensate for network anomalies and certain data |

| | | |
|---|---|---|
| Sorting Algorithms | DL-CSC-7 | To sort a list of PDUs into a sequence which supports some user-specific operation. |
| Sort Routines | DL-CSC-7.1 | To sort a list of PDUs based on an associated sort key (i.e., distance, velocity). |
| Sort List By Distance | DL-CSC-7.2 | To sort a list of PDUs based on the distance between the position component within each PDU and a reference position. |
| Sort List By Velocity | DL-CSC-7.3 | To sort PDUs based on the magnitude of the velocity vector within each PDU. |

**Table 3.1.1-2**
**CSC-to-CSC Interfaces for the DL CSCI**tc " 3.1.1-2 CSC-to-CSC Interfaces for the DL CSCI" \f t§

| Data Element | Description/Use |
|---|---|
| Entity Coordinate Position | Contains the location of an entity or event after rotation into an entity coordinate system. It is used to calculate azimuth and elevation; both of which operations require a coordinate transformation into the entity coordinate system of a specified DIS entity. |
| Entity Identifier | Contains the Site ID, Application ID, and Entity ID for a DIS entity. It is used to uniquely identify a DIS entity. |
| Entity State Data List Pointer | Points to a list of structures containing selected data from Entity State PDUs. The data consists of a time tag, a position (in world coordinates), and a velocity vector. It is used for the purposes of smoothing values for position and velocity. |
| Entity State PDU | See IST-CR-93-15, paragraph 5.4.3.1 for a description of Entity State PDU. It is used to hold related information regarding a DIS entity, including position, velocity, and dead-reckoning data. The Entity State PDU is used as input to the CSCs which perform smoothing and dead-reckoning. It is also used to specify the origin of an entity coordinate system. |
| Geocentric Position | Contains the position of an entity or event in world coordinates. It is used calculate distance (without a coordinate transformation), and azimuth and elevation (following a coordinate transformation). |
| PDU Calculation Data | Contains input data for the Calculation Algorithms CSC. It is used to derive such information as distance, azimuth, velocity magnitude, and elevation. |
| PDU Calculation Results | Contains results from the Calculation Algorithms CSC. It is used as input for the Filtering Routines CSC, and is also available for user-specific purposes via the DL external interface. |

### 3.1.2 System States and Modestc "3.1.2 System States and Modes"§

The DL CSCI does not operate based on specific states or modes. Therefore, this paragraph does not apply, and has been tailored out of this SDD.

### 3.1.3 Memory and Processing Time Allocationtc "3.1.3 Memory and Processing Time Allocation"§

The CSCI memory and processing time requirements have been tailored out of this SDD.

## 3.2 DL CSCI Design Descriptiontc "3.2 DL CSCI Design Description"§

The following paragraphs provide a design description for each CSC of the DL CSCI of the ADIS system.

**3.2.1 Conversion Algorithms CSC (DL-CSC-1)**tc "3.2.1 Conversion Algorithms CSC (DL-CSC-1)"§

The Conversion Algorithms CSC (DL-CSC-1) provides support for converting between various coordinate and orientation systems. The requirements of this CSC have been allocated entirely to sub-level CSCs, described in the following subparagraphs.

**3.2.1.1 Coordinate System Conversion CSC (DL-CSC-1.1)**tc "3.2.1.1 Coordinate System Conversion CSC (DL-CSC-1.1)"§

The Coordinate System Conversion CSC (DL-CSC-1.1) provides routines to convert between geocentric and either geodetic, UTM, local or entity coordinate systems. A geocentric coordinate system uses offsets along an orthogonal axis system with its origin at the center of the earth, as described in *IEEE Standard for Information Technology - Protocols for Distributed Interactive Simulation Applications, Version 2.0* (IST-CR-93-15), Section 1.2.2. A geodetic coordinate system uses latitude, longitude, and altitude relative to sea level. A UTM coordinate system uses zone, North position, and East position. A local coordinate system (for purposes of this CSCI) is one which uses North, East, and elevation offsets from a centerpoint defined in geodetic coordinates. An entity coordinate system uses offsets along an orthogonal axis system with its origin at the center of an entity, as described in IST-CR-93-15, Section 1.2.2. Figure 3.2.1.1-1 shows the control flow for this CSC, and Figure 3.2.1.1-2 shows the data flow. (In these figures, solid lines with arrows indicate data flow and dashed lines with arrows indicate control flow. Rectangles with solid borders represent CSCs internal to this CSCI, and rectangles with dashed borders indicate external CSCs and CSCIs.) There are no derived design requirements for this CSC. There are no additional design constraints imposed on or by this CSC.

**Figure 3.2.1.1-1**
**Coordinate System Conversion CSC Control Flow**tc "3.2.1.1-1 Coordinate System Conversion CSC Control Flow" \f f§

**Figure 3.2.1.1-2**

**Coordinate System Conversion CSC Data Flow**tc "3.2.1.1-2	Coordinate System Conversion CSC Data Flow" \f f§

**3.2.1.2 Orientation System Conversion CSC (DL-CSC-1.2)**tc "3.2.1.2 Orientation System Conversion CSC (DL-CSC-1.2)"§

The Orientation System Conversion CSC (DL-CSC-1.2) provides routines to convert between Euler angles and orientation angles. An Euler angle system uses psi, theta, and phi to describe a rotational transformation between a geocentric and entity coordinate system, as described in *IEEE Standard for Information Technology - Protocols for Distributed Interactive Simulation Applications, Version 2.0*, (IST-CR-93-15), Section 1.2.2. An orientation angle system uses heading, pitch, and roll to describe the orientation of an entity. Figure 3.2.1.2-1 shows the control flow for this CSC, and Figure 3.2.1.2-2 shows the data flow. (In these figures, solid lines with arrows indicate data flow and dashed lines with arrows indicate control flow. Rectangles with solid borders represent CSCs internal to this CSCI, and rectangles with dashed borders indicate external CSCs and CSCIs.) There are no derived design requirements for this CSC. There are no additional design constraints imposed on or by this CSC.

**Figure 3.2.1.2-1**
**Orientation System Conversion CSC Control Flow**tc "3.2.1.2-1 Orientation System Conversion CSC Control Flow" \f f§

µ §

**Figure 3.2.1.2-2**
**Orientation System Conversion CSC Data Flow**tc "3.2.1.2-2 Orientation System Conversion CSC Data Flow" \f f§

**3.2.2 Dead-Reckoning Algorithms CSC (DL-CSC-2)**tc "3.2.2 Dead-Reckoning Algorithms CSC (DL-CSC-2)"§

The Dead-Reckoning Algorithms CSC (DL-CSC-2) provides support for a subset of the Dead-Reckoning Models (DRMs) defined in *Enumeration and Bit Encoded Values for Use with Protocols for Distributed Interactive Simulation Applications* (IST-CR-93-19) Section 7. The supported subset consists of the following DRMs: Static, DRM(F,P,W), DRM(R,P,W), DRM(R,V,W), and DRM(F,V,W). All other algorithms are handled using the Static DRM. Figure 3.2.2-1 shows the control flow for this CSC, and Figure 3.2.2-2 shows the data flow. (In these figures, solid lines with arrows indicate data flow and dashed lines with arrows indicate control flow. Rectangles with solid borders represent CSCs internal to this CSCI, and rectangles with dashed borders indicate external CSCs and CSCIs.) There are no derived design requirements for this CSC. There are no additional design constraints imposed on or by this CSC.

µ §

**Figure 3.2.2-1**
**Dead-Reckoning Algorithms CSC Control Flow**tc "3.2.2-1 Dead-Reckoning Algorithms CSC Control Flow" \f f§

**Figure 3.2.2-2**
**Dead-Reckoning Algorithms CSC Data Flow**tc "3.2.2-2 Dead-Reckoning Algorithms CSC Data Flow" \f f§

### 3.2.3 Calculation Algorithms CSC (DL-CSC-3)tc "3.2.3 Calculation Algorithms CSC (DL-CSC-3)"§

The Calculation Algorithms CSC (DL-CSC-3) provides the capability to calculate specific numeric quantities based on components within DIS PDUs. This CSC contains the units necessary to calculate quantities for distance, azimuth, elevation and velocity. Figure 3.2.3-1 shows the control flow for this CSC, and Figure 3.2.3-2 shows the data flow. (In these figures, solid lines with arrows indicate data flow and dashed lines with arrows indicate control flow. Rectangles with solid borders represent CSCs internal to this CSCI, and rectangles with dashed borders indicate external CSCs and CSCIs.) There are no derived design requirements for this CSC. There are no additional design constraints imposed on or by this CSC.

µ §

**Figure 3.2.3-1**
**Calculation Algorithms CSC Control Flow**tc "3.2.3-1 Calculation Algorithms CSC Control Flow" \f f§

µ §

**Figure 3.2.3-2**
**Calculation Algorithms CSC Data Flow**tc "3.2.3-2 Calculation Algorithms CSC Data Flow" \f f§

## 3.2.4 Filtering Algorithms CSC (DL-CSC-4)tc "3.2.4 Filtering Algorithms CSC (DL-CSC-4)"§

The Filtering Algorithms CSC (DL-CSC-4) provides the capability to filter individual PDUs and lists of PDUs based on user-specified thresholds.  This CSC contains the sub-level CSCs which are responsible for evaluating components within individual PDUs against threshold values, and evaluating lists of PDUs.  Figure 3.2.4-1 shows the control flow for the sub-level CSC to sub-level CSC relationships, and Figure 3.2.4-2 shows the data flow.  (In these figures, solid lines with arrows indicate data flow and dashed lines with arrows indicate control flow.  Rectangles with solid borders represent CSCs internal to this CSCI, and rectangles with dashed borders indicate external CSCs and CSCIs.)  There are no derived design requirements for this CSC.  There are no additional design constraints imposed on or by this CSC.

µ §

**Figure 3.2.4-1**
**Filtering Algorithms CSC Control Flow**tc "3.2.4-1 Filtering Algorithms CSC Control Flow" \f f§

**Figure 3.2.4.-2**
**Filtering Algorithms CSC Data Flow**tc "3.2.4-2 Filtering Algorithms CSC Data Flow" \f f §

**3.2.4.1 Filtering Routines CSC (DL-CSC-4.1)**tc "3.2.4.1 Filtering Routines CSC (DL-CSC-4.1)"§

The Filtering Algorithms CSC (DL-CSC-4.1) provides the capability to evaluate the position or velocity components from a single PDU for the purposes of accepting or rejecting the PDU. This CSC contains the units necessary to evaluate PDUs based on distance, azimuth, elevation, and velocity.

**3.2.4.2 Filter List By Distance CSC (DL-CSC-4.2)**tc "3.2.4.2 Filter List By Distance CSC (DL-CSC-4.2)"§

The Filter List By Distance CSC (DL-CSC-4.2) provides the capability to filter individual PDUs within a list based on the distance between the position component within each PDU and a reference PDU. This CSC contains the units necessary to filter lists of Entity State PDUs, Fire PDUs, Detonation PDUs, Laser PDUs, and Transmitter PDUs based on distance.

**3.2.4.3 Filter List By Orientation CSC (DL-CSC-4.3)**tc "3.2.4.3 Filter List By Orientation CSC (DL-CSC-4.3)"§

The Filter List By Orientation CSC (DL-CSC-4.3) provides the capability to filter individual PDUs within a list based on the angle between the position of an entity or event with respect to a reference entity or event. This CSC contains the units necessary to filter lists of Entity State PDUs, Fire PDUs, Detonation PDUs, Laser PDUs, and Transmitter PDUs based on azimuth and/or elevation.

**3.2.4.4 Filter List By Velocity CSC (DL-CSC-4.4)**tc "3.2.4.4 Filter List By Velocity CSC (DL-CSC-4.4)"§

The Filter List By Velocity CSC (DL-CSC-4.4) provides the capability to accept or reject individual PDUs within a list of PDUs based on a comparison of the magnitude of the velocity vector within the PDU and a velocity threshold. This CSC contains the unit necessary to Filter lists of Entity State PDUs against a minimum or maximum velocity threshold

**3.2.5 Hashing Algorithms CSC (DL-CSC-5)**tc "3.2.5 Hashing Algorithms CSC (DL-CSC-5)"§

The Hashing Algorithms CSC (DL-CSC-5) provides the capability to store and retrieve previously processed Entity State PDU information. This CSC contains the units necessary to allocate and manage this storage space. Figure 3.2.5-1 shows the control flow for this CSC, and Figure 3.2.5-2 shows the data flow. (In these figures, solid lines with arrows indicate data flow and dashed lines with arrows indicate control flow. Rectangles with solid borders represent CSCs internal to this CSCI, and rectangles with dashed borders indicate external CSCs and CSCIs.) There are no derived design requirements for this CSC. There are no additional design constraints imposed on or by this CSC.

µ §

**Figure 3.2.5-1**
**Hashing Algorithms CSC Control Flow**tc "3.2.5-1 Hashing Algorithms CSC Control Flow" \f f§

µ §

**Figure 3.2.5-2**
**Hashing Algorithms CSC Data Flow**tc "3.2.5-2 Hashing Algorithms CSC Data Flow" \f f§

### 3.2.6 Smoothing Algorithms CSC (DL-CSC-6)tc "3.2.6 Smoothing Algorithms CSC (DL-CSC-6)"§

The Smoothing Algorithms CSC (DL-CSC-6) provides the capability to compensate for network anomalies and certain data errors regarding the position and velocity of entities. This CSC contains the units necessary to limit the rate of change of position or velocity, and to dampen the effect of position changes for an entity or event. Figure 3.2.6-1 shows the control flow for this CSC, and Figure 3.2.6-2 shows the data flow. (In these figures, solid lines with arrows indicate data flow and dashed lines with arrows indicate control flow. Rectangles with solid borders represent CSCs internal to this CSCI, and rectangles with dashed borders indicate external CSCs and CSCIs.) There are no derived design requirements for this CSC. There are no additional design constraints imposed on or by this CSC.

µ §

**Figure 3.2.6-1**
**Smoothing Algorithms CSC Control Flow**tc "3.2.6-1 Smoothing Algorithms CSC Control Flow" \f f§

µ §

**Figure 3.2.6-2**
**Smoothing Algorithms CSC Data Flow**tc "3.2.6-2 Smoothing Algorithms CSC Data Flow" \f f§

**3.2.7 Sorting Algorithms CSC (DL-CSC-7)**tc "3.2.7 Sorting Algorithms CSC (DL-CSC-7)"§

The Sorting Algorithms CSC (DL-CSC-7) provides the capability to sort lists of specific PDU types based on a specific criteria. This CSC contains the sub-level CSCs which are responsible for sorting PDUs based on distance and velocity.

**3.2.7.1 Sort Routines CSC (DL-CSC-7.1)**tc "3.2.7.1 Sort Routines CSC (DL-CSC-7.1)"§

The Sort Routines CSC (DL-CSC-7.1) provides the capability to sort lists of pointers to PDUs based on distance or velocity. This CSC contains the unit necessary to sort a generic array structure. Figure 3.2.7.1-1 shows the control flow for the sub-level CSC to sub-level CSC relationships, and Figure 3.2.7.1-2 shows the data flow. (In these figures, solid lines with arrows indicate data flow and dashed lines with arrows indicate control flow. Rectangles with solid borders represent CSCs internal to this CSCI, and rectangles with dashed borders indicate external CSCs and CSCIs.) There are no derived design requirements for this CSC. There are no additional design constraints imposed on or by this CSC.

µ §

**Figure 3.2.7.1-1**
**Sorting Algorithms CSC Control Flow**tc "3.2.7.1-1 Sorting Algorithms CSC Control Flow" \f f§

**Figure 3.2.7.1-2**
**Sorting Algorithms CSC Data Flow** tc "3.2.7.1-2	Sorting Algorithms Data Flow CSC" \f f §

**3.2.7.2 Sort By Distance CSC (DL-CSC-7.2)**tc "3.2.7.2 Sort By Distance CSC (DL-CSC-7.2)"§

The Sort List By Distance CSC (DL-CSC-7.2) provides the capability to sort a list of PDUs based on the distance between the position element within each PDU and a reference PDU. This CSC contains the units necessary to sort lists of Entity State PDUs, Fire PDUs, Detonation PDUs, Laser PDUs, and Transmitter PDUs based on distance.

**3.2.7.3 Sort By Velocity CSC (DL-CSC-7.3)**tc "3.2.7.3 Sort By Velocity CSC (DL-CSC-7.3)"§

The Sort List By Velocity CSC (DL-CSC-7.3) provides the capability to sort a list of PDUs based on the magnitude of the velocity vector within each PDU. This CSC contains the units necessary to sort lists of Entity State PDUs based on velocity.

# 4 Detailed Designtc "4 Detailed Design"§

## 4.1 Conversion Algorithms CSC (DL-CSC-1)tc "4.1 Conversion Algorithms CSC (DL-CSC-1)"§

The following subparagraphs identify and describe each of the sub-level CSCs of the Conversion Algorithms CSC (DL-CSC-1).

### 4.1.1 Coordinate System Conversion CSC (DL-CSC-1.1)tc "4.1.1 Coordinate System Conversion CSC (DL-CSC-1.1)"§

The following subparagraphs identify and describe each of the CSUs of the Coordinate System Conversion CSC (DL-CSC-1.1). Figure 4.1.1-1 identifies the CSUs of this CSC. Figure 4.1.1-2 shows the execution control flow relationships of the CSUs of this CSC. Figure 4.1.1-3 shows the data flow relationships of the CSUs of this CSC. (In these figures, solid lines with no arrows indicate a hierarchical relationship, solid lines with arrows indicate data flow and dashed lines with arrows indicate control flow. Rectangles with solid borders represent units internal to the CSC, and rectangles with dashed borders indicate external CSCs and CSCIs.)

**Figure 4.1.1-1**
**Coordinate System  Conversion CSC Computer Software Units**tc "4.1.1-1 Coordinate System  Conversion CSC Computer Software Units" \f f§

**Figure 4.1.1-2**
**Coordinate System  Conversion CSC Execution Control Flow**tc "4.1.1-2 Coordinate System  Conversion CSC Execution Control Flow" \f f§

**Figure 4.1.1-3**
**Coordinate System  Conversion CSC Data Flow**tc "4.1.1-3	Coordinate System Conversion CSC Data Flow" \f f§

**4.1.1.1 Geocentric-Geodetic Conversion CSU (DL-CSU-1.1.1)**tc "4.1.1.1 Geocentric-Geodetic Conversion CSU (DL-CSU-1.1.1)"§

The Geocentric-Geodetic Conversion CSU (DL-CSU-1.1.1) converts from geocentric to geodetic coordinate systems. The following subparagraphs provide the design information for this CSU.

**4.1.1.1.1 Geocentric-Geodetic Conversion CSU Design Specification/Constraints**tc "4.1.1.1.1 Geocentric-Geodetic Conversion CSU Design Specification/Constraints"§

There are no design constraints applicable to this CSU.

**4.1.1.1.2 Geocentric-Geodetic Conversion CSU Design**tc "4.1.1.1.2 Geocentric-Ge

The following subparagraphs provide the detailed design of the Geocentric-Geodetic Conversion CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.1.1.1.2.1 Geocentric-Geodetic Conversion CSU Input/Output Data Elements**tc "4.1.1.1.2.1 Geocentric-Geodetic Conversion CSU Input/Output Data Elements"§

Table 4.1.1.1.2.1-1 identifies and states the purpose of each input and output data element of the Geocentric-Geodetic Conversion CSU.

**Table 4.1.1.1.2.1-1**
**Geocentric-Geodetic Conversion CSU I/O Data**tc "4.1.1.1.2.1-1 Geocentric-Geodetic Conversion CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS Table 3.2.2-2 DL-D-26 |
| Geocentric Coordinate | Input | Provides location using offsets from center of earth. | See DL IRS Table 3.2.2-2 |
| Geodetic Coordinate | Output | Provides location using latitude, longitude, and elevation. | See DL IRS Table 3.2.2-2 |

**4.1.1.1.2.2 Geocentric-Geodetic Conversion CSU Local Data Elements**tc "4.1.1.1.2.2 Geocentric-Geodetic Conversion CSU Local Data Elements"§

There are no local data elements utilized by this CSU.

**4.1.1.1.2.3 Geocentric-Geodetic Conversion CSU Global Data Elements**tc "4.1.1.1.2.3 Geocentric-Geodetic Conversion CSU Global Data Elements"§

There are no global data elements utilized by this CSU.

**4.1.1.1.2.4 Geocentric-Geodetic Conversion CSU Local and Shared Data Structures**tc "4.1.1.1.2.4 Geocentric-Geodetic Conversion CSU Local and Shared Data Structures"§

There are no local or shared data structures utilized by this CSU.

**4.1.1.1.2.5 Geocentric-Geodetic Conversion CSU Interrupts and Signals**tc "4.1.1.1.2.5 Geocentric-Geodetic Conversion CSU Interrupts and Signals"§

There are no interrupts or signals processed by this CSU.

**4.1.1.1.2.6 Geocentric-Geodetic Conversion CSU Error Handling**tc "4.1.1.1.2.6 Geocentric-Geodetic Conversion CSU Error Handling"§

The Geocentric-Geodetic Conversion CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of GCCGDC_FAILURE to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Geocentric-Geodetic Conversion CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.1.1.1.2.7 Geocentric-Geodetic Conversion CSU Use of Other Elements**tc "4.1.1.1.2.7 Geocentric-Geodetic Conversion CSU Use of Other Elements"§

There are no other elements utilized by this CSU.

**4.1.1.1.2.8 Geocentric-Geodetic Conversion CSU Logic Flow**tc "4.1.1.1.2.8 Geocentric-Geodetic Conversion CSU Logic Flow"§

The logic flow of the Geocentric-UTM Conversion CSU is described in Figure 4.1.1.1.2.8-1.

**Figure 4.1.1.1.2.8-1**
**Geocentric-Geodetic Conversion CSU Flowchart**tc "4.1.1.1.2.8-1 Geocentric-Geodetic Conversion CSU Flowchart" \f f§

**4.1.1.1.2.9 Geocentric-Geodetic Conversion CSU Algorithms**tc "4.1.1.1.2.9 Geocentric-Geodetic Conversion CSU Algorithms"§

The software will be based on algorithms developed by IST.

**4.1.1.1.2.10 Geocentric-Geodetic Conversion CSU Limitations**tc "4.1.1.1.2.10 Geocentric-Geodetic Conversion CSU Limitations"§

There are no limitations on this CSU.

**4.1.1.2 Geocentric-UTM Conversion CSU (DL-CSU-1.1.2)**tc "4.1.1.2 Geocentric-UTM Conversion CSU (DL-CSU-1.1.2)"§

The Geocentric-UTM Conversion CSU (DL-CSU-1.1.2) converts from geocentric to UTM coordinate systems. The following subparagraphs provide the design information for this CSU.

**4.1.1.2.1 Geocentric-UTM Conversion CSU Design Specification/Constraints**tc "4.1.1.2.1 Geocentric-UTM Conversion CSU Design Specification/Constraints"§

There are no design constraints applicable to this CSU.

**4.1.1.2.2 Geocentric-UTM Conversion CSU Design**tc "4.1.1.2.2 Geocentric-UTM Conversion CSU Design"§

The following subparagraphs provide the detailed design of the Geocentric-UTM Conversion CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.1.1.2.2.1 Geocentric-UTM Conversion CSU Input/Output Data Elements**tc "4.1.1.2.2.1 Geocentric-UTM Conversion CSU Input/Output Data Elements"§

Table 4.1.1.2.2.1-1 identifies and states the purpose of each input and output data element of the Geocentric-UTM Conversion CSU.

**Table 4.1.1.2.2.1-1**
**Geocentric-UTM Conversion CSU I/O Data**tc "4.1.1.2.2.1-1 Geocentric-UTM Conversion CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS Table 3.2.2-2 DL-D-26 |
| Geocentric Coordinate | Input | Provides location using offsets from center of earth. | See DL IRS Table 3.2.2-2 |
| UTM Coordinate | Output | Provides location using zone, North offset, and East offset. | See DL IRS Table 3.2.2-2 |

**4.1.1.2.2.2 Geocentric-UTM Conversion CSU Local Data Elements**tc "4.1.1.2.2.2 Geocentric-UTM Conversion CSU Local Data Elements"§

There are no local data elements utilized by this CSU.

**4.1.1.2.2.3 Geocentric-UTM Conversion CSU Global Data Elements**tc "4.1.1.2.2.3 Geocentric-UTM Conversion CSU Global Data Elements"§

There are no global data elements utilized by this CSU.

**4.1.1.2.2.4 Geocentric-UTM Conversion CSU Local and Shared Data Structures**tc "4.1.1.2.2.4 Geocentric-UTM Conversion CSU Local and Shared Data Structures"§

There are no local or shared data structures utilized by this CSU.

**4.1.1.2.2.5 Geocentric-UTM Conversion CSU Interrupts and Signals**tc "4.1.1.2.2.5 Geocentric-UTM Conversion CSU Interrupts and Signals"§

There are no interrupts or signals processed by this CSU.

**4.1.1.2.2.6 Geocentric-UTM Conversion CSU Error Handling**tc "4.1.1.2.2.6 Geocentric-UTM Conversion CSU Error Handling"§

The Geocentric-UTM Conversion CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of GCCUTM_FAILURE to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Geocentric-UTM Conversion CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.1.1.2.2.7 Geocentric-UTM Conversion CSU Use of Other Elements**tc "4.1.1.2.2.7 Geocentric-UTM Conversion CSU Use of Other Elements"§

There are no other elements utilized by this CSU.

**4.1.1.2.2.8 Geocentric-UTM Conversion CSU Logic Flow**tc "4.1.1.2.2.8 Geocentric-UTM Conversion CSU Logic Flow"§

The logic flow of the Geocentric-UTM Conversion CSU is described in Figure 4.1.1.2.2.8-1.

**Figure 4.1.1.2.2.8-1**
**Geocentric-UTM Conversion CSU Flowchart**tc "4.1.1.2.2.8-1 Geocentric-UTM Conversion CSU Flowchart" \f f§

**4.1.1.2.2.9 Geocentric-UTM Conversion CSU Algorithms**tc "4.1.1.2.2.9 Geocentric-UTM Conversion CSU Algorithms"§

The software will be based on algorithms developed by IST.

**4.1.1.2.2.10 Geocentric-UTM Conversion CSU Limitations**tc "4.1.1.2.2.10 Geocentric-UTM Conversion CSU Limitations"§

There are no limitations on this CSU.

**4.1.1.3 Geocentric-Local Conversion CSU (DL-CSU-1.1.3)**tc "4.1.1.3 Geocentric-Local Conversion CSU (DL-CSU-1.1.3)"§

The Geocentric-Local Conversion CSU (DL-CSU-1.1.3) converts from geocentric to local coordinate systems. The following subparagraphs provide the design information for this CSU.

**4.1.1.3.1 Geocentric-Local Conversion CSU Design Specification/Constraints**tc "4.1.1.3.1 Geocentric-Local Conversion CSU Design Specification/Constraints"§

There are no design constraints applicable to this CSU.

**4.1.1.3.2 Geocentric-Local Conversion CSU Design**tc "4.1.1.3.2 Geocentric-Local Conversion CSU Design"§

The following subparagraphs provide the detailed design of the Geocentric-Local Conversion CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.1.1.3.2.1 Geocentric-Local Conversion CSU Input/Output Data Elements**tc "4.1.1.3.2.1 Geocentric-Local Conversion CSU Input/Output Data Elements"§

Table 4.1.1.3.2.1-1 identifies and states the purpose of each input and output data element of the Geocentric-Local Conversion CSU.

**Table 4.1.1.3.2.1-1**
**Geocentric-Local Conversion CSU I/O Data**tc "4.1.1.3.2.1-1 Geocentric-Local Conversion CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS Table 3.2.2-2 DL-D-26 |
| Geocentric Coordinate | Input | Provides location using offsets from center of earth. | See DL IRS Table 3.2.2-2 |
| Local Centerpoint | Input | Provides the centerpoint of the local coordinate system as a geodetic coordinate. | See DL IRS Table 3.2.2-2 |
| Local Coordinate | Output | Provides location using North, East, and vertical offsets from the local centerpoint. | See DL IRS Table 3.2.2-2 |

**4.1.1.3.2.2 Geocentric-Local Conversion CSU Local Data Elements**tc "4.1.1.3.2.2 Geocentric-Local Conversion CSU Local Data Elements"§

There are no local data elements utilized by this CSU.

**4.1.1.3.2.3 Geocentric-Local Conversion CSU Global Data Elements**tc "4.1.1.3.2.3 Geocentric-Local Conversion CSU Global Data Elements"§

There are no global data elements utilized by this CSU.

**4.1.1.3.2.4 Geocentric-Local Conversion CSU Local and Shared Data Structures**tc "4.1.1.3.2.4 Geocentric-Local Conversion CSU Local and Shared Data Structures"§

There are no local or shared data structures utilized by this CSU.

**4.1.1.3.2.5 Geocentric-Local Conversion CSU Interrupts and Signals**tc "4.1.1.3.2.5 Geocentric-Local Conversion CSU Interrupts and Signals"§

There are no interrupts or signals processed by this CSU.

**4.1.1.3.2.6 Geocentric-Local Conversion CSU Error Handling**tc "4.1.1.3.2.6 Geocentric-Local Conversion CSU Error Handling"§

The Geocentric-Local Conversion CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of GCCLOC_FAILURE to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Geocentric-Local Conversion CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.1.1.3.2.7 Geocentric-Local Conversion CSU Use of Other Elements**tc "4.1.1.3.2.7 Geocentric-Local Conversion CSU Use of Other Elements"§

There are no other elements utilized by this CSU.

**4.1.1.3.2.8 Geocentric-Local Conversion CSU Logic Flow**tc "4.1.1.3.2.8 Geocentric-Local Conversion CSU Logic Flow"§

The logic flow of the Geocentric-Local Conversion CSU is described in Figure 4.1.1.3.2.8-1.

**Figure 4.1.1.3.2.8-1**
**Geocentric-Local Conversion CSU Flowchart**tc "4.1.1.3.2.8-1	Geocentric-Local Conversion CSU Flowchart" \f f§

**4.1.1.3.2.9 Geocentric-Local Conversion CSU Algorithms**tc "4.1.1.3.2.9 Geocentric-Local Conversion CSU Algorithms"§

The software will be based on algorithms developed by IST.

**4.1.1.3.2.10 Geocentric-Local Conversion CSU Limitations**tc "4.1.1.3.2.10 Geocentric-Local Conversion CSU Limitations"§

There are no limitations on this CSU.

**4.1.1.4 Geocentric-Entity Conversion CSU (DL-CSU-1.1.4)**tc "4.1.1.4 Geocentric-Entity Conversion CSU (DL-CSU-1.1.4)"§

The Geocentric-Entity Conversion CSU (DL-CSU-1.1.4) converts from geocentric to entity coordinate systems. The following subparagraphs provide the design information for this CSU.

**4.1.1.4.1 Geocentric-Entity Conversion CSU Design Specification/Constraints**tc "4.1.1.4.1 Geocentric-Entity Conversion CSU Design Specification/Constraints"§

There are no design constraints applicable to this CSU.

**4.1.1.4.2 Geocentric-Entity Conversion CSU Design**tc "4.1.1.4.2 Geocentric-Entity Conversion CSU Design"§

The following subparagraphs provide the detailed design of the Geocentric-Entity Conversion CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.1.1.4.2.1 Geocentric-Entity Conversion CSU Input/Output Data Elements**tc "4.1.1.4.2.1 Geocentric-Entity Conversion CSU Input/Output Data Elements"§

Table 4.1.1.4.2.1-1 identifies and states the purpose of each input and output data element of the Geocentric-Entity Conversion CSU.

**Table 4.1.1.4.2.1-1**
**Geocentric-Entity Conversion CSU I/O Data**tc "4.1.1.4.2.1-1 Geocentric-Entity Conversion CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS Table 3.2.2-2 DL-D-26 |
| Euler Angles | Input | Provides orientation information in the form of a rotational transformation matrix. | See DL IRS Table 3.2.2-2 |
| Geocentric Coordinate (Entity Location) | Input | Provides location using offsets from center of earth for the entity upon which the entity coordinate system will be based. | See DL IRS Table 3.2.2-2 |
| Geocentric Coordinate (Target Location) | Input | Provides location using offsets from center of earth for the entity whose location is being converted into the first entity's coordinate system. | See DL IRS Table 3.2.2-2 |
| Entity Coordinate | Output | Provides location in coordinate system centered on an entity and oriented relative to that entity's orientation. | See DL IRS Table 3.2.2-2 |

**4.1.1.4.2.2 Geocentric-Entity Conversion CSU Local Data Elements**tc "4.1.1.4.2.2 Geocentric-Entity Conversion CSU Local Data Elements"§

There are no local data elements utilized by this CSU.

**4.1.1.4.2.3 Geocentric-Entity Conversion CSU Global Data Elements**tc "4.1.1.4.2.3 Geocentric-Entity Conversion CSU Global Data Elements"§

There are no global data elements utilized by this CSU.

**4.1.1.4.2.4 Geocentric-Entity Conversion CSU Local and Shared Data Structures**tc "4.1.1.4.2.4 Geocentric-Entity Conversion CSU Local and Shared Data Structures"§

There are no local or shared data structures utilized by this CSU.

**4.1.1.4.2.5 Geocentric-Entity Conversion CSU Interrupts and Signals**tc "4.1.1.4.2.5 Geocentric-Entity Conversion CSU Interrupts and Signals"§

There are no interrupts or signals processed by this CSU.

**4.1.1.4.2.6 Geocentric-Entity Conversion CSU Error Handling**tc "4.1.1.4.2.6 Geocentric-Entity Conversion CSU Error Handling"§

The Geocentric-Entity Conversion CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of GCCENT_FAILURE to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Geocentric-Entity Conversion CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.1.1.4.2.7 Geocentric-Entity Conversion CSU Use of Other Elements**tc "4.1.1.4.2.7 Geocentric-Entity Conversion CSU Use of Other Elements"§

There are no other elements utilized by this CSU.

**4.1.1.4.2.8 Geocentric-Entity Conversion CSU Logic Flow**tc "4.1.1.4.2.8 Geocentric-Entity Conversion CSU Logic Flow"§

The logic flow of the Geocentric-Entity Conversion CSU is described in Figure 4.1.1.4.2.8-1.

**Figure 4.1.1.4.2.8-1**
**Geocentric-Entity Conversion CSU Flowchart**tc "4.1.1.4.2.8-1 Geocentric-Entity Conversion CSU Flowchart" \f f§

**4.1.1.4.2.9 Geocentric-Entity Conversion CSU Algorithms**tc "4.1.1.4.2.9 Geocentric-Entity Conversion CSU Algorithms"§

The software will be based on algorithms developed by IST.

**4.1.1.4.2.10 Geocentric-Entity Conversion CSU Limitations**tc "4.1.1.4.2.10 Geocentric-Entity Conversion CSU Limitations"§

There are no limitations on this CSU.

**4.1.1.5 Geodetic-Geocentric Conversion CSU (DL-CSU-1.1.5)**tc "4.1.1.5 Geodetic-Geocentric Conversion CSU (DL-CSU-1.1.5)"§

The Geodetic-Geocentric Conversion CSU (DL-CSU-1.1.5) converts from geodetic to geocentric coordinate systems. The following subparagraphs provide the design information for this CSU.

**4.1.1.5.1 Geodetic-Geocentric Conversion CSU Design Specification/Constraints**tc "4.1.1.5.1 Geodetic-Geocentric Conversion CSU Design Specification/Constraints"§

There are no design constraints applicable to this CSU.

**4.1.1.5.2 Geodetic-Geocentric Conversion CSU Design**tc "4.1.1.5.2 Geodetic-Geocentric Conversion CSU Design"§

The following subparagraphs provide the detailed design of the Geodetic-Geocentric Conversion CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.1.1.5.2.1 Geodetic-Geocentric Conversion CSU Input/Output Data Elements**tc "4.1.1.5.2.1 Geodetic-Geocentric Conversion CSU Input/Output Data Elements"§

Table 4.1.1.5.2.1-1 identifies and states the purpose of each input and output data element of the Geodetic-Geocentric Conversion CSU.

**Table 4.1.1.5.2.1-1**
**Geodetic-Geocentric Conversion CSU I/O Data**tc "4.1.1.5.2.1-1 Geodetic-Geocentric Conversion CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS Table 3.2.2-2 DL-D-26 |
| Geodetic Coordinate | Input | Provides location using latitude, longitude, and elevation. | See DL IRS Table 3.2.2-2 |
| Geocentric Coordinate | Output | Provides location using offsets from center of earth. | See DL IRS Table 3.2.2-2 |

**4.1.1.5.2.2 Geodetic-Geocentric Conversion CSU Local Data Elements**tc "4.1.1.5.2.2 Geodetic-Geocentric Conversion CSU Local Data Elements"§

There are no local data elements utilized by this CSU.

**4.1.1.5.2.3 Geodetic-Geocentric Conversion CSU Global Data Elements**tc "4.1.1.5.2.3 Geodetic-Geocentric Conversion CSU Global Data Elements"§

There are no global data elements utilized by this CSU.

**4.1.1.5.2.4 Geodetic-Geocentric Conversion CSU Local and Shared Data Structures**tc "4.1.1.5.2.4 Geodetic-Geocentric Conversion CSU Local and Shared Data Structures"§

There are no local or shared data structures utilized by this CSU.

**4.1.1.5.2.5 Geodetic-Geocentric Conversion CSU Interrupts and Signals**tc "4.1.1.5.2.5 Geodetic-Geocentric Conversion CSU Interrupts and Signals"§

There are no interrupts or signals processed by this CSU.

**4.1.1.5.2.6 Geodetic-Geocentric Conversion CSU Error Handling**tc "4.1.1.5.2.6 Geodetic-Geocentric Conversion CSU Error Handling"§

The Geodetic-Geocentric Conversion CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of GDCGCC_FAILURE to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Geodetic-Geocentric Conversion CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.1.1.5.2.7 Geodetic-Geocentric Conversion CSU Use of Other Elements**tc "4.1.1.5.2.7 Geodetic-Geocentric Conversion CSU Use of Other Elements"§

There are no other elements utilized by this CSU.

**4.1.1.5.2.8 Geodetic-Geocentric Conversion CSU Logic Flow**tc "4.1.1.5.2.8 Geodetic-Geocentric Conversion CSU Logic Flow"§

The logic flow of the Geodetic-Geocentric Conversion CSU is described in Figure 4.1.1.5.2.8-1.

**Figure 4.1.1.5.2.8-1**
**Geodetic-Geocentric Conversion CSU Flowchart**tc "4.1.1.5.2.8-1	Geodetic-Geocentric Conversion CSU Flowchart" \f f§

**4.1.1.5.2.9 Geodetic-Geocentric Conversion CSU Algorithms**tc "4.1.1.5.2.9 Geodetic-Geocentric Conversion CSU Algorithms"§

The software will be based on algorithms developed by IST.

**4.1.1.5.2.10 Geodetic-Geocentric Conversion CSU Limitations**tc "4.1.1.5.2.10 Geodetic-Geocentric Conversion CSU Limitations"§

There are no limitations on this CSU.

**4.1.1.6 UTM-Geocentric Conversion CSU (DL-CSU-1.1.6)**tc "4.1.1.6 UTM-Geocentric Conversion CSU (DL-CSU-1.1.6)"§

The UTM-Geocentric Conversion CSU (DL-CSU-1.1.6) converts from UTM to geocentric coordinate systems. The following subparagraphs provide the design information for this CSU.

**4.1.1.6.1 UTM-Geocentric Conversion CSU Design Specification/Constraints**tc "4.1.1.6.1 UTM-Geocentric Conversion CSU Design Specification/Constraints"§

There are no design constraints applicable to this CSU.

**4.1.1.6.2 UTM-Geocentric Conversion CSU Design**tc "4.1.1.6.2 UTM-Geocentric Conversion CSU Design"§

The following subparagraphs provide the detailed design of the UTM-Geocentric Conversion CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.1.1.6.2.1 UTM-Geocentric Conversion CSU Input/Output Data Elements**tc "4.1.1.6.2.1 UTM-Geocentric Conversion CSU Input/Output Data Elements"§

Table 4.1.1.6.2.1-1 identifies and states the purpose of each input and output data element of the UTM-Geocentric Conversion CSU.

**Table 4.1.1.6.2.1-1**
**UTM-Geocentric Conversion CSU I/O Data**tc "4.1.1.6.2.1-1 UTM-Geocentric Conversion CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS Table 3.2.2-2 DL-D-26 |
| UTM Coordinate | Input | Provides location using zone, North offset, and East offset. | See DL IRS Table 3.2.2-2 |
| Geocentric Coordinate | Output | Provides location using offsets from center of earth. | See DL IRS Table 3.2.2-2 |

**4.1.1.6.2.2 UTM-Geocentric Conversion CSU Local Data Elements**tc "4.1.1.6.2.2 UTM-Geocentric Conversion CSU Local Data Elements"§

There are no local data elements utilized by this CSU.

**4.1.1.6.2.3 UTM-Geocentric Conversion CSU Global Data Elements**tc "4.1.1.6.2.3 UTM-Geocentric Conversion CSU Global Data Elements"§

There are no global data elements utilized by this CSU.

**4.1.1.6.2.4 UTM-Geocentric Conversion CSU Local and Shared Data Structures**tc "4.1.1.6.2.4 UTM-Geocentric Conversion CSU Local and Shared Data Structures"§

There are no local or shared data structures utilized by this CSU.

**4.1.1.6.2.5 UTM-Geocentric Conversion CSU Interrupts and Signals**tc "4.1.1.6.2.5 UTM-Geocentric Conversion CSU Interrupts and Signals"§

There are no interrupts or signals processed by this CSU.

**4.1.1.6.2.6 UTM-Geocentric Conversion CSU Error Handling**tc "4.1.1.6.2.6 UTM-Geocentric Conversion CSU Error Handling"§

The UTM-Geocentric Conversion CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of UTMGCC_FAILURE to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the UTM-Geocentric Conversion CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.1.1.6.2.7 UTM-Geocentric Conversion CSU Use of Other Elements**tc "4.1.1.6.2.7 UTM-Geocentric Conversion CSU Use of Other Elements"§

There are no other elements utilized by this CSU.

**4.1.1.6.2.8 UTM-Geocentric Conversion CSU Logic Flow**tc "4.1.1.6.2.8 UTM-Geocentric Conversion CSU Logic Flow"§

The logic flow of the UTM-Geocentric Conversion CSU is described in Figure 4.1.1.6.2.8-1.

**Figure 4.1.1.6.2.8-1**
**UTM-Geocentric Conversion CSU Flowchart**tc "4.1.1.6.2.8-1 UTM-Geocentric Conversion CSU Flowchart" \f f§

**4.1.1.6.2.9 UTM-Geocentric Conversion CSU Algorithms**tc "4.1.1.6.2.9 UTM-Geocentric Conversion CSU Algorithms"§

The software will be based on algorithms developed by IST.

**4.1.1.6.2.10 UTM-Geocentric Conversion CSU Limitations**tc "4.1.1.6.2.10 UTM-Geocentric Conversion CSU Limitations"§

There are no limitations on this CSU.

**4.1.1.7 Local-Geocentric Conversion CSU (DL-CSU-1.1.7)**tc "4.1.1.7 Local-Geocentric Conversion CSU (DL-CSU-1.1.7)"§

The Local-Geocentric Conversion CSU (DL-CSU-1.1.7) converts from local to geocentric coordinate systems. The following subparagraphs provide the design information for this CSU.

**4.1.1.7.1 Local-Geocentric Conversion CSU Design Specification/Constraints**tc "4.1.1.7.1 Local-Geocentric Conversion CSU Design Specification/Constraints"§

There are no design constraints applicable to this CSU.

**4.1.1.7.2 Local-Geocentric Conversion CSU Design**tc "4.1.1.7.2 Local-Geocentric Conversion CSU Design"§

The following subparagraphs provide the detailed design of the Local-Geocentric Conversion CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.1.1.7.2.1 Local-Geocentric Conversion CSU Input/Output Data Elements**tc "4.1.1.7.2.1 Local-Geocentric Conversion CSU Input/Output Data Elements"§

Table 4.1.1.7.2.1-1 identifies and states the purpose of each input and output data element of the Local-Geocentric Conversion CSU.

**Table 4.1.1.7.2.1-1**
**Local-Geocentric Conversion CSU I/O Data**tc "4.1.1.7.2.1-1 Local-Geocentric Conversion CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS Table 3.2.2-2 DL-D-26 |
| Local Centerpoint | Input | Provides the centerpoint of the local coordinate system as a geodetic coordinate. | See DL IRS Table 3.2.2-2 |
| Local Coordinate | Input | Provides location using North, East, and vertical offsets from the local centerpoint. | See DL IRS Table 3.2.2-2 |
| Geocentric Coordinate | Output | Provides location using offsets from center of earth. | See DL IRS Table 3.2.2-2 |

**4.1.1.7.2.2 Local-Geocentric Conversion CSU Local Data Elements**tc "4.1.1.7.2.2 Local-Geocentric Conversion CSU Local Data Elements"§

There are no local data elements utilized by this CSU.

**4.1.1.7.2.3 Local-Geocentric Conversion CSU Global Data Elements**tc "4.1.1.7.2.3 Local-Geocentric Conversion CSU Global Data Elements"§

There are no global data elements utilized by this CSU.

**4.1.1.7.2.4 Local-Geocentric Conversion CSU Local and Shared Data Structures**tc "4.1.1.7.2.4 Local-Geocentric Conversion CSU Local and Shared Data Structures"§

There are no local or shared data structures utilized by this CSU.

**4.1.1.7.2.5 Local-Geocentric Conversion CSU Interrupts and Signals**tc "4.1.1.7.2.5 Local-Geocentric Conversion CSU Interrupts and Signals"§

There are no interrupts or signals processed by this CSU.

**4.1.1.7.2.6 Local-Geocentric Conversion CSU Error Handling**tc "4.1.1.7.2.6 Local-Geocentric Conversion CSU Error Handling"§

The Local-Geocentric Conversion CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of LOCGCC_FAILURE to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Local-Geocentric Conversion CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.1.1.7.2.7 Local-Geocentric Conversion CSU Use of Other Elements**tc "4.1.1.7.2.7 Local-Geocentric Conversion CSU Use of Other Elements"§

There are no other elements utilized by this CSU.

**4.1.1.7.2.8 Local-Geocentric Conversion CSU Logic Flow**tc "4.1.1.7.2.8 Local-Geocentric Conversion CSU Logic Flow"§

The logic flow of the Local-Geocentric Conversion CSU is described in Figure 4.1.1.7.2.8-1.

**Figure 4.1.1.7.2.8-1**
**Local-Geocentric Conversion CSU Flowchart**tc "4.1.1.7.2.8-1 Local-Geocentric Conversion CSU Flowchart" \f f§

**4.1.1.7.2.9 Local-Geocentric Conversion CSU Algorithms**tc "4.1.1.7.2.9 Local-Geocentric Conversion CSU Algorithms"§

The software will be based on algorithms developed by IST.

**4.1.1.7.2.10 Local-Geocentric Conversion CSU Limitations**tc "4.1.1.7.2.10 Local-Geocentric Conversion CSU Limitations"§

There are no limitations on this CSU.

**4.1.1.8 Entity-Geocentric Conversion CSU (DL-CSU-1.1.8)**tc "4.1.1.8 Entity-Geocentric Conversion CSU (DL-CSU-1.1.8)"§

The Entity-Geocentric Conversion CSU (DL-CSU-1.1.8) converts from entity to geocentric coordinate systems. The following subparagraphs provide the design information for this CSU.

**4.1.1.8.1 Entity-Geocentric Conversion CSU Design Specification/Constraints**tc "4.1.1.8.1 Entity-Geocentric Conversion CSU Design Specification/Constraints"§

There are no design constraints applicable to this CSU.

**4.1.1.8.2 Entity-Geocentric Conversion CSU Design**tc "4.1.1.8.2 Entity-Geocentric Conversion CSU Design"§

The following subparagraphs provide the detailed design of the Entity-Geocentric Conversion CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.1.1.8.2.1 Entity-Geocentric Conversion CSU Input/Output Data Elements**tc "4.1.1.8.2.1 Entity-Geocentric Conversion CSU Input/Output Data Elements"§

Table 4.1.1.8.2.1-1 identifies and states the purpose of each input and output data element of the Entity-Geocentric Conversion CSU.

**Table 4.1.1.8.2.1-1**
**Entity-Geocentric Conversion CSU I/O Data**tc "4.1.1.8.2.1-1 Entity-Geocentric Conversion CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS Table 3.2.2-2 DL-D-26 |
| Entity Coordinate | Input | Provides location in coordinate system centered on an entity and oriented relative to that entity's orientation. | See DL IRS Table 3.2.2-2 |
| Euler Angles | Input | Provides orientation information in the form of a rotational transformation matrix. | See DL IRS Table 3.2.2-2 |
| Geocentric Coordinate (Entity Location) | Input | Provides location using offsets from center of earth for the entity upon which the entity coordinate system will be based. | See DL IRS Table 3.2.2-2 |
| Geocentric Coordinate (Target Location) | Output | Provides location using offsets from center of earth for the entity whose location is being converted into the first entity's coordinate system. | See DL IRS Table 3.2.2-2 |

**4.1.1.8.2.2 Entity-Geocentric Conversion CSU Local Data Elements**tc "4.1.1.8.2.2 Entity-Geocentric Conversion CSU Local Data Elements"§

There are no local data elements utilized by this CSU.

**4.1.1.8.2.3 Entity-Geocentric Conversion CSU Global Data Elements**tc "4.1.1.8.2.3 Entity-Geocentric Conversion CSU Global Data Elements"§

There are no global data elements utilized by this CSU.

**4.1.1.8.2.4 Entity-Geocentric Conversion CSU Local and Shared Data Structures**tc "4.1.1.8.2.4 Entity-Geocentric Conversion CSU Local and Shared Data Structures"§

There are no local or shared data structures utilized by this CSU.

**4.1.1.8.2.5 Entity-Geocentric Conversion CSU Interrupts and Signals**tc "4.1.1.8.2.5 Entity-Geocentric Conversion CSU Interrupts and Signals"§

There are no interrupts or signals processed by this CSU.

**4.1.1.8.2.6 Entity-Geocentric Conversion CSU Error Handling**tc "4.1.1.8.2.6 Entity-Geocentric Conversion CSU Error Handling"§

The Entity-Geocentric Conversion CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of ENTGCC_FAILURE to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Entity-Geocentric Conversion CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.1.1.8.2.7 Entity-Geocentric Conversion CSU Use of Other Elements**tc "4.1.1.8.2.7 Entity-Geocentric Conversion CSU Use of Other Elements"§

There are no other elements utilized by this CSU.

**4.1.1.8.2.8 Entity-Geocentric Conversion CSU Logic Flow**tc "4.1.1.8.2.8 Entity-Geocentric Conversion CSU Logic Flow"§

The logic flow of the Entity-Geocentric Conversion CSU is described in Figure 4.1.1.8.2.8-1.

**Figure 4.1.1.8.2.8-1**
**Entity-Geocentric Conversion CSU Flowchart**tc "4.1.1.8.2.8-1 Entity-Geocentric Conversion CSU Flowchart" \f f§

**4.1.1.8.2.9 Entity-Geocentric Conversion CSU Algorithms**tc "4.1.1.8.2.9 Entity-Geocentric Conversion CSU Algorithms"§

The software will be based on algorithms developed by IST.

**4.1.1.8.2.10 Entity-Geocentric Conversion CSU Limitations**tc "4.1.1.8.2.10 Entity-Geocentric Conversion CSU Limitations"§

There are no limitations on this CSU.

**4.1.2 Orientation System Conversion CSC (DL-CSC-1.2)**tc "4.1.2 Orientation System Conversion CSC (DL-CSC-1.2)"§

The following subparagraphs identify and describe each of the CSUs of the Orientation System Conversion CSC (DL-CSC-1.2). Figure 4.1.2-1 identifies the CSUs of this CSC. Figure 4.1.2-2 shows the execution control flow relationships of the CSUs of this CSC. Figure 4.1.2-3 shows the data flow relationships of the CSUs of this CSC. (In these figures, solid lines with no arrows indicate a hierarchical relationship, solid lines with arrows indicate data flow and dashed lines with arrows indicate control flow. Rectangles with solid borders represent units internal to the CSC, and rectangles with dashed borders indicate external CSCs and CSCIs.)

**Figure 4.1.2-1**
**Orientation System Conversion CSC Computer Software Units**tc "4.1.2-1 Orientation System Conversion CSC Computer Software Units" \f f§

**Figure 4.1.2-2**
**Orientation System  Conversion CSC Execution Control Flow**tc "4.1.2-2 Orientation System  Conversion CSC Execution Control Flow" \f f§

**Figure 4.1.2-3**
**Orientation System  Conversion CSC Data Flow**tc "4.1.2-3 Orientation System  Conversion CSC Data Flow" \f f§

**4.1.2.1 Euler-Orientation Conversion CSU (DL-CSU-1.1.1)**tc "4.1.2.1 Euler-Orientation Conversion CSU (DL-CSU-1.1.1)"§

The Euler-Orientation Conversion CSU (DL-CSU-1.1.1) converts from Euler to orientation systems. The following subparagraphs provide the design information for this CSU.

**4.1.2.1.1 Euler-Orientation Conversion CSU Design Specification/Constraints**tc "4.1.2.1.1 Euler-Orientation Conversion CSU Design Specification/Constraints"§

There are no design constraints applicable to this CSU.

**4.1.2.1.2 Euler-Orientation Conversion CSU Design**tc "4.1.2.1.2 Euler-Orientation Conversion CSU Design"§

The following subparagraphs provide the detailed design of the Euler-Orientation Conversion CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.1.2.1.2.1 Euler-Orientation Conversion CSU Input/Output Data Elements**tc "4.1.2.1.2.1 Euler-Orientation Conversion CSU Input/Output Data Elements"§

Table 4.1.2.1.2.1-1 identifies and states the purpose of each input and output data element of the Euler-Orientation Conversion CSU.

**Table 4.1.2.1.2.1-1**
**Euler-Orientation Conversion CSU I/O Data**tc "4.1.2.1.2.1-1 Euler-Orientation Conversion CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS Table 3.2.2-2 DL-D-26 |
| Euler Angles | Input | Provides orientation information in the form of a rotational transformation matrix. | See DL IRS Table 3.2.2-2 |
| Geocentric Coordinate | Input | Provides location using offsets from center of earth. | See DL IRS Table 3.2.2-2 |
| Orientation Angles | Output | Provides orientation information in terms of heading, pitch, and roll. | See DL IRS Table 3.2.2-2 |

**4.1.2.1.2.2 Euler-Orientation Conversion CSU Local Data Elements**tc "4.1.2.1.2.2 Euler-Orientation Conversion CSU Local Data Elements"§

There are no local data elements utilized by this CSU.

**4.1.2.1.2.3 Euler-Orientation Conversion CSU Global Data Elements**tc "4.1.2.1.2.3 Euler-Orientation Conversion CSU Global Data Elements"§

There are no global data elements utilized by this CSU.

**4.1.2.1.2.4 Euler-Orientation Conversion CSU Local and Shared Data Structures**tc "4.1.2.1.2.4 Euler-Orientation Conversion CSU Local and Shared Data Structures"§

There are no local or shared data structures utilized by this CSU.

**4.1.2.1.2.5 Euler-Orientation Conversion CSU Interrupts and Signals**tc "4.1.2.1.2.5 Euler-Orientation Conversion CSU Interrupts and Signals"§

There are no interrupts or signals processed by this CSU.

**4.1.2.1.2.6 Euler-Orientation Conversion CSU Error Handling**tc "4.1.2.1.2.6 Euler-Orientation Conversion CSU Error Handling"§

The Euler-Orientation Conversion CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of EULORI_FAILURE to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Euler-Orientation Conversion CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.1.2.1.2.7 Euler-Orientation Conversion CSU Use of Other Elements**tc "4.1.2.1.2.7 Euler-Orientation Conversion CSU Use of Other Elements"§

There are no other elements utilized by this CSU.

**4.1.2.1.2.8 Euler-Orientation Conversion CSU Logic Flow**tc "4.1.2.1.2.8 Euler-Orientation Conversion CSU Logic Flow"§

The logic flow of the Euler-Orientation Conversion CSU is described in Figure 4.1.2.1.2.8-1.

**Figure 4.1.2.1.2.8-1**
**Euler-Orientation Conversion CSU Flowchart**tc "4.1.2.1.2.8-1   Euler-Orientation Conversion CSU Flowchart" \f f§

**4.1.2.1.2.9 Euler-Orientation Conversion CSU Algorithms**tc "4.1.2.1.2.9 Euler-Orientation Conversion CSU Algorithms"§

The software will be based on algorithms developed by IST.

**4.1.2.1.2.10 Euler-Orientation Conversion CSU Limitations**tc "4.1.2.1.2.10 Euler-Orientation Conversion CSU Limitations"§

There are no limitations on this CSU.

**4.1.2.2 Orientation-Euler Conversion CSU (DL-CSU-1.1.2)**tc "4.1.2.2 Orientation-Euler Conversion CSU (DL-CSU-1.1.2)"§

The Orientation-Euler Conversion CSU (DL-CSU-1.1.2) converts from orientation to Euler systems. The following subparagraphs provide the design information for this CSU.

**4.1.2.2.1 Orientation-Euler Conversion CSU Design Specification/Constraints**tc "4.1.2.2.1 Orientation-Euler Conversion CSU Design Specification/Constraints"§

There are no design constraints applicable to this CSU.

**4.1.2.2.2 Orientation-Euler Conversion CSU Design**tc "4.1.2.2.2 Orientation-Euler Conversion CSU Design"§

The following subparagraphs provide the detailed design of the Orientation-Euler Conversion CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.1.2.2.2.1 Orientation-Euler Conversion CSU Input/Output Data Elements**tc "4.1.2.2.2.1 Orientation-Euler Conversion CSU Input/Output Data Elements"§

Table 4.1.2.2.2.1-1 identifies and states the purpose of each input and output data element of the Orientation-Euler Conversion CSU.

**Table 4.1.2.2.2.1-1**
**Orientation-Euler Conversion CSU I/O Data**tc "4.1.2.2.2.1-1 Orientation-Euler Conversion CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS Table 3.2.2-2 DL-D-26 |
| Geocentric Coordinate | Input | Provides location using offsets from center of earth. | See DL IRS Table 3.2.2-2 |
| Orientation Angles | Input | Provides orientation information in terms of heading, pitch, and roll. | See DL IRS Table 3.2.2-2 |
| Euler Angles | Output | Provides orientation information in the form of a rotational transformation matrix. | See DL IRS Table 3.2.2-2 |

**4.1.2.2.2.2 Orientation-Euler Conversion CSU Local Data Elements**tc "4.1.2.2.2.2 Orientation-Euler Conversion CSU Local Data Elements"§

There are no local data elements utilized by this CSU.

**4.1.2.2.2.3 Orientation-Euler Conversion CSU Global Data Elements**tc "4.1.2.2.2.3 Orientation-Euler Conversion CSU Global Data Elements"§

There are no global data elements utilized by this CSU.

**4.1.2.2.2.4 Orientation-Euler Conversion CSU Local and Shared Data Structures**tc "4.1.2.2.2.4 Orientation-Euler Conversion CSU Local and Shared Data Structures"§

There are no local or shared data structures utilized by this CSU.

**4.1.2.2.2.5 Orientation-Euler Conversion CSU Interrupts and Signals**tc "4.1.2.2.2.5 Orientation-Euler Conversion CSU Interrupts and Signals"§

There are no interrupts or signals processed by this CSU.

**4.1.2.2.2.6 Orientation-Euler Conversion CSU Error Handling**tc "4.1.2.2.2.6 Orientation-Euler Conversion CSU Error Handling"§

The Orientation-Euler Conversion CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of ORIEUL_FAILURE to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Orientation-Euler Conversion CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.1.2.2.2.7 Orientation-Euler Conversion CSU Use of Other Elements**tc "4.1.2.2.2.7 Orientation-Euler Conversion CSU Use of Other Elements"§

There are no other elements utilized by this CSU.

**4.1.2.2.2.8 Orientation-Euler Conversion CSU Logic Flow**tc "4.1.2.2.2.8 Orientation-Euler Conversion CSU Logic Flow"§

The logic flow of the Orientation-Euler Conversion CSU is described in Figure 4.1.2.2.2.8-1.

**Figure 4.1.2.2.2.8-1**
**Orientation-Euler Conversion CSU Flowchart**tc "4.1.2.2.2.8-1   Orientation-Euler Conversion CSU Flowchart" \f f§

**4.1.2.2.2.9 Orientation-Euler Conversion CSU Algorithms**tc "4.1.2.2.2.9 Orientation-Euler Conversion CSU Algorithms"§

The software will be based on algorithms developed by IST.

**4.1.2.2.2.10 Orientation-Euler Conversion CSU Limitations**tc "4.1.2.2.2.10 Orientation-Euler Conversion CSU Limitations"§

There are no limitations on this CSU.

**4.2 Dead-Reckoning Algorithms CSC (DL-CSC-2)**tc "4.1 Dead-Reckoning Algorithms CSC (DL-CSC-2)§

The following subparagraphs identify and describe each of the CSUs of the Dead-Reckoning Algorithms CSC (DL-CSC-2). Figure 4.2-1 identifies the CSUs of this CSC. Figure 4.2-2 shows the relationship of the CSUs in terms of execution control. Figure 4.2-3 shows the data flow relationships of the CSUs of this CSC. Figure 4.2-3 shows the data flow relationships of the CSUs of this CSC. (In these figures, solid lines with no arrows indicate a hierarchical relationship, solid lines with arrows indicate data flow and dashed lines with arrows indicate control flow. Rectangles with solid borders represent units internal to the CSC, and rectangles with dashed borders indicate external CSCs and CSCIs.)

**Figure 4.2-1**
**Dead-Reckoning Algorithms CSC Computer Software Units**tc "4.2-1 Dead-Reckoning Algorithms CSC Computer Software Units" \f f§

**Figure 4.2-2**
**Dead-Reckoning Algorithms CSC Execution Control Flow**tc "4.2-2 Dead-Reckoning Algorithms CSC Execution Control Flow" \f f§

§

**Figure 4.2-3**
**Dead-Reckoning Algorithms CSC Data Flow**tc "4.2-3 Dead-Reckoning Algorithms CSC Data Flow" \f f§

Note: *Status* is not shown but it is an output from all of the CSUs

**4.2.1 Update Dead Reckoned Position CSU (DL-CSU-2.1)**tc "4.2.1 Update Dead Reckoned Position CSU (DL-CSU-2.1)"§

The Update Dead Reckoned Position CSU (DL-CSU-2.1) updates the entity position data contained in an Entity State PDU using the Dead-Reckoning Model (DRM) and associated parameters contained in the Entity State PDU. The following subparagraphs provide the design information for this CSU.

**4.2.1.1 Update Dead Reckoned Position CSU Design Specification/Constraints**tc "4.2.1.1 Update Dead Reckoned Position CSU Design Specification/Constraints"§

There are no design constraints applicable to this CSU.

**4.2.1.2 Update Dead Reckoned Position CSU Design**tc "4.2.1.2 Update Dead Reckoned Position CSU Design"§

The following subparagraphs provide the detailed design of the Update Dead Reckoned Position CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.2.1.2.1 Update Dead Reckoned Position CSU Input/Output Data Elements**tc "4.2.1.2.1 Update Dead Reckoned Position CSU Input/Output Data Elements"§

Table 4.2.1.2.1-1 identifies and states the purpose of each input and output data element of the Update Dead Reckoned Position CSU.

**Table 4.2.1.2.1-1**
**Update Dead Reckoned Position CSU I/O Data**tc "4.2.1.2.1-1 Update Dead Reckoned Position CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS Table 3.2.2-2 DL-D-26 |
| Entity State PDU | Input/Output | Provides information about an entity, including position, orientation, linear velocity and acceleration, and angular velocity. | See DL IRS Table 3.2.2-2 |

#### 4.2.1.2.2 Update Dead Reckoned Position CSU Local Data Elementstc "4.2.1.2.2 Update Dead Reckoned Position CSU Local Data Elements"§

Table 4.2.1.2.2-1 identifies and describes each data element that originates within the Update Dead Reckoned Position CSU and which is not used by any other CSU.

**Table 4.2.1.2.2-1**
**Update Dead Reckoned Position CSU Local Data Descriptions**tc "4.2.1.2.2-1 Local Data Descriptions for the Update Dead Reckoned Position CSU" \f t§

| Name | Description | Data Type |
|---|---|---|
| Delta Time | Time elapsed since last entity update. | See Table 5-2, DL-GD-1 |

**4.2.1.2.3 Update Dead Reckoned Position CSU Global Data Elements**tc "4.2.1.2.3 Update Dead Reckoned Position CSU Global Data Elements"§

There are no global data elements utilized by this CSU.

**4.2.1.2.4 Update Dead Reckoned Position CSU Local and Shared Data Structures**tc "4.2.1.2.4 Update Dead Reckoned Position CSU Local and Shared Data Structures"§

There are no local or shared data structures utilized by this CSU.

**4.2.1.2.5 Update Dead Reckoned Position CSU Interrupts and Signals**tc "4.2.1.2.5 Update Dead Reckoned Position CSU Interrupts and Signals"§

There are no interrupts or signals processed by this CSU.

**4.2.1.2.6 Update Dead Reckoned Position CSU Error Handling**tc "4.2.1.2.6 Update Dead Reckoned Position CSU Error Handling"§

The Update Dead Reckoned Position CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of UDP_FAILURE to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Update Dead Reckoned Position CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.2.1.2.7 Update Dead Reckoned Position CSU Use of Other Elements**tc "4.2.1.2.7 Update Dead Reckoned Position CSU Use of Other Elements"§

There are no other elements utilized by this CSU.

**4.2.1.2.8 Update Dead Reckoned Position CSU Logic Flow**tc "4.2.1.2.8 Update Dead Reckoned Position CSU Logic Flow"§

The logic flow of the Update Dead Reckoned Position CSU is described in Figure 4.2.1.2.8-1.

**Figure 4.2.1.2.8-1**
**Update Dead Reckoned Position CSU Flowchart**tc "4.2.1.2.8-1 Update Dead Reckoned Position CSU Flowchart" \f f§

**4.2.1.2.9 Update Dead Reckoned Position CSU Algorithms**tc "4.2.1.2.9 Update Dead Reckoned Position CSU Algorithms"§

The Update Dead Reckoned Position CSU does not utilize any algorithms.

**4.2.1.2.10 Update Dead Reckoned Position CSU Limitations**tc "4.2.1.2.10 Update Dead Reckoned Position CSU Limitations"§

There are no limitations on this CSU.

**4.2.2 DRM Static CSU (DL-CSU-2.2)**tc "4.2.2 DRM Static CSU (DL-CSU-2.2)"§

The DRM Static CSU (DL-CSU-2.2) is a conceptual CSU used to aid in documentation and requirements traceability. Since the DRM Static algorithm is used for entities which are immobile, no positional update is performed, and therefore no code is required to implement the algorithm.

**4.2.2.1 DRM Static CSU Design Specification/Constraints**tc "4.2.2.1 DRM Static CSU Design Specification/Constraints"§

There are no design constraints applicable to this CSU.

**4.2.2.2 DRM Static CSU Design**tc "4.2.2.2 DRM Static CSU Design"§

The detailed design of the DRM Static CSU has been omitted, since no code is required to implement the algorithm.

**4.2.3 DRM FPW CSU (DL-CSU-2.3)**tc "4.2.3 DRM FPW CSU (DL-CSU-2.3)"§

The DRM FPW CSU (DL-CSU-2.3) calculates a new position for a non-rotational entity with a constant rate of position change in world coordinates. The following subparagraphs provide the design information for this CSU.

**4.2.3.1 DRM FPW CSU Design Specification/Constraints**tc "4.2.3.1 DRM FPW CSU Design Specification/Constraints"§

There are no design constraints applicable to this CSU.

**4.2.3.2 DRM FPW CSU Design**tc "4.2.3.2 DRM FPW CSU Design"§

The following subparagraphs provide the detailed design of the DRM FPW CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.2.3.2.1 DRM FPW CSU Input/Output Data Elements**tc "4.2.3.2.1 DRM FPW CSU Input/Output Data Elements"§

Table 4.2.3.2.1-1 identifies and states the purpose of each input and output data element of the DRM FPW CSU.

**Table 4.2.3.2.1-1**
**DRM FPW CSU I/O Data**tc "4.2.3.2.1-1 DRM FPW CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS Table 3.2.2-2 DL-D-26 |
| Delta Time | Input | Provides time elapsed since last update. | See Table 5-1, DL-DG-1 |
| Entity Linear Velocity | Input | Provides linear velocity of entity in geocentric (world) coordinates. | See Table 5-1, DL-DG-4 |
| Entity Location | Input/Output | Provides position of entity in geocentric (world) coordinates. | See Table 5-1, DL-DG-5 |

**4.2.3.2.2 DRM FPW CSU Local Data Elements**tc "4.2.3.2.2 DRM FPW CSU Local Data Elements"§

There are no local data elements utilized by this CSU.

**4.2.3.2.3 DRM FPW CSU Global Data Elements**tc "4.2.3.2.3 DRM FPW CSU Global Data Elements"§

There are no global data elements utilized by this CSU.

**4.2.3.2.4 DRM FPW CSU Local and Shared Data Structures**tc "4.2.3.2.4 DRM FPW CSU Local and Shared Data Structures"§

There are no local or shared data structures utilized by this CSU.

**4.2.3.2.5 DRM FPW CSU Interrupts and Signals**tc "4.2.3.2.5 DRM FPW CSU Interrupts and Signals"§

There are no interrupts or signals processed by this CSU.

**4.2.3.2.6 DRM FPW CSU Error Handling**tc "4.2.3.2.6 DRM FPW CSU Error Handling"§

The DRM FPW CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of FPW_FAILURE to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the DRM FPW CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.2.3.2.7 DRM FPW CSU Use of Other Elements**tc "4.2.3.2.7 DRM FPW CSU Use of Other Elements"§

There are no other elements utilized by this CSU.

**4.2.3.2.8 DRM FPW CSU Logic Flow**tc "4.2.3.2.8DRM FPW CSU Logic Flow"§

The logic flow of the DRM FPW CSU is described in Figure 4.2.3.2.8-1.

**Figure 4.2.3.2.8-1**
**DRM FPW CSU Flowchart**tc "4.2.3.2.8-1  DRM FPW CSU Flowchart" \f f§

**4.2.3.2.9 DRM FPW CSU Algorithms**tc "4.2.3.2.9 DRM FPW CSU Algorithms"§

The DRM FPW CSU updates the entity location based upon the entity linear velocity using the following algorithm:

Equation (a):

where:

Previous entity location (position), updated entity location (position)

Entity linear velocity

Time elapsed since last position update

**4.2.3.2.10 DRM FPW CSU Limitations**tc "4.2.3.2.10 DRM FPW CSU Limitations"§

There are no limitations on this CSU.

**4.2.4 DRM FVW CSU (DL-CSU-2.4)**tc "4.2.4 DRM FVW CSU (DL-CSU-2.4)"§

The DRM FVW CSU (DL-CSU-2.4) calculates a new position for a non-rotational entity with a constant rate of velocity change in world coordinates. The following subparagraphs provide the design information for this CSU.

**4.2.4.1 DRM FVW CSU Design Specification/Constraints**tc "4.2.4.1 DRM FVW CSU Design Specification/Constraints"§

There are no design constraints applicable to this CSU.

**4.2.4.2 DRM FVW CSU Design**tc "4.2.4.2 DRM FVW CSU Design"§

The following subparagraphs provide the detailed design of the DRM FVW CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.2.4.2.1 DRM FVW CSU Input/Output Data Elements**tc "4.2.4.2.1 DRM FVW CSU Input/Output Data Elements"§

Table 4.2.4.2.1-1 identifies and states the purpose of each input and output data element of the DRM FVW CSU.

**Table 4.2.4.2.1-1**
**DRM FVW CSU I/O Data**tc "4.2.4.2.1-1 DRM FVW CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS Table 3.2.2-2 DL-D-26 |
| Delta Time | Input | Provides time elapsed since last update. | See Table 5-1, DL-DG-1 |
| Entity Linear Acceleration | Input | Provides linear acceleration of entity in geocentric (world) coordinates. | See Table 5-1, DL-DG-3 |
| Entity Linear Velocity | Input/Output | Provides linear velocity of entity in geocentric (world) coordinates. | See Table 5-1, DL-DG-4 |
| Entity Location | Input/Output | Provides position of entity in geocentric (world) coordinates. | See Table 5-1, DL-DG-5 |

**4.2.4.2.2 DRM FVW CSU Local Data Elements**tc "4.2.4.2.2 DRM FVW CSU Local Data Elements"§

There are no local data elements utilized by this CSU.

**4.2.4.2.3 DRM FVW CSU Global Data Elements**tc "4.2.4.2.3 DRM FVW CSU Global Data Elements"§

There are no global data elements utilized by this CSU.

**4.2.4.2.4 DRM FVW CSU Local and Shared Data Structures**tc "4.2.4.2.4 DRM FVW CSU Local and Shared Data Structures"§

There are no local or shared data structures utilized by this CSU.

**4.2.4.2.5 DRM FVW CSU Interrupts and Signals**tc "4.2.4.2.5 DRM FVW CSU Interrupts and Signals"§

There are no interrupts or signals processed by this CSU.

**4.2.4.2.6 DRM FVW CSU Error Handling**tc "4.2.4.2.6 DRM FVW CSU Error Handling"§

The DRM FVW CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of FVW_FAILURE to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the DRM FVW CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.2.4.2.7 DRM FVW CSU Use of Other Elements**tc "4.2.4.2.7 DRM FVW CSU Use of Other Elements"§

There are no other elements utilized by this CSU.

**4.2.4.2.8 DRM FVW CSU Logic Flow**tc "4.2.4.2.8 DRM FVW CSU Logic Flow"§

The logic flow of the DRM FVW CSU is described in Figure 4.2.4.2.8-1.

**Figure 4.2.4.2.8-1**
**DRM FVW CSU Flowchart**tc "4.2.4.2.8-1  DRM FVW CSU Flowchart" \f f§

**4.2.4.2.9 DRM FVW CSU Algorithms**tc "4.2.4.2.9 DRM FVW CSU Algorithms"§

The DRM FVW CSU updates the entity linear velocity based upon the entity linear acceleration using the following algorithm:

Equation (a):

where:

Previous entity linear velocity, updated entity linear velocity

Entity linear acceleration

Time elapsed since last velocity update

**4.2.4.2.10 DRM FVW CSU Limitations**tc "4.2.4.2.10 DRM FVW CSU Limitations"§

There are no limitations on this CSU.

**4.2.5 DRM RPW CSU (DL-CSU-2.5)**tc "4.2.5 DRM RPW CSU (DL-CSU-2.5)"§

The DRM RPW CSU (DL-CSU-2.5) calculates a new position for a rotating entity with a constant rate of position change in world coordinates. The following subparagraphs provide the design information for this CSU.

**4.2.5.1 DRM RPW CSU Design Specification/Constraints**tc "4.2.5.1 DRM RPW CSU Design Specification/Constraints"§

There are no design constraints applicable to this CSU.

**4.2.5.2 DRM RPW CSU Design**tc "4.2.5.2 DRM RPW CSU Design"§

The following subparagraphs provide the detailed design of the DRM RPW CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.2.5.2.1 DRM RPW CSU Input/Output Data Elements**tc "4.2.5.2.1 DRM RPW CSU Input/Output Data Elements"§

Table 4.2.5.2.1-1 identifies and states the purpose of each input and output data element of the DRM RPW CSU.

**Table 4.2.5.2.1-1**
**DRM RPW CSU I/O Data**tc "4.2.5.2.1-1 DRM RPW CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS Table 3.2.2-2 DL-D-26 |
| Delta Time | Input | Provides time elapsed since last update. | See Table 5-1, DL-DG-1 |
| Entity Angular Velocity | Input | Provides rate at which an entity's orientation is changing, using the entity's own coordinate system. | See Table 5-1, DL-DG-2 |
| Entity Linear Velocity | Input | Provides linear velocity of entity in geocentric (world) coordinates. | See Table 5-1, DL-DG-4 |
| Entity Location | Input/Output | Provides position of entity in geocentric (world) coordinates. | See Table 5-1, DL-DG-5 |
| Entity Orientation | Input/Output | Provides orientation of entity using Euler angles (which form a rotational transformation matrix). | See Table 5-1, DL-DG-6 |

**4.2.5.2.2 DRM RPW CSU Local Data Elements**tc "4.2.5.2.2 DRM RPW CSU Local Data Elements"§

There are no local data elements utilized by this CSU.

**4.2.5.2.3 DRM RPW CSU Global Data Elements**tc "4.2.5.2.3 DRM RPW CSU Global Data Elements"§

There are no global data elements utilized by this CSU.

**4.2.5.2.4 DRM RPW CSU Local and Shared Data Structures**tc "4.2.5.2.4 DRM RPW CSU Local and Shared Data Structures"§

There are no local or shared data structures utilized by this CSU.

**4.2.5.2.5 DRM RPW CSU Interrupts and Signals**tc "4.2.5.2.5 DRM RPW CSU Interrupts and Signals"§

There are no interrupts or signals processed by this CSU.

**4.2.5.2.6 DRM RPW CSU Error Handling**tc "4.2.5.2.6 DRM RPW CSU Error Handling"§

The DRM RPW CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of RPW_FAILURE to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the DRM RPW CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.2.5.2.7 DRM RPW CSU Use of Other Elements**tc "4.2.5.2.7 DRM RPW CSU Use of Other Elements"§

There are no other elements utilized by this CSU.

**4.2.5.2.8 DRM RPW CSU Logic Flow**tc "4.2.5.2.8 DRM RPW CSU Logic Flow"§

The logic flow of the DRM RPW CSU is described in Figure 4.2.5.2.8-1.

**Figure 4.2.5.2.8-1**
**DRM RPW CSU Flowchart**tc "4.2.5.2.8-1 DRM RPW CSU Flowchart" \f f§

**4.2.5.2.9 DRM RPW CSU Algorithms**tc "4.2.5.2.9 DRM RPW CSU Algorithms"§

The DRM RPW CSU updates the entity orientation based upon the entity angular velocity using the following algorithm:

Equation (a):

where:

Previous Euler angles

Updated Euler angles

Angular velocities

Time elapsed since last orientation update

**4.2.5.2.10 DRM RPW CSU Limitations**tc "4.2.5.2.10 DRM RPW CSU Limitations"§

There are no limitations on this CSU.

**4.2.6 DRM RVW CSU (DL-CSU-2.6)**tc "4.2.6 DRM RVW CSU (DL-CSU-2.6)"§

The DRM RVW CSU (DL-CSU-2.6) calculates a new position for a rotating entity with a constant rate of velocity change in world coordinates. The following subparagraphs provide the design information for this CSU.

**4.2.6.1 DRM RVW CSU Design Specification/Constraints**tc "4.2.6.1 DRM RVW CSU Design Specification/Constraints"§

There are no design constraints applicable to this CSU.

**4.2.6.2 DRM RVW CSU Design**tc "4.2.6.2 DRM RVW CSU Design"§

The following subparagraphs provide the detailed design of the DRM RVW CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.2.6.2.1 DRM RVW CSU Input/Output Data Elements**tc "4.2.6.2.1 DRM RVW CSU Input/Output Data Elements"§

Table 4.2.6.2.1-1 identifies and states the purpose of each input and output data element of the DRM RVW CSU.

**Table 4.2.6.2.1-1**
**DRM RVW CSU I/O Data**tc "4.2.6.2.1-1    DRM RVW CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS Table 3.2.2-2 DL-D-26 |
| Delta Time | Input | Provides time elapsed since last update. | See Table 5-1, DL-DG-1 |
| Entity Angular Velocity | Input | Provides rate at which an entity's orientation is changing, using the entity's own coordinate system. | See Table 5-1, DL-DG-2 |
| Entity Linear Acceleration | Input | Provides linear acceleration of entity in geocentric (world) coordinates. | See Table 5-1, DL-DG-3 |
| Entity Location | Input/Output | Provides position of entity in geocentric (world) coordinates. | See Table 5-1, DL-DG-5 |
| Entity Orientation | Input/Output | Provides orientation of entity using Euler angles (which form a rotational transformation matrix). | See Table 5-1, DL-DG-6 |
| Entity Linear Velocity | Input/Output | Provides linear velocity of entity in geocentric (world) coordinates. | See Table 5-1, DL-DG-4 |

**4.2.6.2.2 DRM RVW CSU Local Data Elements**tc "4.2.6.2.2 DRM RVW CSU Local Data Elements"§

There are no local data elements utilized by this CSU.

**4.2.6.2.3 DRM RVW CSU Global Data Elements**tc "4.2.6.2.3 DRM RVW CSU Global Data Elements"§

There are no global data elements utilized by this CSU.

**4.2.6.2.4 DRM RVW CSU Local and Shared Data Structures**tc "4.2.6.2.4 DRM RVW CSU Local and Shared Data Structures"§

There are no local or shared data structures utilized by this CSU.

**4.2.6.2.5 DRM RVW CSU Interrupts and Signals**tc "4.2.6.2.5 DRM RVW CSU Interrupts and Signals"§

There are no interrupts or signals processed by this CSU.

**4.2.6.2.6 DRM RVW CSU Error Handling**tc "4.2.6.2.6 DRM RVW CSU Error Handling"§

The DRM RVW CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of RVW_FAILURE to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the DRM RVW CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.2.6.2.7 DRM RVW CSU Use of Other Elements**tc "4.2.6.2.7 DRM RVW CSU Use of Other Elements"§

There are no other elements utilized by this CSU.

**4.2.6.2.8 DRM RVW CSU Logic Flow**tc "4.2.6.2.8 DRM RVW CSU Logic Flow"§

The logic flow of the DRM RVW CSU is described in Figure 4.2.6.2.8-1.

**Figure 4.2.6.2.8-1**
**DRM RVW CSU Flowchart**tc "4.2.6.2.8-1 DRM RVW CSU Flowchart" \f f§

**4.2.6.2.9  DRM RVW CSU Algorithms**tc "4.2.6.2.9	DRM RVW CSU Algorithms"§

The DRM RVW CSU updates the entity orientation using the same algorithm for updating the entity orientation as the DRM RPW CSU (see paragraph 4.2.5.2.9).

**4.2.6.2.10  DRM RVW CSU Limitations**tc "4.2.6.2.10	DRM RVW CSU Limitations"§

There are no limitations on this CSU.

**4.3  Calculation Algorithms CSC (DL-CSC-3)**tc "4.3	Calculation Algorithms CSC (DL-CSC-3)"§

The following subparagraphs identify and describe each of the CSUs of the Calculation Algorithms CSC (DL-CSC-3).  Figure 4.3-1 identifies the CSUs of this CSC.  Figure 4.3-2 shows the execution control flow relationships of the CSUs of this CSC.  Figure 4.3-3 shows the data flow relationships of the CSUs of this CSC.  (In these figures, solid lines with no arrows indicate a hierarchical relationship, solid lines with arrows indicate data flow, and dashed lines with arrows indicate control flow.  Rectangles with solid borders represent units internal to the CSC, and rectangles with dashed borders or small square boxes indicate external CSCs and CSCIs.)

µ §

**Figure 4.3-1**
**Calculation Algorithms CSC Computer Software Units**tc "4.3-1Calculation Algorithms CSC Computer Software Units" \f f§

µ §

**Figure 4.3-2**
**Calculation Algorithms CSC Execution Control**tc "4.3-2 Calculation Algorithms CSC Execution Control" \f f§

µ §

**Figure 4.3-3**
**Calculation Algorithms CSC Data Flow**tc "4.3-3 Calculation Algorithms CSC Data Flow" \f f §

### 4.3.1 Calculate Distance CSU (DL-CSU-3.1)tc "4.3.1 Calculate Distance CSU (DL-CSU-3.1)"§

The Calculate Distance CSU (DL-CSU-3.1) determines the distance between two positions, each of which represent entities or events in the same orthogonal coordinate system. The following subparagraphs provide the design information for this CSU.

#### 4.3.1.1 Calculate Distance CSU Design Specification/Constraintstc "4.3.1.1 Calculate Distance CSU Design Specification/Constraints"§

Table 4.3.1.1-1 identifies the requirements that are satisfied or partially satisfied by the Calculate Distance CSU. There are no design constraints for this CSU.

**Table 4.3.1.1-1**
**Requirements Allocated to the Calculate Distance CSU**tc " 4.3.1.1-1 Requirements Allocated to the Calculate Distance CSU" \f t§

| Requirement ID | DL SRS Paragraph |
|---|---|
| DL-R-6 | 3.2.3.1 |

### 4.3.1.2 Calculate Distance CSU Designtc "4.3.1.2 Calculate Distance CSU Design"§

The following subparagraphs specify the design of the Calculate Distance CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

#### 4.3.1.2.1 Calculate Distance CSU Input/Output Data Elementstc "4.3.1.2.1 Calculate Distance CSU Input/Output Data Elements"§

Table 4.3.1.2.1-1 identifies and states the purpose of each input and output data element to the Calculate Distance CSU.

**Table 4.3.1.2.1-1**
**Calculate Distance CSU I/O Data**tc " 4.3.1.2.1-1 Calculate Distance CSU I/O" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Position 1 | Input | Holds the position of an entity or event. | See DL IRS, Table 3.2.2-2, DL-D-03 |
| Position 2 | Input | Holds the position of a different entity or event. | See DL IRS, Table 3.2.2-2, DL-D-03 |
| Distance | Output | Represents the distance between the two input positions. | See DL IRS, Table 3.2.2-2, DL-D-06 |

**4.3.1.2.2 Calculate Distance CSU Local Data Elements**tc "4.3.1.2.2 Calculate Distance CSU Local Data Elements"§

The Calculate Distance CSU does not use any local data elements.

**4.3.1.2.3 Calculate Distance CSU Global Data Elements**tc "4.3.1.2.3 Calculate Distance CSU Global Data Elements"§

The Calculate Distance CSU does not utilize any global data elements.

**4.3.1.2.4 Calculate Distance CSU Local and Shared Data Structures**tc "4.3.1.2.4 Calculate Distance CSU Local and Shared Data Structures"§

The Calculate Distance CSU does not implement any local or shared data structures.

**4.3.1.2.5 Calculate Distance CSU Interrupts and Signals**tc "4.3.1.2.5 Calculate Distance CSU Interrupts and Signals"§

The Calculate Distance CSU does not handle any interrupts or signals.

**4.3.1.2.6 Calculate Distance CSU Error Handling**tc "4.3.1.2.6 Calculate Distance CSU Error Handling"§

The Calculate Distance CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of CALC_DIST to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS.

**4.3.1.2.7 Calculate Distance CSU Use of Other Elements**tc "4.3.1.2.7 Calculate Distance CSU Use of Other Elements"§

The Calculate Distance CSU does not use system service routines, global data files, or other global elements.

**4.3.1.2.8 Calculate Distance CSU Logic Flow**tc "4.3.1.2.8 Calculate Distance CSU Logic Flow"§

The logic flow of the Calculate Distance CSU is represented graphically by a flowchart in Figure 4.3.1.2.8-1. Execution of this CSU is initiated by the Filtering Routines CSC (DL-CSC-4.1) and by external CSCIs. This CSU does not initiate any other units.

**Figure 4.3.1.2.8-1**
**Calculate Distance CSU Flowchart**tc "4.3.1.2.8-1 Calculate Distance CSU Flowchart" \f f§

**4.3.1.2.9 Calculate Distance CSU Algorithms**tc "4.3.1.2.9 Calculate Distance CSU Algorithms"§

The Calculate Distance CSU utilizes the distance formula to determine the distance between two input positions. The equation is as follows:

Equation (a):

µ §

where:

| | |
|---|---|
| *d* | is the distance between the two positions |
| µ § | represent the x, y, z components of one position |
| µ § | represent the x, y, z components of a different position |

**4.3.1.2.10 Calculate Distance CSU Limitations**tc "4.3.1.2.10 Calculate Distance CSU Limitations"§

There are no limitations or unusual features in the Calculate Distance CSU.

**4.3.2 Calculate Azimuth CSU (DL-CSU-3.2)**tc "4.3.2 Calculate Azimuth CSU (DL-CSU-3.2)"§

The Calculate Azimuth CSU (DL-CSU-3.2) determines the azimuth of an entity or event with respect to an entity. The result will be an angle measuring between 0.0 and ($360 - 2^{-23}$) degrees. The following subparagraphs provide the design information for this CSU.

**4.3.2.1 Calculate Azimuth CSU Design Specification/Constraints**tc "4.3.2.1 Calculate Azimuth CSU Design Specification/Constraints"§

Table 4.3.2.1-1 identifies the requirements that are satisfied or partially satisfied by the Calculate Azimuth CSU. There are no design constraints for this CSU.

**Table 4.3.2.1-1**
**Requirements Allocated to the Calculate Azimuth CSU**tc " 4.3.2.1-1 Requirements Allocated to the Calculate Azimuth CSU" \f t§

| Requirement ID | DL SRS Paragraph |
|---|---|
| DL-R-7 | 3.2.3.2 |

### 4.3.2.2 Calculate Azimuth CSU Designtc "4.3.2.2 Calculate Azimuth CSU Design"§

The following subparagraphs specify the design of the Calculate Azimuth CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

#### 4.3.2.2.1 Calculate Azimuth CSU Input/Output Data Elementstc "4.3.2.2.1 Calculate Azimuth CSU Input/Output Data Elements"§

Table 4.3.2.2.1-1 identifies and states the purpose of each input and output data element to the Calculate Azimuth CSU.

**Table 4.3.2.2.1-1**
**Calculate Azimuth CSU I/O Data**tc " 4.3.2.2.1-1 Calculate Azimuth CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Entity State PDU | Input | Describes the entity which lies at the origin of the entity coordinate system. | See DL IRS, Table 3.2.2-2, DL-D-01 |
| Position Of Interest | Input | Holds a geocentric position which is translated into the entity coordinate system of the entity described by the Entity State PDU. | See DL IRS, Table 3.2.2-2, DL-D-03 |
| Azimuth | Output | Represents the relative angle of the position of interest to the heading of the entity described by the Entity State PDU. | See DL IRS, Table 3.2.2-2, DL-D-09 |

#### 4.3.2.2.2 Calculate Azimuth CSU Local Data Elementstc "4.3.2.2.2 Calculate Azimuth CSU Local Data Elements"§

Table 4.3.2.2.2-1 identifies and describes each data element that originates within the Calculate Azimuth CSU and which is not used by any other CSU.

**Table 4.3.2.2.2-1**
**Calculate Azimuth CSU Local Data Descriptions**tc " 4.3.2.2.2-1 Calculate Azimuth CSU Local Data Descriptions" \f t§

| Name | Description | Data Type |
|---|---|---|
| Transformed Position | The element which will be used to hold the results of the coordinate transformation between the input reference position and the entity coordinate system. | See DL IRS Table 3.2.2-2, DL-D-05 |

**4.3.2.2.3 Calculate Azimuth CSU Global Data Elements**tc "4.3.2.2.3 Calculate Azimuth CSU Global Data Elements"§

The Calculate Azimuth CSU does not utilize any global data elements.

**4.3.2.2.4 Calculate Azimuth CSU Local and Shared Data Structures**tc "4.3.2.2.4 Calculate Azimuth CSU Local and Shared Data Structures"§

The Calculate Azimuth CSU does not implement any local or shared data structures.

**4.3.2.2.5 Calculate Azimuth CSU Interrupts and Signals**tc "4.3.2.2.5 Calculate Azimuth CSU Interrupts and Signals"§

The Calculate Azimuth CSU does not handle any interrupts or signals.

**4.3.2.2.6 Calculate Azimuth CSU Error Handling**tc "4.3.2.2.6 Calculate Azimuth CSU Error Handling"§

The Calculate Azimuth CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of CALC_AZ to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Calculate Azimuth CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.3.2.2.7 Calculate Azimuth CSU Use of Other Elements**tc "4.3.2.2.7 Calculate Azimuth CSU Use of Other Elements"§

The Calculate Azimuth CSU does not use system service routines, global data files, or other global elements.

**4.3.2.2.8 Calculate Azimuth CSU Logic Flow**tc "4.3.2.2.8 Calculate Azimuth CSU Logic Flow"§

The logic flow of the Calculate Azimuth CSU is represented graphically by a flowchart in Figure 4.3.2.2.8-1. Execution of this CSU is initiated by the Filtering Routines CSC (DL-CSC-4.1) and by external CSCIs. This CSU initiates the Geocentric-Entity Conversion CSU (DL-CSU-1.1.6)

µ §

**Figure 4.3.2.2.8-1**
**Calculate Azimuth CSU Flowchart**tc "4.3.2.2.8-1 Calculate Azimuth CSU Flowchart" \f f§

**4.3.2.2.9 Calculate Azimuth CSU Algorithms**tc "4.3.2.2.9 Calculate Azimuth CSU Algorithms"§

The following text describes, in steps, the operations required to calculate the azimuth of an object. Figure 4.3.2.2.9-1 illustrates an example azimuth calculation:

**<u>Step (A)</u>**

The Calculate Azimuth CSU determines the relative angle of the reference position to the heading of the entity using the following formula:

µ §

where:

| | |
|---|---|
| azimuth | is the relative angle between the heading of the input entity and the position of interest. |
| y | is the value along the y-axis of the position of interest after being transformed into the entity coordinate system |
| x | is the value along the x-axis of the position of interest after being transformed into the entity coordinate system |

**<u>Step (B)</u>**

Make the following adjustments to bearing:

if x is negative then add 180.0

if x is positive and y is negative then add 360.0

otherwise add nothing

µ §

**Figure 4.3.2.2.9-1**
**Example Azimuth Calculation**tc "4.3.2.2.9-1 Example Azimuth Calculation" \f f§

**4.3.2.2.10 Calculate Azimuth CSU Limitations**tc "4.3.2.2.10 Calculate Azimuth CSU Limitations"§

There are no limitations or unusual features in the Calculate Azimuth CSU.

**4.3.3 Calculate Velocity CSU (DL-CSU-3.3)**tc "4.3.3 Calculate Velocity CSU (DL-CSU-3.3)"§

The Calculate Velocity CSU (DL-CSU-3.3) determines the magnitude of an input velocity vector. The following subparagraphs provide the design information for this CSU.

**4.3.3.1 Calculate Velocity CSU Design Specification/Constraints**tc "4.3.3.1 Calculate Velocity CSU Design Specification/Constraints"§

Table 4.3.3.1-1 identifies the requirements that are satisfied or partially satisfied by the Calculate Velocity CSU. There are no design constraints for this CSU.

**Table 4.3.3.1-1**
**Requirements Allocated to the Calculate Velocity CSU**tc " 4.3.3.1-1 Requirements Allocated to the Calculate Velocity CSU" \f t§

| **Requirement ID** | **DL SRS Paragraph** |
|---|---|
| DL-R-8 | 3.2.3.3 |

**4.3.3.2 Calculate Velocity CSU Design**tc "4.3.3.2 Calculate Velocity CSU Design"§

The following subparagraphs specify the design of the Calculate Velocity CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.3.3.2.1 Calculate Velocity CSU Input/Output Data Elements**tc "4.3.3.2.1 Calculate Velocity CSU Input/Output Data Elements"§

Table 4.3.3.2.1-1 identifies and states the purpose of each input and output data element to the Calculate Velocity CSU.

**Table 4.3.3.2.1-1**
**Calculate Velocity CSU I/O Data**tc " 4.3.3.2.1-1 Calculate Velocity CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Velocity Vector | Input | Specifies the velocity of an entity or event in terms of the x,y,z components. | See DL IRS, Table 3.2.2-2, DL-D-04 |
| Velocity Magnitude | Output | Represents the velocity magnitude of the entity or event. | See DL IRS, Table 3.2.2-2, DL-D-07 |

**4.3.3.2.2 Calculate Velocity CSU Local Data Elements**tc "4.3.3.2.2 Calculate Velocity CSU Local Data Elements"§

The Calculate Velocity CSU does not utilize any local data elements.

**4.3.3.2.3 Calculate Velocity CSU Global Data Elements**tc "4.3.3.2.3 Calculate Velocity CSU Global Data Elements"§

The Calculate Velocity CSU does not utilize any global data elements.

**4.3.3.2.4 Calculate Velocity CSU Local and Shared Data Structures**tc "4.3.3.2.4 Calculate Velocity CSU Local and Shared Data Structures"§

The Calculate Velocity CSU does not implement any local or shared data structures.

**4.3.3.2.5 Calculate Velocity CSU Interrupts and Signals**tc "4.3.3.2.5 Calculate Velocity CSU Interrupts and Signals"§

The Calculate Velocity CSU does not handle any interrupts or signals.

**4.3.3.2.6 Calculate Velocity CSU Error Handling**tc "4.3.3.2.6 Calculate Velocity CSU Error Handling"§

The Calculate Velocity CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of CALC_VEL to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS.

**4.3.3.2.7 Calculate Velocity CSU Use of Other Elements**tc "4.3.3.2.7 Calculate Velocity CSU Use of Other Elements"§

The Calculate Velocity CSU does not use system service routines, global data files, or other global elements.

**4.3.3.2.8 Calculate Velocity CSU Logic Flow**tc "4.3.3.2.8 Calculate Velocity CSU Logic Flow"§

The logic flow of the Calculate Velocity CSU is represented graphically by a flowchart in Figure 4.3.3.2.8-1. Execution of this CSU is initiated by the Filtering Routines CSC (DL-CSC-4.1) and by external CSCIs. This CSU does not initiate any other units.

**Figure 4.3.3.2.8-1**
**Calculate Velocity CSU Flowchart**tc "4.3.3.2.8-1 Calculate Velocity CSU Flowchart" \f f§

**4.3.3.2.9 Calculate Velocity CSU Algorithms**tc "4.3.3.2.9 Calculate Velocity CSU Algorithms"§

The Calculate Velocity CSU computes the magnitude of a velocity vector given the x,y, and z components. The equation is as follows:

Equation (a)
µ §

where:

µ § is the magnitude of the velocity vector

µ § represent the x, y, z components of the velocity vector

**4.3.3.2.10 Calculate Velocity CSU Limitations**tc "4.3.3.2.10 Calculate Velocity CSU Limitations"§

There are no limitations or unusual features in the Calculate Velocity CSU.

**4.3.4 Calculate Elevation CSU (DL-CSU-3.4)**tc "4.3.4 Calculate Elevation CSU (DL-CSU-3.4)"§

The Calculate Elevation CSU (DL-CSU-3.4) determines the elevation of an entity or event with respect to an entity. The result will be an angle measuring between (-180.0 + $2^{-23}$) to 180.0 degrees. The following subparagraphs provide the design information for this CSU.

**4.3.4.1 Calculate Elevation CSU Design Specification/Constraints**tc "4.3.4.1 Calculate Elevation CSU Design Specification/Constraints"§

Table 4.3.4.1-1 identifies the requirements that are satisfied or partially satisfied by the Calculate Elevation CSU. There are no design constraints for this CSU.

**Table 4.3.4.1-1**
**Requirements Allocated to the Calculate Elevation CSU**tc " 4.3.4.1-1 Requirements Allocated to the Calculate Elevation CSU" \f t§

| **Requirement ID** | **DL SRS Paragraph** |
|---|---|
| DL-R-9 | 3.2.3.4 |

**4.3.4.2 Calculate Elevation CSU Design**tc "4.3.4.2 Calculate Elevation CSU Design"§

The following subparagraphs specify the design of the Calculate Elevation CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.3.4.2.1 Calculate Elevation CSU Input/Output Data Elements**tc "4.3.4.2.1 Calculate Elevation CSU Input/Output Data Elements"§

Table 4.3.4.2.1-1 identifies and states the purpose of each input and output data element to the Calculate Elevation CSU.

**Table 4.3.4.2.1-1**
**Calculate Elevation CSU I/O Data**tc " 4.3.4.2.1-1 Calculate Elevation CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Entity State PDU | Input | Describes the entity which lies at the origin of the entity coordinate system. | See DL IRS, Table 3.2.2-2, DL-D-01 |
| Position Of Interest | Input | Holds a geocentric position which is translated into the entity coordinate system of the entity described by the Entity State PDU. | See DL IRS, Table 3.2.2-2, DL-D-03 |
| Elevation | Output | Represents the relative angle of the position of interest to the pitch of the entity described by the Entity State PDU. | See DL IRS, Table 3.2.2-2, DL-D-10 |

**4.3.4.2.2 Calculate Elevation CSU Local Data Elements**tc "4.3.4.2.2 Calculate Elevation CSU Local Data Elements"§

Table 4.3.4.2.2-1 identifies and describes each data element that originates within the Calculate Elevation CSU and which is not used by any other CSU.

**Table 4.3.4.2.2-1**
**Calculate Elevation CSU Local Data Descriptions**tc " 4.3.4.2.2-1 Calculate Elevation CSU Local Data Descriptions" \f t§

| Name | Description | Data Type |
|---|---|---|
| Transformed Position | The element which will be used to hold the results of the coordinate transformation between the input reference position and the entity coordinate system. | See DL IRS Table 3.2.2-2, DL-D-05 |

**4.3.4.2.3 Calculate Elevation CSU Global Data Elements**tc "4.3.4.2.3 Calculate Elevation CSU Global Data Elements"§

The Calculate Elevation CSU does not utilize any global data elements.

**4.3.4.2.4 Calculate Elevation CSU Local and Shared Data Structures**tc "4.3.4.2.4 Calculate Elevation CSU Local and Shared Data Structures"§

The Calculate Elevation CSU does not implement any local or shared data structures.

**4.3.4.2.5 Calculate Elevation CSU Interrupts and Signals**tc "4.3.4.2.5 Calculate Elevation CSU Interrupts and Signals"§

The Calculate Elevation CSU does not handle any interrupts or signals.

**4.3.4.2.6 Calculate Elevation CSU Error Handling**tc "4.3.4.2.6 Calculate Elevation CSU Error Handling"§

The Calculate Elevation CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of CALC_EL to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Calculate Elevation CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.3.4.2.7  Calculate Elevation CSU Use of Other Elements**tc "4.3.4.2.7	Calculate Elevation CSU Use of Other Elements"§

The Calculate Elevation CSU does not use system service routines, global data files, or other global elements.

**4.3.4.2.8  Calculate Elevation CSU Logic Flow**tc "4.3.4.2.8	Calculate Elevation CSU Logic Flow"§

The logic flow of the Calculate Elevation CSU is represented graphically by a flowchart in Figure 4.3.4.2.8-1.  Execution of this CSU is initiated by the Filtering Routines CSC (DL-CSC-4.1) and by external CSCIs.  This CSU initiates the Geocentric-Entity Conversion CSU (DL-CSU-1.1.6).

µ §

**Figure 4.3.4.2.8-1**
**Calculate Elevation CSU Flowchart**tc "4.3.4.2.8-1Calculate Elevation CSU Flowchart" \f f§

**4.3.4.2.9 Calculate Elevation CSU Algorithms**tc "4.3.4.2.9 Calculate Elevation CSU Algorithms"§

The following text describes, in steps, the operations required to calculate the Elevation of an object. Figure 4.3.4.2.9-1 illustrates an example Elevation calculation:

**Step (A)**

The Calculate Elevation CSU determines the relative angle of the reference position to the pitch of the entity using the following formula:

µ §

where:

| | |
|---|---|
| Elevation | is the relative angle between the pitch of the input entity and the position of interest. |
| z | is the value along the z-axis of the position of interest after being transformed into the entity coordinate system |
| x | is the value along the x-axis of the position of interest after being transformed into the entity coordinate system |

**Step (B)**

Make the following adjustments to elevation:

| | | | |
|---|---|---|---|
| if | x is positive | then | reverse the sign of elevation |

otherwise,

| | | | | | |
|---|---|---|---|---|---|
| if | x is negative | and | z is positive | then | subtract 90.0 |
| | x is negative | and | z is negative | then | add 90.0 |

µ §

**Figure 4.3.4.2.9-1**
**Example Elevation Calculation**tc "4.3.4.2.9-1 Example Elevation Calculation" \f f§

**4.3.4.2.10 Calculate Elevation CSU Limitations**tc "4.3.4.2.10 Calculate Elevation CSU Limitations"§

There are no limitations or unusual features in the Calculate Elevation CSU.

**4.3.5 Calculate Azimuth and Elevation CSU (DL-CSU-3.5)**tc "4.3.5 Calculate Azimuth and Elevation CSU (DL-CSU-3.5)"§

The Calculate Azimuth and Elevation CSU (DL-CSU-3.5) determines the azimuth and elevation of an entity or event with respect to an entity. The result will be two angles measuring between 0.0 and (360 - $2^{-23}$) degrees. The following subparagraphs provide the design information for this CSU.

**4.3.5.1 Calculate Azimuth and Elevation CSU Design Specification/Constraints**tc "4.3.5.1 Calculate Azimuth and Elevation CSU Design Specification/Constraints"§

Table 4.3.5.1-1 identifies the requirements that are satisfied or partially satisfied by the Calculate Azimuth and Elevation CSU. There are no design constraints for this CSU.

**Table 4.3.5.1-1**
**Requirements Allocated to the Calculate Azimuth and Elevation CSU**tc " 4.3.5.1-1 Requirements Allocated to the Calculate Azimuth and Elevation CSU" \f t§

| Requirement ID | DL SRS Paragraph |
|---|---|
| DL-R-10 | 3.2.3.5 |

**4.3.5.2 Calculate Azimuth and Elevation CSU Design**tc "4.3.5.2Calculate Azimuth and Elevation CSU Design"§

The following subparagraphs specify the design of the Calculate Azimuth and Elevation CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.3.5.2.1 Calculate Azimuth and Elevation CSU Input/Output Data Elements**tc "4.3.5.2.1 Calculate Azimuth and Elevation CSU Input/Output Data Elements"§

Table 4.3.5.2.1-1 identifies and states the purpose of each input and output data element to the Calculate Azimuth and Elevation CSU.

**Table 4.3.5.2.1-1**
**Calculate Azimuth and Elevation CSU I/O Data**tc " 4.3.5.2.1-1 Calculate Azimuth and Elevation CSU I/O Data" \f t§

| **Data Element** | **Input/Output** | **Purpose** | **Data Type** |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Entity State PDU | Input | Describes the entity which lies at the origin of the entity coordinate system. | See DL IRS, Table 3.2.2-2, DL-D-01 |
| Position Of Interest | Input | Holds a geocentric position which is translated into the entity coordinate system of the entity described by the Entity State PDU. | See DL IRS, Table 3.2.2-2, DL-D-03 |
| Azimuth | Output | Represents the relative angle of the position of interest to the heading of the entity described by the Entity State PDU. | See DL IRS, Table 3.2.2-2, DL-D-09 |
| Elevation | Output | Represents the relative angle of the position of interest to the pitch of the entity described by the Entity State PDU. | See DL IRS, Table 3.2.2-2, DL-D-10 |

### 4.3.5.2.2 Calculate Azimuth and Elevation CSU Local Data Elementstc "4.3.5.2.2 Calculate Azimuth and Elevation CSU Local Data Elements"§

Table 4.3.5.2.2-1 identifies and describes each data element that originates within the Calculate Azimuth and Elevation CSU and which is not used by any other CSU.

**Table 4.3.5.2.2-1**
**Calculate Azimuth and Elevation CSU Local Data Descriptions**tc " 4.3.5.2.2-1 Local Data Descriptions for the Calculate Azimuth and Elevation CSU" \f t§

| Name | Description | Data Type |
|---|---|---|
| Transformed Position | The element which will be used to hold the results of the coordinate transformation between the input reference position and the entity coordinate system. | See DL IRS Table 3.2.2-2, DL-D-05 |

**4.3.5.2.3  Calculate Azimuth and Elevation CSU Global Data Elements**tc "4.3.5.2.3 Calculate Azimuth and Elevation CSU Global Data Elements"§

The Calculate Azimuth and Elevation CSU does not utilize any global data elements.

**4.3.5.2.4  Calculate Azimuth and Elevation CSU Local and Shared Data Structures**tc "4.3.5.2.4 Calculate Azimuth and Elevation CSU Local and Shared Data Structures"§

The Calculate Azimuth and Elevation CSU does not implement any local or shared data structures.

**4.3.5.2.5  Calculate Azimuth and Elevation CSU Interrupts and Signals**tc "4.3.5.2.5 Calculate Azimuth and Elevation CSU Interrupts and Signals"§

The Calculate Azimuth and Elevation CSU does not handle any interrupts or signals.

**4.3.5.2.6  Calculate Azimuth and Elevation CSU Error Handling**tc "4.3.5.2.6 Calculate Azimuth and Elevation CSU Error Handling"§

The Calculate Azimuth and Elevation CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of  CALC_AZ_AND_EL to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Calculate Azimuth and Elevation CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.3.5.2.7 Calculate Azimuth and Elevation CSU Use of Other Elements**tc "4.3.5.2.7 Calculate Azimuth and Elevation CSU Use of Other Elements"§

The Calculate Azimuth and Elevation CSU does not use system service routines, global data files, or other global elements.

**4.3.5.2.8 Calculate Azimuth and Elevation CSU Logic Flow**tc "4.3.5.2.8 Calculate Azimuth and Elevation CSU Logic Flow"§

The logic flow of the Calculate Azimuth and Elevation CSU is represented graphically by a flowchart in Figure 4.3.5.2.8-1. Execution of this CSU is initiated by the Filtering Routines CSC (DL-CSC-4.1) and by external CSCIs. This CSU initiates the Geocentric-Entity Conversion CSU (DL-CSU-1.1.6)

µ §

**Figure 4.3.5.2.8-1**
**Calculate Azimuth and Elevation CSU Flowchart**tc "4.3.5.2.8-1 Calculate Azimuth and Elevation CSU Flowchart" \f f§

**4.3.5.2.9 Calculate Azimuth and Elevation CSU Algorithms**tc "4.3.5.2.9 Calculate Azimuth and Elevation CSU Algorithms"§

The Calculate Azimuth and Elevation CSU performs algorithms which have been previously defined for other units. For the algorithm used in calculating azimuth, refer to paragraph 4.3.2.2.9. For the algorithm used in calculating elevation, refer to paragraph 4.3.4.2.9. For an example on calculating azimuth, refer to Figure 4.3.2.2.9-1. For an example on calculating elevation, refer to Figure 4.3.4.2.9-1.

**4.3.5.2.10 Calculate Azimuth and Elevation CSU Limitations**tc "4.3.5.2.10 Calculate Azimuth and Elevation CSU Limitations"§

There are no limitations or unusual features in the Calculate Azimuth and Elevation CSU.

**4.4 Filtering Algorithms CSC (DL-CSC-4)**tc "4.4 Filtering Algorithms CSC (DL-CSC-4)"§

The Filtering Algorithms CSC (DL-CSC-4) consists of five sub-level CSCs: the Filtering Routines CSC (DL-CSC-4.1), the Filter List By Distance CSC (DL-CSC-4.2), the Filter List By Orientation CSC (DL-CSC-4.3), and the Filter List By Velocity CSC (DL-CSC-4.4). The following subparagraphs describe the design for each sub-level CSC.

**4.4.1 Filtering Routines CSC (DL-CSC-4.1)**tc "4.4.1 Filtering Routines CSC (DL-CSC-4.1)"§

The following subparagraphs identify and describe each of the CSUs of the Filtering Routines CSC (DL-CSC-4.1). Figure 4.4.1-1 identifies the CSUs of this CSC. Figure 4.4.1-2 shows the execution control flow relationships of the CSUs of this CSC. Figure 4.4.1-3 shows the data flow relationships of the CSUs of this CSC. (In these figures, solid lines with no arrows indicate a hierarchical relationship, solid lines with arrows indicate data flow, and dashed lines with arrows indicate control flow. Rectangles with solid borders represent units internal to the CSC, and rectangles with dashed borders or small square boxes indicate external CSCs and CSCIs.)

µ §

**Figure 4.4.1-1**
**Filtering Routines CSC Computer Software Units**tc "4.4.1-1 Filtering Routines CSC Computer Software Units" \f f§

µ §

**Figure 4.4.1-2**

**Filtering Routines CSC Execution Control**tc "4.4.1-2 Filtering Routines CSC Execution Control" \f f§

**Figure 4.4.1-3**
**Filtering Routines CSC Data Flow**tc "4.4.1-3	Filtering Routines CSC Data Flow" \f f §

**4.4.1.1  Filter Distance CSU**tc "4.4.1.1	Filter Distance CSU"§

The Filter Distance CSU (DL-CSU-4.1.1) determines whether the distance between two entities or events is within the specified maximum distance threshold value.  The following subparagraphs provide the design information for this CSU.

**4.4.1.1.1  Filter Distance CSU Design Specification/Constraints**tc "4.4.1.1.1	Filter Distance CSU Design Specification/Constraints"§

Table 4.4.1.1.1-1 identifies the requirements that are satisfied or partially satisfied by the Filter Distance CSU.  There are no design constraints for this CSU.

**Table 4.4.1.1.1-1**
**Requirements Allocated to the Filter Distance CSU**tc " 4.4.1.1.1-1	Requirements Allocated to the Filter Distance CSU" \f t§

| **Requirement ID** | **DL SRS Paragraph** |
|---|---|
| DL-R-10 | 3.2.4.1.1 |

**4.4.1.1.2 Filter Distance CSU Design**tc "4.4.1.1.2 Filter Distance CSU Design"§

The following subparagraphs specify the design of the Filter Distance CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.4.1.1.2.1 Filter Distance CSU Input/Output Data Elements**tc "4.4.1.1.2.1 Filter Distance CSU Input/Output Data Elements"§

Table 4.4.1.1.2.1-1 identifies and states the purpose of each input and output data element to the Filter Distance CSU.

**Table 4.4.1.1.2.1-1**
**Filter Distance CSU I/O Data**tc " 4.4.1.1.2.1-1 Filter Distance CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Position 1 | Input | Describes the position of an entity or event in terms of the x,y,z components. | See DL IRS, Table 3.2.2-2, DL-D-03 |
| Position 2 | Input | Describes the position of a different entity or event in terms of the x,y,z components. | See DL IRS, Table 3.2.2-2, DL-D-03 |
| Distance Threshold | Input | Specifies the maximum acceptable distance between the positions. | See DL IRS, Table 3.2.2-2, DL-D-06 |
| Evaluation | Output | Indicates whether the distance between the positions is within the threshold or beyond the threshold. | See DL IRS, Table 3.2.2-2, DL-D-08 |

**4.4.1.1.2.2 Filter Distance CSU Local Data Elements**tc "4.3.2.1.2.2 Filter Distance CSU Local Data Elements"§

The Filter Distance CSU does not utilize any local data elements.

**4.4.1.1.2.3 Filter Distance CSU Global Data Elements**tc "4.4.1.1.2.3 Filter Distance CSU Global Data Elements"§

The Filter Distance CSU does not utilize any global data elements.

**4.4.1.1.2.4 Filter Distance CSU Local and Shared Data Structures**tc "4.4.1.1.2.4 Filter Distance CSU Local and Shared Data Structures"§

The Filter Distance CSU does not implement any local or shared data structures.

**4.4.1.1.2.5 Filter Distance CSU Interrupts and Signals**tc "4.4.1.1.2.5 Filter Distance CSU Interrupts and Signals"§

The Filter Distance CSU does not handle any interrupts or signals.

**4.4.1.1.2.6 Filter Distance CSU Error Handling**tc "4.4.1.1.2.6 Filter Distance CSU Error Handling"§

The Filter Distance CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of FILT_DIST to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Filter Distance CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.4.1.1.2.7 Filter Distance CSU Use of Other Elements**tc "4.4.1.1.2.7 Filter Distance CSU Use of Other Elements"§

The Filter Distance CSU does not use system service routines, global data files, or other global elements.

**4.4.1.1.2.8 Filter Distance CSU Logic Flow**tc "4.4.1.1.2.8Filter Distance CSU Logic Flow"§

The logic flow of the Filter Distance CSU is represented graphically by a flowchart in Figures 4.4.1.1.2.8-1 and 4.4.1.1.2.8-2. Execution of this CSU is initiated by the Filter List By Distance CSC (DL-CSC-4.2) and by external CSCIs. This CSU initiates the Calculate Distance CSU (DL-CSU-3.1).

µ §

**Figure 4.4.1.1.2.8-1**
**Filter Distance CSU Flowchart (Part 1)**tc "4.4.1.1.2.8-1 Filter Distance CSU Flowchart (Part 1)" \f f§

µ §

**Figure 4.4.1.1.2.8-2**
**Filter Distance CSU Flowchart (Part 2)**tc "4.4.1.1.2.8-2 Filter Distance CSU Flowchart (Part 2)" \f f§

**4.4.1.1.2.9 Filter Distance CSU Algorithms**tc "4.4.1.1.2.9 Filter Distance CSU Algorithms"§

The Filter Distance CSU does not utilize any algorithms.

**4.4.1.1.2.10 Filter Distance CSU Limitations**tc "4.4.1.1.2.10 Filter Distance CSU Limitations"§

There are no limitations or unusual features in the Filter Distance CSU.

**4.4.1.2 Filter Azimuth CSU**tc "4.4.1.2 Filter Azimuth CSU"§

The Filter Azimuth CSU (DL-CSU-4.1.2) determines whether the azimuth of an entity or event with respect to a reference entity is within the specified threshold range. The azimuth is considered to be within the range if it lies between the first and second threshold angles. The following subparagraphs provide the design information for this CSU.

**4.4.1.2.1 Filter Azimuth CSU Design Specification/Constraints**tc "4.4.1.2.1 Filter Azimuth CSU Design Specification/Constraints"§

Table 4.4.1.2.1-1 identifies the requirements that are satisfied or partially satisfied by the Filter Azimuth CSU. There are no design constraints for this CSU.

**Table 4.4.1.2.1-1**
**Requirements Allocated to the Filter Azimuth CSU**tc " 4.4.1.2.1-1 Requirements Allocated to the Filter Azimuth CSU" \f t§

| Requirement ID | DL SRS Paragraph |
|---|---|
| DL-R-11 | 3.2.4.1.2 |

#### 4.4.1.2.2 Filter Azimuth CSU Designtc "4.4.1.2.2 Filter Azimuth CSU Design"§

The following subparagraphs specify the design of the Filter Azimuth CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

##### 4.4.1.2.2.1 Filter Azimuth CSU Input/Output Data Elementstc "4.4.1.2.2.1 Filter Azimuth CSU Input/Output Data Elements"§

Table 4.4.1.2.2.1-1 identifies and states the purpose of each input and output data element to the Filter Azimuth CSU.

**Table 4.4.1.2.2.1-1**
**Filter Azimuth CSU I/O Data**tc " 4.4.1.2.2.1-1 Filter Azimuth CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Entity State PDU | Input | Describes the entity which lies at the origin of the entity coordinate system. | See DL IRS, Table 3.2.2-2, DL-D-01 |
| Position Of Interest | Input | Describes the geocentric position of a different entity or event in terms of the x,y,z components. | See DL IRS, Table 3.2.2-2, DL-D-03 |
| First Azimuth Threshold | Input | Specifies the first value in the azimuth threshold range. | See DL IRS, Table 3.2.2-2, DL-D-09 |
| Second Azimuth Threshold | Input | Specifies the second value in the azimuth threshold range. | See DL IRS, Table 3.2.2-2, DL-D-09 |
| Evaluation | Output | Indicates whether the position is at an angle that is within the specified threshold range, with respect to the entity described by the Entity State PDU. | See DL IRS, Table 3.2.2-2, DL-D-08 |

**4.4.1.2.2.2 Filter Azimuth CSU Local Data Elements**tc "4.4.1.2.2.2 Filter Azimuth CSU Local Data Elements"§

The Filter Azimuth CSU does not utilize any local data elements.

**4.4.1.2.2.3 Filter Azimuth CSU Global Data Elements**tc "4.4.1.2.2.3 Filter Azimuth CSU Global Data Elements"§

The Filter Azimuth CSU does not utilize any global data elements.

**4.4.1.2.2.4 Filter Azimuth CSU Local and Shared Data Structures**tc "4.4.1.2.2.4 Filter Azimuth CSU Local and Shared Data Structures"§

The Filter Azimuth CSU does not implement any local or shared data structures.

**4.4.1.2.2.5 Filter Azimuth CSU Interrupts and Signals**tc "4.4.1.2.2.5 Filter Azimuth CSU Interrupts and Signals"§

The Filter Azimuth CSU does not handle any interrupts or signals.

**4.4.1.2.2.6 Filter Azimuth CSU Error Handling**tc "4.4.1.2.2.6 Filter Azimuth CSU Error Handling"§

The Filter Azimuth CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of FILT_AZ to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Filter Azimuth CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.4.1.2.2.7 Filter Azimuth CSU Use of Other Elements**tc "4.4.1.2.2.7 Filter Azimuth CSU Use of Other Elements"§

The Filter Azimuth CSU does not use system service routines, global data files, or other global elements.

**4.4.1.2.2.8 Filter Azimuth CSU Logic Flow**tc "4.4.1.2.2.8Filter Azimuth CSU Logic Flow"§

The logic flow of the Filter Azimuth CSU is represented graphically by a flowchart in Figures 4.4.1.2.2.8-1 and 4.4.1.2.2.8-2. Execution of this CSU is initiated by the Filter List By Orientation CSC (DL-CSC-4.3) and by external CSCIs. This CSU initiates the Calculate Azimuth CSU (DL-CSU-3.2).

µ §

**Figure 4.4.1.2.2.8-1**
**Filter Azimuth CSU Flowchart (Part 1)**tc "4.4.1.2.2.8-1 Filter Azimuth CSU Flowchart (Part 1)" \f f§

µ §

**Figure 4.4.1.2.2.8-2**
**Filter Azimuth CSU Flowchart (Part 2)**tc "4.4.1.2.2.8-2 Filter Azimuth CSU Flowchart (Part 2)" \f f§

**4.4.1.2.2.9 Filter Azimuth CSU Algorithms**tc "4.4.1.2.2.9 Filter Azimuth CSU Algorithms"§

The Filter Azimuth CSU requires multiple comparisons to determine whether a position of interest is within the specified threshold range. The comparisons can be represented as a single block of conditionals which are stated in the following test:

**Test**

```
if (First Azimuth Threshold <= Second Azimuth Threshold) then
  if ((Calculated Azimuth >= First Azimuth Threshold) and
     (Calculated Azimuth <= Second Azimuth Threshold)) then
       Test Result is TRUE
  else
       Test Result is FALSE
  end if
else -- (First Azimuth Threshold > Second Azimuth Threshold)
  if ((Calculated Azimuth >= First Azimuth Threshold) or
     (Calculated Azimuth <= Second Azimuth Threshold)) then
       Test Result is TRUE
  else
       Test Result is FALSE
  end if
end if
```

**4.4.1.2.2.10 Filter Azimuth CSU Limitations**tc "4.4.1.2.2.10 Filter Azimuth CSU Limitations"§

There are no limitations or unusual features in the Filter Azimuth CSU.

**4.4.1.3 Filter Minimum Velocity CSU**tc "4.4.1.3 Filter Minimum Velocity CSU"§

The Filter Minimum Velocity CSU (DL-CSU-4.1.3) determines whether the velocity magnitude of an entity or event is greater than or equal to the minimum velocity magnitude threshold value. The following subparagraphs provide the design information for this CSU.

**4.4.1.3.1 Filter Minimum Velocity CSU Design Specification/Constraints**tc "4.4.1.3.1 Filter Minimum Velocity CSU Design Specification/Constraints"§

Table 4.4.1.3.1-1 identifies the requirements that are satisfied or partially satisfied by the Filter Minimum Velocity CSU. There are no design constraints for this CSU.

**Table 4.4.1.3.1-1**
**Requirements Allocated to the Filter Minimum Velocity CSU**tc " 4.4.1.3.1-1 Requirements Allocated to the Filter Minimum Velocity CSU" \f t§

| **Requirement ID** | **DL SRS Paragraph** |
|---|---|
| DL-R-12 | 3.2.4.1.3 |

**4.4.1.3.2 Filter Minimum Velocity CSU Design**tc "4.4.1.3.2 Filter Minimum Velocity CSU Design"§

The following subparagraphs specify the design of the Filter Minimum Velocity CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.4.1.3.2.1 Filter Minimum Velocity CSU Input/Output Data Elements**tc "4.4.1.3.2.1 Filter Minimum Velocity CSU Input/Output Data Elements"§

Table 4.4.1.3.2.1-1 identifies and states the purpose of each input and output data element to the Filter Minimum Velocity CSU.

**Table 4.4.1.3.2.1-1**
**Filter Minimum Velocity CSU I/O Data**tc " 4.4.1.3.2.1-1 Filter Minimum Velocity CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Velocity Vector | Input | Describes the velocity of an entity or event in terms of the x,y,z components. | See DL IRS, Table 3.2.2-2, DL-D-04 |
| Velocity Magnitude Threshold | Input | Specifies the minimum acceptable velocity magnitude. | See DL IRS, Table 3.2.2-2, DL-D-07 |
| Evaluation | Output | Indicates whether the velocity magnitude is within the threshold or beyond the threshold. | See DL IRS, Table 3.2.2-2, DL-D-08 |

**4.4.1.3.2.2 Filter Minimum Velocity CSU Local Data Elements**tc "4.4.1.3.2.2 Filter Minimum Velocity CSU Local Data Elements"§

The Filter Minimum Velocity CSU does not utilize any local data elements.

**4.4.1.3.2.3 Filter Minimum Velocity CSU Global Data Elements**tc "4.4.1.3.2.3Filter Minimum Velocity CSU Global Data Elements"§

The Filter Minimum Velocity CSU does not utilize any global data elements.

**4.4.1.3.2.4 Filter Minimum Velocity CSU Local and Shared Data Structures**tc "4.4.1.3.2.4 Filter Minimum Velocity CSU Local and Shared Data Structures"§

The Filter Minimum Velocity CSU does not implement any local or shared data structures.

**4.4.1.3.2.5 Filter Minimum Velocity CSU Interrupts and Signals**tc "4.4.1.3.2.5 Filter Minimum Velocity CSU Interrupts and Signals"§

The Filter Minimum Velocity CSU does not handle any interrupts or signals.

**4.4.1.3.2.6 Filter Minimum Velocity CSU Error Handling**tc "4.4.1.3.2.6 Filter Minimum Velocity CSU Error Handling"§

The Filter Minimum Velocity CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of FILT_MIN_VEL to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Filter Minimum Velocity CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.4.1.3.2.7 Filter Minimum Velocity CSU Use of Other Elements**tc "4.4.1.3.2.7 Filter Minimum Velocity CSU Use of Other Elements"§

The Filter Minimum Velocity CSU does not use system service routines, global data files, or other global elements.

**4.4.1.3.2.8 Filter Minimum Velocity CSU Logic Flow**tc "4.4.1.3.2.8 Filter Minimum Velocity CSU Logic Flow"§

The logic flow of the Filter Minimum Velocity CSU is represented graphically by a flowchart in Figures 4.4.1.3.2.8-1 and 4.4.1.3.2.8-2. Execution of this CSU is initiated by the Filter List By Velocity CSC (DL-CSC-4.4) and by external CSCIs. This CSU initiates the Calculate Velocity CSU (DL-CSU-3.3).

µ §

**Figure 4.4.1.3.2.8-1**
**Filter Minimum Velocity CSU Flowchart (Part 1)**tc "4.4.1.3.2.8-1 Filter Minimum Velocity CSU Flowchart (Part 1)" \f f§

µ §

**Figure 4.4.1.3.2.8-2**
**Filter Minimum Velocity CSU Flowchart (Part 2)**tc "4.4.1.3.2.8-2 Filter Minimum Velocity CSU Flowchart (Part 2)" \f f§

#### 4.4.1.3.2.9 Filter Minimum Velocity CSU Algorithmstc "4.4.1.3.2.9 Filter Minimum Velocity CSU Algorithms"§

The Filter Minimum Velocity CSU does not utilize any algorithms.

#### 4.4.1.3.2.10 Filter Minimum Velocity CSU Limitationstc "4.4.1.3.2.10 Filter Minimum Velocity CSU Limitations"§

There are no limitations or unusual features in the Filter Minimum Velocity CSU.

#### 4.4.1.4 Filter Maximum Velocity CSUtc "4.4.1.4 Filter Maximum Velocity CSU"§

The Filter Maximum Velocity CSU (DL-CSU-4.1.4) determines whether the velocity magnitude of an entity or event is less than or equal to the maximum velocity magnitude threshold value. The following subparagraphs provide the design information for this CSU.

#### 4.4.1.4.1 Filter Maximum Velocity CSU Design Specification/Constraintstc "4.4.1.4.1 Filter Maximum Velocity CSU Design Specification/Constraints"§

Table 4.4.1.4.1-1 identifies the requirements that are satisfied or partially satisfied by the Filter Maximum Velocity CSU. There are no design constraints for this CSU.

**Table 4.4.1.4.1-1**
**Requirements Allocated to the Filter Maximum Velocity CSU**tc " 4.4.1.4.1-1 Requirements Allocated to the Filter Maximum Velocity CSU" \f t§

| Requirement ID | DL SRS Paragraph |
|---|---|
| DL-R-13 | 3.2.4.1.4 |

**4.4.1.4.2 Filter Maximum Velocity CSU Design**tc "4.4.1.4.2 Filter Maximum Velocity CSU Design"§

The following subparagraphs specify the design of the Filter Maximum Velocity CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.4.1.4.2.1 Filter Maximum Velocity CSU Input/Output Data Elements**tc "4.4.1.4.2.1 Filter Maximum Velocity CSU Input/Output Data Elements"§

Table 4.4.1.4.2.1-1 identifies and states the purpose of each input and output data element to the Filter Maximum Velocity CSU.

**Table 4.4.1.4.2.1-1**
**Filter Maximum Velocity CSU I/O Data**tc " 4.4.1.4.2.1-1 Filter Maximum Velocity CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Velocity Vector | Input | Describes the velocity of an entity or event in terms of the x,y,z components. | See DL IRS, Table 3.2.2-2, DL-D-04 |
| Velocity Threshold Magnitude | Input | Specifies the maximum acceptable velocity magnitude. | See DL IRS, Table 3.2.2-2, DL-D-07 |
| Evaluation | Output | Indicates whether the velocity is within the threshold or beyond the threshold. | See DL IRS, Table 3.2.2-2, DL-D-08 |

**4.4.1.4.2.2 Filter Maximum Velocity CSU Local Data Elements**tc "4.4.1.4.2.2 Filter Maximum Velocity CSU Local Data Elements"§

The Filter Maximum Velocity CSU does not utilize any local data elements.

**4.4.1.4.2.3 Filter Maximum Velocity CSU Global Data Elements**tc "4.4.1.4.2.3 Filter Maximum Velocity CSU Global Data Elements"§

The Filter Maximum Velocity CSU does not utilize any global data elements.

**4.4.1.4.2.4 Filter Maximum Velocity CSU Local and Shared Data Structures**tc "4.4.1.4.2.4 Filter Maximum Velocity CSU Local and Shared Data Structures"§

The Filter Maximum Velocity CSU does not implement any local or shared data structures.

**4.4.1.4.2.5 Filter Maximum Velocity CSU Interrupts and Signals**tc "4.4.1.4.2.5 Filter Maximum Velocity CSU Interrupts and Signals"§

The Filter Maximum Velocity CSU does not handle any interrupts or signals.

**4.4.1.4.2.6 Filter Maximum Velocity CSU Error Handling**tc "4.4.1.4.2.6 Filter Maximum Velocity CSU Error Handling"§

The Filter Maximum Velocity CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of FILT_MAX_VEL to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Filter Maximum Velocity CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.4.1.4.2.7 Filter Maximum Velocity CSU Use of Other Elements**tc "4.4.1.4.2.7 Filter Maximum Velocity CSU Use of Other Elements"§

The Filter Maximum Velocity CSU does not use system service routines, global data files, or other global elements.

**4.4.1.4.2.8 Filter Maximum Velocity CSU Logic Flow**tc "4.4.1.4.2.8 Filter Maximum Velocity CSU Logic Flow"§

The logic flow of the Filter Maximum Velocity CSU is represented graphically by a flowchart in Figures 4.4.1.4.2.8-1 and 4.4.1.4.2.8-2. Execution of this CSU is initiated by the Filter List By Velocity CSC (DL-CSC-4.4) and by external CSCIs. This CSU initiates the Calculate Velocity CSU (DL-CSU-3.3).

µ §

**Figure 4.4.1.4.2.8-1**
**Filter Maximum Velocity CSU Flowchart (Part 1)**tc "4.4.1.4.2.8-1 Filter Maximum Velocity CSU Flowchart (Part 1)" \f f§

µ §

**Figure 4.4.1.4.2.8-2**
**Filter Maximum Velocity CSU Flowchart (Part 2)**tc "4.4.1.4.2.8-2 Filter Maximum Velocity CSU Flowchart (Part 2)" \f f§

#### **4.4.1.4.2.9 Filter Maximum Velocity CSU Algorithms**tc "4.4.1.4.2.9 Filter Maximum Velocity CSU Algorithms"§

The Filter Maximum Velocity CSU does not utilize any algorithms.

#### **4.4.1.4.2.10 Filter Maximum Velocity CSU Limitations**tc "4.4.1.4.2.10 Filter Maximum Velocity CSU Limitations"§

There are no limitations or unusual features in the Filter Maximum Velocity CSU.

### **4.4.1.5 Filter Elevation CSU**tc "4.4.1.5 Filter Elevation CSU"§

The Filter Elevation CSU (DL-CSU-4.1.5) determines whether the elevation of an entity or event with respect to a reference entity is within the specified threshold range. The elevation is considered to be within the range if it lies between the first and second threshold angles. The following subparagraphs provide the design information for this CSU.

#### **4.4.1.5.1 Filter Elevation CSU Design Specification/Constraints**tc "4.4.1.5.1 Filter Elevation CSU Design Specification/Constraints"§

Table 4.4.1.5.1-1 identifies the requirements that are satisfied or partially satisfied by the Filter Elevation CSU. There are no design constraints for this CSU.

**Table 4.4.1.5.1-1**
**Requirements Allocated to the Filter Elevation CSU**tc " 4.4.1.5.1-1 Requirements Allocated to the Filter Elevation CSU" \f t§

| **Requirement ID** | **DL SRS Paragraph** |
|---|---|
| DL-R-16 | 3.2.4.1.5 |

**4.4.1.5.2 Filter Elevation CSU Design**tc "4.4.1.5.2Filter Elevation CSU Design"§

The following subparagraphs specify the design of the Filter Elevation CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.4.1.5.2.1 Filter Elevation CSU Input/Output Data Elements**tc "4.4.1.5.2.1 Filter Elevation CSU Input/Output Data Elements"§

Table 4.4.1.5.2.1-1 identifies and states the purpose of each input and output data element to the Filter Elevation CSU.

**Table 4.4.1.5.2.1-1**
**Filter Elevation CSU I/O Data**tc " 4.4.1.5.2.1-1 Filter Elevation CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Entity State PDU | Input | Describes the entity which lies at the origin of the entity coordinate system. | See DL IRS, Table 3.2.2-2, DL-D-01 |
| Position Of Interest | Input | Describes the geocentric position of a different entity or event in terms of the x,y,z components. | See DL IRS, Table 3.2.2-2, DL-D-03 |
| First Elevation Threshold | Input | Specifies the first value in the elevation threshold range. | See DL IRS, Table 3.2.2-2, DL-D-09 |
| Second Elevation Threshold | Input | Specifies the second value in the elevation threshold range. | See DL IRS, Table 3.2.2-2, DL-D-09 |
| Evaluation | Output | Indicates whether the position is at an angle that is within the specified threshold range, with respect to the entity described by the Entity State PDU. | See DL IRS, Table 3.2.2-2, DL-D-08 |

**4.4.1.5.2.2 Filter Elevation CSU Local Data Elements**tc "4.4.1.5.2.2 Filter Elevation CSU Local Data Elements"§

The Filter Elevation CSU does not utilize any local data elements.

**4.4.1.5.2.3 Filter Elevation CSU Global Data Elements**tc "4.4.1.5.2.3 Filter Elevation CSU Global Data Elements"§

The Filter Elevation CSU does not utilize any global data elements.

**4.4.1.5.2.4 Filter Elevation CSU Local and Shared Data Structures**tc "4.4.1.5.2.4 Filter Elevation CSU Local and Shared Data Structures"§

The Filter Elevation CSU does not implement any local or shared data structures.

**4.4.1.5.2.5 Filter Elevation CSU Interrupts and Signals**tc "4.4.1.5.2.5 Filter Elevation CSU Interrupts and Signals"§

The Filter Elevation CSU does not handle any interrupts or signals.

**4.4.1.5.2.6 Filter Elevation CSU Error Handling**tc "4.4.1.5.2.6 Filter Elevation CSU Error Handling"§

The Filter Elevation CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of FILT_EL to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Filter Elevation CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.4.1.5.2.7 Filter Elevation CSU Use of Other Elements**tc "4.4.1.5.2.7 Filter Elevation CSU Use of Other Elements"§

The Filter Elevation CSU does not use system service routines, global data files, or other global elements.

**4.4.1.5.2.8 Filter Elevation CSU Logic Flow**tc "4.4.1.5.2.8 Filter Elevation CSU Logic Flow"§

The logic flow of the Filter Elevation CSU is represented graphically by a flowchart in Figures 4.4.1.5.2.8-1 and 4.4.1.5.2.8-2. Execution of this CSU is initiated by the Filter List By Orientation CSC (DL-CSC-4.3) and by external CSCIs. This CSU initiates the Calculate Elevation CSU (DL-CSU-3.4).

µ §

**Figure 4.4.1.5.2.8-1**
**Filter Elevation CSU Flowchart (Part 1)**tc "4.4.1.5.2.8-1 Filter Elevation CSU Flowchart (Part 1)" \f f§

µ §

**Figure 4.4.1.5.2.8-2**
**Filter Elevation CSU Flowchart (Part 2)**tc "4.4.1.5.2.8-2 Filter Elevation CSU Flowchart (Part 2)" \f f§

**4.4.1.5.2.9 Filter Elevation CSU Algorithms**tc "4.4.1.5.2.9 Filter Elevation CSU Algorithms"§

The Filter Elevation CSU requires multiple comparisons to determine whether a position of interest is within the specified threshold range. The comparisons can be represented as a single block of conditionals which are stated in the following test:

**<u>Test</u>**

```
if (First Elevation Threshold <= Second Elevation Threshold) then
  if ((Calculated Elevation >= First Elevation Threshold) and
    (Calculated Elevation <= Second Elevation Threshold)) then
      Test Result is TRUE
  else
      Test Result is FALSE
  end if
else -- (First Elevation Threshold > Second Elevation Threshold)
  if ((Calculated Elevation >= First Elevation Threshold) or
     (Calculated Elevation <= Second Elevation Threshold)) then
      Test Result is TRUE
  else
      Test Result is FALSE
  end if
end if
```

**4.4.1.5.2.10 Filter Elevation CSU Limitations**tc "4.4.1.5.2.10 Filter Elevation CSU Limitations"§

There are no limitations or unusual features in the Filter Elevation CSU.

**4.4.1.6 Filter Azimuth and Elevation CSU**tc "4.4.1.6 Filter Azimuth and Elevation CSU"§

The Filter Azimuth and Elevation CSU (DL-CSU-4.1.6) filters an entity or event based on azimuth and elevation. The filter will determine whether the azimuth and elevation of an entity or event with respect to a reference entity are within the specified threshold ranges. The azimuth is considered to be within the range if it lies between the first and second azimuth threshold angles. The elevation is considered to be within the range if it lies between the first and second elevation threshold angles. The following subparagraphs provide the design information for this CSU.

#### 4.4.1.6.1 Filter Azimuth and Elevation CSU Design Specification/Constraintstc "4.4.1.6.1 Filter Azimuth and Elevation CSU Design Specification/Constraints"§

Table 4.4.1.6.1-1 identifies the requirements that are satisfied or partially satisfied by the Filter Azimuth and Elevation CSU. There are no design constraints for this CSU.

**Table 4.4.1.6.1-1**
**Requirements Allocated to the Filter Azimuth and Elevation CSU**tc " 4.4.1.6.1-1 Requirements Allocated to the Filter Azimuth and Elevation CSU" \f t§

| Requirement ID | DL SRS Paragraph |
|---|---|
| DL-R-17 | 3.2.4.1.6 |

**4.4.1.6.2 Filter Azimuth and Elevation CSU Design**tc "4.4.1.6.2 Filter Azimuth and Elevation CSU Design"§

The following subparagraphs specify the design of the Filter Azimuth and Elevation CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.4.1.6.2.1 Filter Azimuth and Elevation CSU Input/Output Data Elements**tc "4.4.1.6.2.1 Filter Azimuth and Elevation CSU Input/Output Data Elements"§

Table 4.4.1.6.2.1-1 identifies and states the purpose of each input and output data element to the Filter Azimuth and Elevation CSU.

**Table 4.4.1.6.2.1-1**
**Filter Azimuth and Elevation CSU I/O Data**tc " 4.4.1.6.2.1-1 Filter Azimuth and Elevation CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Entity State PDU | Input | Describes the entity which lies at the origin of the entity coordinate system. | See DL IRS, Table 3.2.2-2, DL-D-01 |
| Position Of Interest | Input | Describes the geocentric position of a different entity or event in terms of the x,y,z components. | See DL IRS, Table 3.2.2-2, DL-D-03 |
| First Azimuth Threshold | Input | Specifies the first value in the azimuth threshold range. | See DL IRS, Table 3.2.2-2, DL-D-09 |
| Second Azimuth Threshold | Input | Specifies the second value in the azimuth threshold range. | See DL IRS, Table 3.2.2-2, DL-D-09 |
| First Elevation Threshold | Input | Specifies the first value in the elevation threshold range. | See DL IRS, Table 3.2.2-2, DL-D-09 |
| Second | Input | Specifies the second value in the | See DL IRS, |

| | | | |
|---|---|---|---|
| Elevation Threshold | | elevation threshold range. | Table 3.2.2-2, DL-D-09 |
| Evaluation | Output | Indicates whether the position is at an angle that is within the specified threshold range, with respect to the entity described by the Entity State PDU. | See DL IRS, Table 3.2.2-2, DL-D-08 |

**4.4.1.6.2.2 Filter Azimuth and Elevation CSU Local Data Elements**tc "4.4.1.6.2.2 Filter Azimuth and Elevation CSU Local Data Elements"§

The Filter Azimuth and Elevation CSU does not utilize any local data elements.

**4.4.1.6.2.3 Filter Azimuth and Elevation CSU Global Data Elements**tc "4.4.1.6.2.3 Filter Azimuth and Elevation CSU Global Data Elements"§

The Filter Azimuth and Elevation CSU does not utilize any global data elements.

**4.4.1.6.2.4 Filter Azimuth and Elevation CSU Local and Shared Data Structures**tc "4.4.1.6.2.4 Filter Azimuth and Elevation CSU Local and Shared Data Structures"§

The Filter Azimuth and Elevation CSU does not implement any local or shared data structures.

**4.4.1.6.2.5 Filter Azimuth and Elevation CSU Interrupts and Signals**tc "4.4.1.6.2.5 Filter Azimuth and Elevation CSU Interrupts and Signals"§

The Filter Azimuth and Elevation CSU does not handle any interrupts or signals.

**4.4.1.6.2.6 Filter Azimuth and Elevation CSU Error Handling**tc "4.4.1.6.2.6 Filter Azimuth and Elevation CSU Error Handling"§

The Filter Azimuth and Elevation CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of FILT_AZ_AND_EL to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Filter Azimuth and Elevation CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.4.1.6.2.7 Filter Azimuth and Elevation CSU Use of Other Elements**tc "4.4.1.6.2.7 Filter Azimuth and Elevation CSU Use of Other Elements"§

The Filter Azimuth and Elevation CSU does not use system service routines, global data files, or other global elements.

**4.4.1.6.2.8 Filter Azimuth and Elevation CSU Logic Flow**tc "4.4.1.6.2.8 Filter Azimuth and Elevation CSU Logic Flow"§

The logic flow of the Filter Azimuth and Elevation CSU is represented graphically by a flowchart in Figures 4.4.1.6.2.8-1 and 4.4.1.6.2.8-2. Execution of this CSU is initiated by the Filter List By Orientation CSC (DL-CSC-4.3) and by external CSCIs. This CSU initiates the Calculate Azimuth and Elevation CSU (DL-CSU-3.2).

µ §

**Figure 4.4.1.6.2.8-1**
**Filter Azimuth and Elevation CSU Flowchart (Part 1)**tc "4.4.1.6.2.8-1 Filter Azimuth and Elevation CSU Flowchart (Part 1)" \f f§

µ §

**Figure 4.4.1.6.2.8-2**
**Filter Azimuth and Elevation CSU Flowchart (Part 2)**tc "4.4.1.6.2.8-2 Filter Azimuth and Elevation CSU Flowchart (Part 2)" \f f§

**4.4.1.6.2.9 Filter Azimuth and Elevation CSU Algorithms**tc "4.4.1.6.2.9 Filter Azimuth and Elevation CSU Algorithms"§

The Filter Azimuth and Elevation CSU utilizes algorithms which have been previously defined. The logic flow for this CSU references an algorithm for testing azimuth values in paragraph 4.4.1.2.2.9, and an algorithm for testing elevation values in paragraph 4.4.1.5.2.9.

**4.4.1.6.2.10 Filter Azimuth and Elevation CSU Limitations**tc "4.4.1.6.2.10 Filter Azimuth and Elevation CSU Limitations"§

There are no limitations or unusual features in the Filter Azimuth and Elevation CSU.

**4.4.2 Filter List By Distance CSC (DL-CSC-4.2)**tc "4.4.2Filter List By Distance CSC (DL-CSC-4.2)"§

The following subparagraphs identify and describe each of the CSUs of the Filter List By Distance CSC (DL-CSC-4.2). Figure 4.4.2-1 identifies the CSUs of this CSC. Figure 4.4.2-2 shows the execution control flow relationships of the CSUs of this CSC. Figure 4.4.2-3 shows the data flow relationships of the CSUs of this CSC. (In these figures, solid lines with no arrows indicate a hierarchical relationship, solid lines with arrows indicate data flow, and dashed lines with arrows indicate control flow. Rectangles with solid borders represent units internal to the CSC, and rectangles with dashed borders or small square boxes indicate external CSCs and CSCIs.)

µ §

**Figure 4.4.2-1**
**Filter List By Distance CSC Computer Software Units**tc "4.4.2-1 Filtering Routines CSC Computer Software Units" \f f§

µ §

**Figure 4.4.2-2**
**Filter List By Distance CSC Execution Control**tc "4.4.2-2 Filter List By Distance CSC Execution Control " \f f§
µ §

**Figure 4.4.2-3**
**Filter List By Distance CSC Data Flow**tc "4.4.2-3 Filter List By Distance CSC Data Flow" \f f §

**4.4.2.1 Filter Entity State Distance CSU**tc "4.4.2.1 Filter Entity State Distance CSU"§

The Filter Entity State Distance CSU (DL-CSU-4.2.1) evaluates each PDU within an input list of Entity State PDUs, and removes from the list those PDUs which do not meet the distance criteria. If the distance between the geocentric position specified in the PDU and the reference position is greater than the distance threshold, then the PDU is considered not to meet the distance criteria. The following subparagraphs provide the design information for this CSU.

**4.4.2.1.1 Filter Entity State Distance CSU Design Specification/Constraints**tc "4.4.2.1.1 Filter Entity State Distance CSU Design Specification/Constraints"§

Table 4.4.2.1.1-1 identifies the requirements that are satisfied or partially satisfied by the Filter Entity State Distance CSU. There are no design constraints for this CSU.

**Table 4.4.2.1.1-1**
**Requirements Allocated to the Filter Entity State Distance CSU**tc " 4.4.2.1.1-1 Requirements Allocated to the Filter Entity State Distance CSU" \f t§

| Requirement ID | DL SRS Paragraph |
|---|---|
| DL-R-14 | 3.2.4.2.1 |

**4.4.2.1.2 Filter Entity State Distance CSU Design**tc "4.4.2.1.2 Filter Entity State Distance CSU Design"§

The following subparagraphs specify the design of the Filter Entity State Distance CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.4.2.1.2.1 Filter Entity State Distance CSU Input/Output Data Elements**tc "4.4.2.1.2.1 Filter Entity State Distance CSU Input/Output Data Elements"§

Table 4.4.2.1.2.1-1 identifies and states the purpose of each input and output data element to the Filter Entity State Distance CSU.

**Table 4.4.2.1.2.1-1**
**Filter Entity State Distance CSU I/O Data**tc " 4.4.2.1.2.1-1 Filter Entity State Distance CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Entity State PDU List Head | Input/Output | Points to a list of pointers to Entity State PDUs. | See DL IRS, Table 3.2.2-2, DL-D-13 |
| Reference Position | Input | Used as a reference position to calculate the distance to each entity described by an Entity State PDU in the list. | See DL IRS, Table 3.2.2-2, DL-D-03 |
| Distance Threshold | Input | Specifies the maximum acceptable distance between the entities and the reference position. | See DL IRS, Table 3.2.2-2, DL-D-06 |

**4.4.2.1.2.2 Filter Entity State Distance CSU Local Data Elements**tc "4.4.2.1.2.2 Filter Entity State Distance CSU Local Data Elements"§

Table 4.4.2.1.2.2-1 identifies and describes each data element that originates within the Filter Entity State Distance CSU and which is not used by any other CSU.

| Name | Description | Data Type |
|---|---|---|
| Evaluation | Indicates whether an Entity State PDU (pointed to by a node in the input list ) contains a position whose distance from the reference position is within the threshold. | See DL IRS, Table 3.2.2-2, DL-D-08 |

**4.4.2.1.2.3 Filter Entity State Distance CSU Global Data Elements**tc "4.4.2.1.2.3 Filter Entity State Distance CSU Global Data Elements"§

The Filter Entity State Distance CSU does not utilize any global data elements.

**4.4.2.1.2.4 Filter Entity State Distance CSU Local and Shared Data Structures**tc "4.4.2.1.2.4 Filter Entity State Distance CSU Local and Shared Data Structures"§

The Filter Entity State Distance CSU does not implement any local or shared data structures.

**4.4.2.1.2.5 Filter Entity State Distance CSU Interrupts and Signals**tc "4.4.2.1.2.5 Filter Entity State Distance CSU Interrupts and Signals"§

The Filter Entity State Distance CSU does not handle any interrupts or signals.

**4.4.2.1.2.6 Filter Entity State Distance CSU Error Handling**tc "4.4.2.1.2.6 Filter Entity State Distance CSU Error Handling"§

The Filter Entity State Distance CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of FILT_ES_DIST to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Filter Entity State Distance CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.4.2.1.2.7 Filter Entity State Distance CSU Use of Other Elements**tc "4.4.2.1.2.7 Filter Entity State Distance CSU Use of Other Elements"§

The Filter Entity State Distance CSU does not use system service routines, global data files, or other global elements.

**4.4.2.1.2.8 Filter Entity State Distance CSU Logic Flow**tc "4.4.2.1.2.8 Filter Entity State Distance CSU Logic Flow"§

The logic flow of the Filter Entity State Distance CSU is represented graphically by a flowchart in Figure 4.4.2.1.2.8-1. Execution of this CSU is initiated as needed by units within external CSCIs.

µ §

**Figure 4.4.2.1.2.8-1**
**Filter Entity State Distance CSU Flowchart**tc "4.4.2.1.2.8-1 Filter Entity State Distance CSU Flowchart" \f f§

**4.4.2.1.2.9 Filter Entity State Distance CSU Algorithms**tc "4.4.2.1.2.9 Filter Entity State Distance CSU Algorithms"§

The Filter Entity State Distance CSU does not utilize any algorithms.

**4.4.2.1.2.10 Filter Entity State Distance CSU Limitations**tc "4.4.2.1.2.10 Filter Entity State Distance CSU Limitations"§

There are no limitations or unusual features in the Filter Entity State Distance CSU.

**4.4.2.2 Filter Fire Distance CSU**tc "4.4.2.2Filter Fire Distance CSU"§

The Filter Fire Distance CSU (DL-CSU-4.2.2) evaluates each PDU within an input list of Fire PDUs, and removes from the list those PDUs which do not meet the distance criteria. If the distance between the geocentric position specified in the PDU and the reference position is greater than the distance threshold, then the PDU is considered not to meet the distance criteria. The following subparagraphs provide the design information for this CSU.

**4.4.2.2.1 Filter Fire Distance CSU Design Specification/Constraints**tc "4.4.2.2.1 Filter Fire Distance CSU Design Specification/Constraints"§

Table 4.4.2.2.1-1 identifies the requirements that are satisfied or partially satisfied by the Filter Fire Distance CSU. There are no design constraints for this CSU.

**Table 4.4.2.2.1-1**
**Requirements Allocated to the Filter Fire Distance CSU**tc " 4.4.2.2.1-1 Requirements Allocated to the Filter Fire Distance CSU" \f t§

| Requirement ID | DL SRS Paragraph |
|---|---|
| DL-R-15 | 3.2.4.2.2 |

**4.4.2.2.2 Filter Fire Distance CSU Design**tc "4.4.2.2.2 Filter Fire Distance CSU Design"§

The following subparagraphs specify the design of the Filter Fire Distance CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.4.2.2.2.1 Filter Fire Distance CSU Input/Output Data Elements**tc "4.4.2.2.2.1 Filter Fire Distance CSU Input/Output Data Elements"§

Table 4.4.2.2.2.1-1 identifies and states the purpose of each input and output data element to the Filter Fire Distance CSU.

**Table 4.4.2.2.2.1-1**
**Filter Fire Distance CSU I/O Data**tc " 4.4.2.2.2.1-1 I/O Filter Fire Distance CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Fire PDU List Head | Input/Output | Points to a list of pointers to Fire PDUs. | See DL IRS, Table 3.2.2-2, DL-D-16 |
| Reference Position | Input | Used as a reference position to calculate the distance to each launched munition described by a Fire PDU in the list. | See DL IRS, Table 3.2.2-2, DL-D-03 |
| Distance Threshold | Input | Specifies the maximum acceptable distance between the entities and the reference position. | See DL IRS, Table 3.2.2-2, DL-D-06 |

**4.4.2.2.2.2 Filter Fire Distance CSU Local Data Elements**tc "4.4.2.2.2.2 Filter Fire Distance CSU Local Data Elements"§

Table 4.4.2.2.2.2-1 identifies and describes each data element that originates within the Filter Fire Distance CSU and which is not used by any other CSU.

| Name | Description | Data Type |
|---|---|---|
| Evaluation | Indicates whether a Fire PDU (pointed to by a node in the input list ) contains a position whose distance from the reference position is within the threshold. | See DL IRS, Table 3.2.2-2, DL-D-08 |

**4.4.2.2.2.3 Filter Fire Distance CSU Global Data Elements**tc "4.4.2.2.2.3 Filter Fire Distance CSU Global Data Elements"§

The Filter Fire Distance CSU does not utilize any global data elements.

**4.4.2.2.2.4 Filter Fire Distance CSU Local and Shared Data Structures**tc "4.4.2.2.2.4 Filter Fire Distance CSU Local and Shared Data Structures"§

The Filter Fire Distance CSU does not implement any local or shared data structures.

**4.4.2.2.2.5 Filter Fire Distance CSU Interrupts and Signals**tc "4.4.2.2.2.5 Filter Fire Distance CSU Interrupts and Signals"§

The Filter Fire Distance CSU does not handle any interrupts or signals.

**4.4.2.2.2.6 Filter Fire Distance CSU Error Handling**tc "4.4.2.2.2.6 Filter Fire Distance CSU Error Handling"§

The Filter Fire Distance CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of FILT_FIRE_DIST to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Filter Fire Distance CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.4.2.2.2.7 Filter Fire Distance CSU Use of Other Elements**tc "4.4.2.2.2.7 Filter Fire Distance CSU Use of Other Elements"§

The Filter Fire Distance CSU does not use system service routines, global data files, or other global elements.

**4.4.2.2.2.8 Filter Fire Distance CSU Logic Flow**tc "4.4.2.2.2.8 Filter Fire Distance CSU Logic Flow"§

The logic flow of the Filter Fire Distance CSU is represented graphically by a flowchart in Figure 4.4.2.2.2.8-1. Execution of this CSU is initiated as needed by units within external CSCIs.

µ §

**Filter Fire Distance CSU Flowchart**tc "4.4.2.2.2.8-1 Filter Fire Distance CSU Flowchart" \f f§

**4.4.2.2.2.9 Filter Fire Distance CSU Algorithms**tc "4.4.2.2.2.9 Filter Fire Distance CSU Algorithms"§

The Filter Fire Distance CSU does not utilize any algorithms.

**4.4.2.2.2.10 Filter Fire Distance CSU Limitations**tc "4.4.2.2.2.10 Filter Fire Distance CSU Limitations"§

There are no limitations or unusual features in the Filter Fire Distance CSU.

**4.4.2.3 Filter Detonation Distance CSU**tc "4.4.2.3 Filter Detonation Distance CSU"§

The Filter Detonation Distance CSU (DL-CSU-4.2.3) evaluates each PDU within an input list of Detonation PDUs, and removes from the list those PDUs which do not meet the distance criteria. If the distance between the geocentric position specified in the PDU and the reference position is greater than the distance threshold, then the PDU is considered not to meet the distance criteria. The following subparagraphs provide the design information for this CSU.

**4.4.2.3.1 Filter Detonation Distance CSU Design Specification/Constraints**tc "4.4.2.3.1 Filter Detonation Distance CSU Design Specification/Constraints"§

Table 4.4.2.3.1-1 identifies the requirements that are satisfied or partially satisfied by the Filter Detonation Distance CSU. There are no design constraints for this CSU.

**Table 4.4.2.3.1-1**
**Requirements Allocated to the Filter Detonation Distance CSU**tc " 4.4.2.3.1-1 Requirements Allocated to the Filter Detonation Distance CSU" \f t§

| Requirement ID | DL SRS Paragraph |
|---|---|
| DL-R-16 | 3.2.4.2.3 |

**4.4.2.3.2 Filter Detonation Distance CSU Design**tc "4.4.2.3.2 Filter Detonation Distance CSU Design"§

The following subparagraphs specify the design of the Filter Detonation Distance CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.4.2.3.2.1 Filter Detonation Distance CSU Input/Output Data Elements**tc "4.4.2.3.2.1 Filter Detonation Distance CSU Input/Output Data Elements"§

Table 4.4.2.3.2.1-1 identifies and states the purpose of each input and output data element to the Filter Detonation Distance CSU.

**Table 4.4.2.3.2.1-1**
**Filter Detonation Distance CSU I/O Data**tc " 4.4.2.3.2.1-1 Filter Detonation Distance CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Detonation PDU List Head | Input/Output | Points to a list of pointers to Detonation PDUs. | See DL IRS, Table 3.2.2-2, DL-D-19 |
| Reference Position | Input | Used as a reference position to calculate the distance to each detonation described by a Detonation PDU in the list. | See DL IRS, Table 3.2.2-2, DL-D-03 |
| Distance Threshold | Input | Specifies the maximum acceptable distance between the entities and the reference position. | See DL IRS, Table 3.2.2-2, DL-D-06 |

**4.4.2.3.2.2 Filter Detonation Distance CSU Local Data Elements**tc "4.4.2.3.2.2 Filter Detonation Distance CSU Local Data Elements"§

Table 4.4.2.3.2.2-1 identifies and describes each data element that originates within the Filter Detonation Distance CSU and which is not used by any other CSU.

| Name | Description | Data Type |
|---|---|---|
| Evaluation | Indicates whether a Detonation PDU (pointed to by a node in the input list ) contains a position whose distance from the reference position is within the threshold. | See DL IRS, Table 3.2.2-2, DL-D-08 |

**4.4.2.3.2.3 Filter Detonation Distance CSU Global Data Elements**tc "4.4.2.3.2.3 Filter Detonation Distance CSU Global Data Elements"§

The Filter Detonation Distance CSU does not utilize any global data elements.

**4.4.2.3.2.4 Filter Detonation Distance CSU Local and Shared Data Structures**tc "4.4.2.3.2.4 Filter Detonation Distance CSU Local and Shared Data Structures"§

The Filter Detonation Distance CSU does not implement any local or shared data structures.

**4.4.2.3.2.5 Filter Detonation Distance CSU Interrupts and Signals**tc "4.4.2.3.2.5 Filter Detonation Distance CSU Interrupts and Signals"§

The Filter Detonation Distance CSU does not handle any interrupts or signals.

**4.4.2.3.2.6 Filter Detonation Distance CSU Error Handling**tc "4.4.2.3.2.6 Filter Detonation Distance CSU Error Handling"§

The Filter Detonation Distance CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of FILT_DETONATION_DIST to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Filter Detonation Distance CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.4.2.3.2.7 Filter Detonation Distance CSU Use of Other Elements**tc "4.4.2.3.2.7 Filter Detonation Distance CSU Use of Other Elements"§

The Filter Detonation Distance CSU does not use system service routines, global data files, or other global elements.

**4.4.2.3.2.8 Filter Detonation Distance CSU Logic Flow**tc "4.4.2.3.2.8 Filter Detonation Distance CSU Logic Flow"§

The logic flow of the Filter Detonation Distance CSU is represented graphically by a flowchart in Figure 4.4.2.3.2.8-1. Execution of this CSU is initiated as needed by units within external CSCIs.

µ §

**Figure 4.4.2.3.2.8-1**
**Filter Detonation Distance CSU Flowchart**tc "4.4.2.3.2.8-1 Filter Detonation Distance CSU Flowchart" \f f§

#### 4.4.2.3.2.9 Filter Detonation Distance CSU Algorithmstc "4.4.2.3.2.9 Filter Detonation Distance CSU Algorithms"§

The Filter Detonation Distance CSU does not utilize any algorithms.

#### 4.4.2.3.2.10 Filter Detonation Distance CSU Limitationstc "4.4.2.3.2.10 Filter Detonation Distance CSU Limitations"§

There are no limitations or unusual features in the Filter Detonation Distance CSU.

### 4.4.2.4 Filter Laser Distance CSUtc "4.4.2.4 Filter Laser Distance CSU"§

The Filter Laser Distance CSU (DL-CSU-4.2.4) evaluates each PDU within an input list of Laser PDUs, and removes from the list those PDUs which do not meet the distance criteria. If the distance between the geocentric position specified in the PDU and the reference position is greater than the distance threshold, then the PDU is considered not to meet the distance criteria. The following subparagraphs provide the design information for this CSU.

#### 4.4.2.4.1 Filter Laser Distance CSU Design Specification/Constraintstc "4.4.2.4.1 Filter Laser Distance CSU Design Specification/Constraints"§

Table 4.4.2.4.1-1 identifies the requirements that are satisfied or partially satisfied by the Filter Laser Distance CSU. There are no design constraints for this CSU.

**Table 4.4.2.4.1-1**
**Requirements Allocated to the Filter Laser Distance CSU**tc " 4.4.2.4.1-1 Requirements Allocated to the Filter Laser Distance CSU" \f t§

| Requirement ID | DL SRS Paragraph |
|---|---|
| DL-R-17 | 3.2.4.2.4 |

#### 4.4.2.4.2 Filter Laser Distance CSU Designtc "4.4.2.4.2 Filter Laser Distance CSU Design"§

The following subparagraphs specify the design of the Filter Laser Distance CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

##### 4.4.2.4.2.1 Filter Laser Distance CSU Input/Output Data Elementstc "4.4.2.4.2.1 Filter Laser Distance CSU Input/Output Data Elements"§

Table 4.4.2.4.2.1-1 identifies and states the purpose of each input and output data element to the Filter Laser Distance CSU.

**Table 4.4.2.4.2.1-1**
**Filter Laser Distance CSU I/O Data**tc " 4.4.2.4.2.1-1 Filter Laser Distance CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Laser PDU List Head | Input/Output | Points to a list of pointers to Laser PDUs. | See DL IRS, Table 3.2.2-2, DL-D-22 |
| Reference Position | Input | Used as a reference position to calculate the distance to each laser spot described by a Laser PDU in the list. | See DL IRS, Table 3.2.2-2, DL-D-03 |
| Distance Threshold | Input | Specifies the maximum acceptable distance between the entities and the reference position. | See DL IRS, Table 3.2.2-2, DL-D-06 |

**4.4.2.4.2.2 Filter Laser Distance CSU Local Data Elements**tc "4.4.2.4.2.2 Filter Laser Distance CSU Local Data Elements"§

Table 4.4.2.4.2.2-1 identifies and describes each data element that originates within the Filter Laser Distance CSU and which is not used by any other CSU.

| Name | Description | Data Type |
|---|---|---|
| Evaluation | Indicates whether a Laser PDU (pointed to by a node in the input list ) contains a position whose distance from the reference position is within the threshold. | See DL IRS, Table 3.2.2-2, DL-D-08 |

**4.4.2.4.2.3 Filter Laser Distance CSU Global Data Elements**tc "4.4.2.4.2.3 Filter Laser Distance CSU Global Data Elements"§

The Filter Laser Distance CSU does not utilize any global data elements.

**4.4.2.4.2.4 Filter Laser Distance CSU Local and Shared Data Structures**tc "4.4.2.4.2.4 Filter Laser Distance CSU Local and Shared Data Structures"§

The Filter Laser Distance CSU does not implement any local or shared data structures.

**4.4.2.4.2.5 Filter Laser Distance CSU Interrupts and Signals**tc "4.4.2.4.2.5 Filter Laser Distance CSU Interrupts and Signals"§

The Filter Laser Distance CSU does not handle any interrupts or signals.

**4.4.2.4.2.6 Filter Laser Distance CSU Error Handling**tc "4.4.2.4.2.6 Filter Laser Distance CSU Error Handling"§

The Filter Laser Distance CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of FILT_LASER_DIST to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Filter Laser Distance CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.4.2.4.2.7  Filter Laser Distance CSU Use of Other Elements**tc "4.4.2.4.2.7	Filter Laser Distance CSU Use of Other Elements"§

The Filter Laser Distance CSU does not use system service routines, global data files, or other global elements.

**4.4.2.4.2.8  Filter Laser Distance CSU Logic Flow**tc "4.4.2.4.2.8	Filter Laser Distance CSU Logic Flow"§

The logic flow of the Filter Laser Distance CSU is represented graphically by a flowchart in Figure 4.4.2.4.2.8-1.  Execution of this CSU is initiated as needed by units within external CSCIs.

µ §

**Figure 4.4.2.4.2.8-1**
**Filter Laser Distance CSU Flowchart**tc "4.4.2.4.2.8-1	Filter Laser Distance CSU Flowchart" \f f§

#### 4.4.2.4.2.9 Filter Laser Distance CSU Algorithmstc "4.4.2.4.2.9 Filter Laser Distance CSU Algorithms"§

The Filter Laser Distance CSU does not utilize any algorithms.

#### 4.4.2.4.2.10 Filter Laser Distance CSU Limitationstc "4.4.2.4.2.10 Filter Laser Distance CSU Limitations"§

There are no limitations or unusual features in the Filter Laser Distance CSU.

#### 4.4.2.5 Filter Transmitter Distance CSUtc "4.4.2.5 Filter Transmitter Distance CSU"§

The Filter Transmitter Distance CSU (DL-CSU-4.2.5) evaluates each PDU within an input list of Transmitter PDUs, and removes from the list those PDUs which do not meet the distance criteria. If the distance between the geocentric position specified in the PDU and the reference position is greater than the distance threshold, then the PDU is considered not to meet the distance criteria. The following subparagraphs provide the design information for this CSU.

#### 4.4.2.5.1 Filter Transmitter Distance CSU Design Specification/Constraintstc "4.4.2.5.1 Filter Transmitter Distance CSU Design Specification/Constraints"§

Table 4.4.2.5.1-1 identifies the requirements that are satisfied or partially satisfied by the Filter Transmitter Distance CSU. There are no design constraints for this CSU.

**Table 4.4.2.5.1-1**
**Requirements Allocated to the Filter Transmitter Distance CSU**tc " 4.4.2.5.1-1 Requirements Allocated to the Filter Transmitter Distance CSU" \f t§

| Requirement ID | DL SRS Paragraph |
|---|---|
| DL-R-18 | 3.2.4.2.5 |

**4.4.2.5.2 Filter Transmitter Distance CSU Design**tc "4.4.2.5.2 Filter Transmitter Distance CSU Design"§

The following subparagraphs specify the design of the Filter Transmitter Distance CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.4.2.5.2.1 Filter Transmitter Distance CSU Input/Output Data Elements**tc "4.4.2.5.2.1 Filter Transmitter Distance CSU Input/Output Data Elements"§

Table 4.4.2.5.2.1-1 identifies and states the purpose of each input and output data element to the Filter Transmitter Distance CSU.

**Table 4.4.2.5.2.1-1**
**Filter Transmitter Distance CSU I/O Data**tc " 4.4.2.5.2.1-1 Filter Transmitter Distance CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Transmitter PDU List Head | Input/Output | Points to a list of pointers to Transmitter PDUs. | See DL IRS, Table 3.2.2-2, DL-D-25 |
| Reference Position | Input | Used as a reference position to calculate the distance to each antenna described by a Transmitter PDU in the list. | See DL IRS, Table 3.2.2-2, DL-D-03 |
| Distance Threshold | Input | Specifies the maximum acceptable distance between the entities and the reference position. | See DL IRS, Table 3.2.2-2, DL-D-06 |

**4.4.2.5.2.2 Filter Transmitter Distance CSU Local Data Elements**tc "4.4.2.5.2.2 Filter Transmitter Distance CSU Local Data Elements"§

Table 4.4.2.5.2.2-1 identifies and describes each data element that originates within the Filter Transmitter Distance CSU and which is not used by any other CSU.

| Name | Description | Data Type |
|---|---|---|
| Evaluation | Indicates whether a Transmitter PDU (pointed to by a node in the input list ) contains a position whose distance from the reference position is within the threshold. | See DL IRS, Table 3.2.2-2, DL-D-08 |

**4.4.2.5.2.3 Filter Transmitter Distance CSU Global Data Elements**tc "4.4.2.5.2.3 Filter Transmitter Distance CSU Global Data Elements"§

The Filter Transmitter Distance CSU does not utilize any global data elements.

**4.4.2.5.2.4 Filter Transmitter Distance CSU Local and Shared Data Structures**tc "4.4.2.5.2.4 Filter Transmitter Distance CSU Local and Shared Data Structures"§

The Filter Transmitter Distance CSU does not implement any local or shared data structures.

**4.4.2.5.2.5 Filter Transmitter Distance CSU Interrupts and Signals**tc "4.4.2.5.2.5 Filter Transmitter Distance CSU Interrupts and Signals"§

The Filter Transmitter Distance CSU does not handle any interrupts or signals.

**4.4.2.5.2.6 Filter Transmitter Distance CSU Error Handling**tc "4.4.2.5.2.6 Filter Transmitter Distance CSU Error Handling"§

The Filter Transmitter Distance CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of FILT_TRANSMITTER_DIST to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Filter Transmitter Distance CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.4.2.5.2.7 Filter Transmitter Distance CSU Use of Other Elements**tc "4.4.2.5.2.7 Filter Transmitter Distance CSU Use of Other Elements"§

The Filter Transmitter Distance CSU does not use system service routines, global data files, or other global elements.

**4.4.2.5.2.8 Filter Transmitter Distance CSU Logic Flow**tc "4.4.2.5.2.8 Filter Transmitter Distance CSU Logic Flow"§

The logic flow of the Filter Transmitter Distance CSU is represented graphically by a flowchart in Figure 4.4.2.5.2.8-1. Execution of this CSU is initiated as needed by units within external CSCIs.

µ §

**Figure 4.4.2.5.2.8-1**
**Filter Transmitter Distance CSU Flowchart**tc "4.4.2.5.2.8-1 Filter Transmitter Distance CSU Flowchart" \f f§

**4.4.2.5.2.10 Filter Transmitter Distance CSU Limitations**tc "4.4.2.5.2.10 Filter Transmitter Distance CSU Limitations"§

There are no limitations or unusual features in the Filter Transmitter Distance CSU.

**4.4.3 Filter List By Orientation CSC (DL-CSC-4.3)**tc "4.4.3 Filter List By Orientation CSC (DL-CSC-4.3)"§

The following subparagraphs identify and describe each of the CSUs of the Filter List By Orientation CSC (DL-CSC-4.3). Figure 4.4.3-1 identifies the CSUs of this CSC. Figure 4.4.3-2 shows the execution control flow relationships of the CSUs of this CSC. Figure 4.4.3-3 shows the data flow relationships of the CSUs of this CSC. (In these figures, solid lines with no arrows indicate a hierarchical relationship, solid lines with arrows indicate data flow, and dashed lines with arrows indicate control flow. Rectangles with solid borders represent units internal to the CSC, and rectangles with dashed borders or small square boxes indicate external CSCs and CSCIs.)

µ §

**Figure 4.4.3-1**
**Filter List By Orientation CSC Computer Software Units**tc "4.4.3-1 Filter List By Orientation CSC Computer Software Units" \f f§

µ §

**Figure 4.4.3-2**
**Filter List By Orientation CSC Execution Control**tc "4.4.3-2 Filter List By Orientation CSC Execution Control" \f f§

**Figure 4.4.3-3**
**Filter List By Orientation CSC Data Flow**tc "4.4.3-3	Filter List By Orientation CSC Data Flow" \f f §

### 4.4.3.1 Filter Entity State Azimuth CSUtc "4.4.3.1 Filter Entity State Azimuth CSU"§

The Filter Entity State Azimuth CSU (DL-CSU-4.3.1) evaluates each PDU within an input list of Entity State PDUs, and removes from the list those PDUs which are not within the specified azimuth range with respect to the heading of the reference entity. The following subparagraphs provide the design information for this CSU.

#### 4.4.3.1.1 Filter Entity State Azimuth CSU Design Specification/Constraintstc "4.4.3.1.1 Filter Entity State Azimuth CSU Design Specification/Constraints"§

Table 4.4.3.1.1-1 identifies the requirements that are satisfied or partially satisfied by the Filter Entity State Azimuth CSU. There are no design constraints for this CSU.

**Table 4.4.3.1.1-1**
**Requirements Allocated to the Filter Entity State Azimuth CSU**tc " 4.4.3.1.1-1 Requirements Allocated to the Filter Entity State Azimuth CSU" \f t§

| Requirement ID | DL SRS Paragraph |
|---|---|
| DL-R-19 | 3.2.4.3.1 |

#### 4.4.3.1.2 Filter Entity State Azimuth CSU Designtc "4.4.3.1.2 Filter Entity State Azimuth CSU Design"§

The following subparagraphs specify the design of the Filter Entity State Azimuth CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

##### 4.4.3.1.2.1 Filter Entity State Azimuth CSU Input/Output Data Elementstc "4.4.3.1.2.1 Filter Entity State Azimuth CSU Input/Output Data Elements"§

Table 4.4.3.1.2.1-1 identifies and states the purpose of each input and output data element to the Filter Entity State Azimuth CSU.

**Table 4.4.3.1.2.1-1**
**Filter Entity State Azimuth CSU I/O Data**tc " 4.4.3.1.2.1-1 Filter Entity State Azimuth CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Entity State PDU List Head | Input/Output | Points to a list of pointers to Entity State PDUs. | See DL IRS, Table 3.2.2-2, DL-D-13 |
| Reference Entity State PDU | Input | Used as a reference entity to calculate the relative azimuth of each entity in the list of Entity State PDUs. | See DL IRS, Table 3.2.2-2, DL-D-01 |
| First Azimuth Threshold | Input | Specifies the first value in the azimuth threshold range. | See DL IRS, Table 3.2.2-2, DL-D-09 |
| Second Azimuth Threshold | Input | Specifies the second value in the azimuth threshold range. | See DL IRS, Table 3.2.2-2, DL-D-09 |

**4.4.3.1.2.2 Filter Entity State Azimuth CSU Local Data Elements**tc "4.4.3.1.2.2 Filter Entity State Azimuth CSU Local Data Elements"§

The Filter Entity State Azimuth CSU does not utilize any local data elements.

**4.4.3.1.2.3 Filter Entity State Azimuth CSU Global Data Elements**tc "4.4.3.1.2.3 Filter Entity State Azimuth CSU Global Data Elements"§

The Filter Entity State Azimuth CSU does not utilize any global data elements.

**4.4.3.1.2.4 Filter Entity State Azimuth CSU Local and Shared Data Structures**tc "4.4.3.1.2.4 Filter Entity State Azimuth CSU Local and Shared Data Structures"§

The Filter Entity State Azimuth CSU does not implement any local or shared data structures.

**4.4.3.1.2.5 Filter Entity State Azimuth CSU Interrupts and Signals**tc "4.4.3.1.2.5 Filter Entity State Azimuth CSU Interrupts and Signals"§

The Filter Entity State Azimuth CSU does not handle any interrupts or signals.

**4.4.3.1.2.6 Filter Entity State Azimuth CSU Error Handling**tc "4.4.3.1.2.6 Filter Entity State Azimuth CSU Error Handling"§

The Filter Entity State Azimuth CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of FILT_ES_AZ to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Filter Entity State Azimuth CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.4.3.1.2.7 Filter Entity State Azimuth CSU Use of Other Elements**tc "4.4.3.1.2.7 Filter Entity State Azimuth CSU Use of Other Elements"§

The Filter Entity State Azimuth CSU does not use system service routines, global data files, or other global elements.

**4.4.3.1.2.8 Filter Entity State Azimuth CSU Logic Flow**tc "4.4.3.1.2.8 Filter Entity State Azimuth CSU Logic Flow"§

The logic flow of the Filter Entity State Azimuth CSU is represented graphically by a flowchart in Figure 4.4.3.1.2.8-1. Execution of this CSU is initiated as needed by units within external CSCIs.

µ §

**Figure 4.4.3.1.2.8-1**
**Filter Entity State Azimuth CSU Flowchart**tc "4.4.3.1.2.8-1 Filter Entity State Azimuth CSU Flowchart" \f f§

**4.4.3.1.2.9 Filter Entity State Azimuth CSU Algorithms**tc "4.4.3.1.2.9 Filter Entity State Azimuth CSU Algorithms"§

The Filter Entity State Azimuth CSU does not utilize any algorithms.

**4.4.3.1.2.10 Filter Entity State Azimuth CSU Limitations**tc "4.4.3.1.2.10 Filter Entity State Azimuth CSU Limitations"§

There are no limitations or unusual features in the Filter Entity State Azimuth CSU.

**4.4.3.2 Filter Fire Azimuth CSU**tc "4.4.3.2 Filter Fire Azimuth CSU"§

The Filter Fire Azimuth CSU (DL-CSU-4.3.2) evaluates each PDU within an input list of Fire PDUs, and removes from the list those PDUs which are not within the specified azimuth range with respect to the heading of the reference entity. The following subparagraphs provide the design information for this CSU.

**4.4.3.2.1 Filter Fire Azimuth Design Specification/Constraints**tc "4.4.3.2.1 Filter Fire Azimuth Design Specification/Constraints"§

Table 4.4.3.2.1-1 identifies the requirements that are satisfied or partially satisfied by the Filter Fire Azimuth CSU. There are no design constraints for this CSU.

**Table 4.4.3.2.1-1**
**Requirements Allocated to the Filter Fire Azimuth CSU**tc " 4.4.3.2.1-1 Requirements Allocated to the Filter Fire Azimuth CSU" \f t§

| Requirement ID | DL SRS Paragraph |
|---|---|
| DL-R-20 | 3.2.4.3.2 |

**4.4.3.2.2 Filter Fire Azimuth Design**tc "4.4.3.2.2 Filter Fire Azimuth Design"§

The following subparagraphs specify the design of the Filter Fire Azimuth CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.4.3.2.2.1 Filter Fire Azimuth Input/Output Data Elements**tc "4.4.3.2.2.1 Filter Fire Azimuth Input/Output Data Elements"§

Table 4.4.3.2.2.1-1 identifies and states the purpose of each input and output data element to the Filter Fire Azimuth CSU.

**Table 4.4.3.2.2.1-1**
**Filter Fire Azimuth CSU I/O Data**tc " 4.4.3.2.2.1-1 Filter Fire Azimuth CSU I/O Data" \f t§

| **Data Element** | **Input/Output** | **Purpose** | **Data Type** |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Fire PDU List Head | Input/Output | Points to a list of pointers to Fire PDUs. | See DL IRS, Table 3.2.2-2, DL-D-16 |
| Reference Entity State PDU | Input | Used as a reference entity to calculate the relative azimuth of each launched munition in the list of Fire PDUs. | See DL IRS, Table 3.2.2-2, DL-D-01 |
| First Azimuth Threshold | Input | Specifies the first value in the azimuth threshold range. | See DL IRS, Table 3.2.2-2, DL-D-09 |
| Second Azimuth Threshold | Input | Specifies the second value in the azimuth threshold range. | See DL IRS, Table 3.2.2-2, DL-D-09 |

**4.4.3.2.2.2 Filter Fire Azimuth Local Data Elements**tc "4.4.3.2.2.2 Filter Fire Azimuth Local Data Elements"§

The Filter Fire Azimuth CSU does not utilize any local data elements.

**4.4.3.2.2.3 Filter Fire Azimuth Global Data Elements**tc "4.4.3.2.2.3 Filter Fire Azimuth Global Data Elements"§

The Filter Fire Azimuth CSU does not utilize any global data elements.

**4.4.3.2.2.4 Filter Fire Azimuth Local and Shared Data Structures**tc "4.4.3.2.2.4 Filter Fire Azimuth Local and Shared Data Structures"§

The Filter Fire Azimuth CSU does not implement any local or shared data structures.

**4.4.3.2.2.5 Filter Fire Azimuth Interrupts and Signals**tc "4.4.3.2.2.5 Filter Fire Azimuth Interrupts and Signals"§

The Filter Fire Azimuth CSU does not handle any interrupts or signals.

**4.4.3.2.2.6 Filter Fire Azimuth Error Handling**tc "4.4.3.2.2.6 Filter Fire Azimuth Error Handling"§

The Filter Fire Azimuth CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of FILT_FIRE_AZ to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Filter Fire Azimuth CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.4.3.2.2.7 Filter Fire Azimuth Use of Other Elements**tc "4.4.3.2.2.7 Filter Fire Azimuth Use of Other Elements"§

The Filter Fire Azimuth CSU does not use system service routines, global data files, or other global elements.

**4.4.3.2.2.8 Filter Fire Azimuth Logic Flow**tc "4.4.3.2.2.8 Filter Fire Azimuth Logic Flow"§

The logic flow of the Filter Fire Azimuth CSU is represented graphically by a flowchart in Figure 4.4.3.2.2.8-1. Execution of this CSU is initiated as needed by units within external CSCIs.

µ §

**Figure 4.4.3.2.2.8-1**
**Filter Fire Azimuth CSU Flowchart**tc "4.4.3.2.2.8-1 Filter Fire Azimuth CSU Flowchart" \f f§

**4.4.3.2.2.9 Filter Fire Azimuth Algorithms**tc "4.4.3.2.2.9 Filter Fire Azimuth Algorithms"§

The Filter Fire Azimuth CSU does not utilize any algorithms.

**4.4.3.2.2.10 Filter Fire Azimuth Limitations**tc "4.4.3.2.2.10 Filter Fire Azimuth Limitations"§

There are no limitations or unusual features in the Filter Fire Azimuth CSU.

**4.4.3.3 Filter Detonation Azimuth CSU**tc "4.4.3.3Filter Detonation Azimuth CSU"§

The Filter Detonation Azimuth CSU (DL-CSU-4.3.3) evaluates each PDU within an input list of Detonation PDUs, and removes from the list those PDUs which are not within the specified azimuth range with respect to the heading of the reference entity. The following subparagraphs provide the design information for this CSU.

**4.4.3.3.1 Filter Detonation Azimuth Design Specification/Constraints**tc "4.4.3.3.1 Filter Detonation Azimuth Design Specification/Constraints"§

Table 4.4.3.3.1-1 identifies the requirements that are satisfied or partially satisfied by the Filter Detonation Azimuth CSU. There are no design constraints for this CSU.

**Table 4.4.3.3.1-1**
**Requirements Allocated to the Filter Detonation Azimuth CSU**tc " 4.4.3.3.1-1 Requirements Allocated to the Filter Detonation Azimuth CSU" \f t§

| Requirement ID | DL SRS Paragraph |
|---|---|
| DL-R-21 | 3.2.4.3.3 |

**4.4.3.3.2 Filter Detonation Azimuth Design**tc "4.4.3.3.2 Filter Detonation Azimuth Design"§

The following subparagraphs specify the design of the Filter Detonation Azimuth CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.4.3.3.2.1 Filter Detonation Azimuth Input/Output Data Elements**tc "4.4.3.3.2.1 Filter Detonation Azimuth Input/Output Data Elements"§

Table 4.4.3.3.2.1-1 identifies and states the purpose of each input and output data element to the Filter Detonation Azimuth CSU.

**Table 4.4.3.3.2.1-1**
**Filter Detonation Azimuth CSU I/O Data**tc " 4.4.3.3.2.1-1 Filter Detonation Azimuth CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Detonation PDU List Head | Input/Output | Points to a list of pointers to Detonation PDUs. | See DL IRS, Table 3.2.2-2, DL-D-19 |
| Reference Entity State PDU | Input | Used as a reference entity to calculate the relative azimuth of each detonation in the list of Detonation PDUs. | See DL IRS, Table 3.2.2-2, DL-D-01 |
| First Azimuth Threshold | Input | Specifies the first value in the azimuth threshold range. | See DL IRS, Table 3.2.2-2, DL-D-09 |
| Second Azimuth Threshold | Input | Specifies the second value in the azimuth threshold range. | See DL IRS, Table 3.2.2-2, DL-D-09 |

**4.4.3.3.2.2 Filter Detonation Azimuth Local Data Elements**tc "4.4.3.3.2.2 Filter Detonation Azimuth Local Data Elements"§

The Filter Detonation Azimuth CSU does not utilize any local data elements.

**4.4.3.3.2.3 Filter Detonation Azimuth Global Data Elements**tc "4.4.3.3.2.3 Filter Detonation Azimuth Global Data Elements"§

The Filter Detonation Azimuth CSU does not utilize any global data elements.

**4.4.3.3.2.4 Filter Detonation Azimuth Local and Shared Data Structures**tc "4.4.3.3.2.4 Filter Detonation Azimuth Local and Shared Data Structures"§

The Filter Detonation Azimuth CSU does not implement any local or shared data structures.

**4.4.3.3.2.5 Filter Detonation Azimuth Interrupts and Signals**tc "4.4.3.3.2.5 Filter Detonation Azimuth Interrupts and Signals"§

The Filter Detonation Azimuth CSU does not handle any interrupts or signals.

**4.4.3.3.2.6 Filter Detonation Azimuth Error Handling**tc "4.4.3.3.2.6 Filter Detonation Azimuth Error Handling"§

The Filter Detonation Azimuth CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of FILT_DETONATION_AZ to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Filter Detonation Azimuth CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.4.3.3.2.7 Filter Detonation Azimuth Use of Other Elements**tc "4.4.3.3.2.7 Filter Detonation Azimuth Use of Other Elements"§

The Filter Detonation Azimuth CSU does not use system service routines, global data files, or other global elements.

**4.4.3.3.2.8 Filter Detonation Azimuth Logic Flow**tc "4.4.3.3.2.8 Filter Detonation Azimuth Logic Flow"§

The logic flow of the Filter Detonation Azimuth CSU is represented graphically by a flowchart in Figure 4.4.3.3.2.8-1. Execution of this CSU is initiated as needed by units within external CSCIs.

µ §

**Figure 4.4.3.3.2.8-1**
**Filter Detonation Azimuth CSU Flowchart**tc "4.4.3.3.2.8-1 Filter Detonation Azimuth CSU Flowchart" \f f§

**4.4.3.3.2.9 Filter Detonation Azimuth Algorithms**tc "4.4.3.3.2.9 Filter Detonation Azimuth Algorithms"§

The Filter Detonation Azimuth CSU does not utilize any algorithms.

**4.4.3.3.2.10 Filter Detonation Azimuth Limitations**tc "4.4.3.3.2.10 Filter Detonation Azimuth Limitations"§

There are no limitations or unusual features in the Filter Detonation Azimuth CSU.

**4.4.3.4 Filter Laser Azimuth CSU**tc "4.4.3.4 Filter Laser Azimuth CSU"§

The Filter Laser Azimuth CSU (DL-CSU-4.3.4) evaluates each PDU within an input list of Laser PDUs, and removes from the list those PDUs which are not within the specified azimuth range with respect to the heading of the reference entity. The following subparagraphs provide the design information for this CSU.

**4.4.3.4.1 Filter Laser Azimuth Design Specification/Constraints**tc "4.4.3.4.1 Filter Laser Azimuth Design Specification/Constraints"§

Table 4.4.3.4.1-1 identifies the requirements that are satisfied or partially satisfied by the Filter Laser Azimuth CSU. There are no design constraints for this CSU.

**Table 4.4.3.4.1-1**
**Requirements Allocated to the Filter Laser Azimuth CSU**tc " 4.4.3.4.1-1 Requirements Allocated to the Filter Laser Azimuth CSU" \f t§

| Requirement ID | DL SRS Paragraph |
|---|---|
| DL-R-22 | 3.2.4.3.4 |

**4.4.3.4.2 Filter Laser Azimuth Design**tc "4.4.3.4.2 Filter Laser Azimuth Design"§

The following subparagraphs specify the design of the Filter Laser Azimuth CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.4.3.4.2.1 Filter Laser Azimuth Input/Output Data Elements**tc "4.4.3.4.2.1 Filter Laser Azimuth Input/Output Data Elements"§

Table 4.4.3.4.2.1-1 identifies and states the purpose of each input and output data element to the Filter Laser Azimuth CSU.

**Table 4.4.3.4.2.1-1**
**Filter Laser Azimuth CSU I/O Data**tc " 4.4.3.4.2.1-1 Filter Laser Azimuth CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Laser PDU List Head | Input/Output | Points to a list of pointers to Laser PDUs. | See DL IRS, Table 3.2.2-2, DL-D-22 |
| Reference Entity State PDU | Input | Used as a reference entity to calculate the relative azimuth of each laser spot in the list of Laser PDUs. | See DL IRS, Table 3.2.2-2, DL-D-01 |
| First Azimuth Threshold | Input | Specifies the first value in the azimuth threshold range. | See DL IRS, Table 3.2.2-2, DL-D-09 |
| Second Azimuth Threshold | Input | Specifies the second value in the azimuth threshold range. | See DL IRS, Table 3.2.2-2, DL-D-09 |

**4.4.3.4.2.2 Filter Laser Azimuth Local Data Elements**tc "4.4.3.4.2.2 Filter Laser Azimuth Local Data Elements"§

The Filter Laser Azimuth CSU does not utilize any local data elements.

**4.4.3.4.2.3 Filter Laser Azimuth Global Data Elements**tc "4.4.3.4.2.3 Filter Laser Azimuth Global Data Elements"§

The Filter Laser Azimuth CSU does not utilize any global data elements.

**4.4.3.4.2.4 Filter Laser Azimuth Local and Shared Data Structures**tc "4.4.3.4.2.4 Filter Laser Azimuth Local and Shared Data Structures"§

The Filter Laser Azimuth CSU does not implement any local or shared data structures.

**4.4.3.4.2.5 Filter Laser Azimuth Interrupts and Signals**tc "4.4.3.4.2.5 Filter Laser Azimuth Interrupts and Signals"§

The Filter Laser Azimuth CSU does not handle any interrupts or signals.

**4.4.3.4.2.6 Filter Laser Azimuth Error Handling**tc "4.4.3.4.2.6 Filter Laser Azimuth Error Handling"§

The Filter Laser Azimuth CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of FILT_LASER_AZ to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Filter Laser Azimuth CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.4.3.4.2.7 Filter Laser Azimuth Use of Other Elements**tc "4.4.3.4.2.7 Filter Laser Azimuth Use of Other Elements"§

The Filter Laser Azimuth CSU does not use system service routines, global data files, or other global elements.

**4.4.3.4.2.8 Filter Laser Azimuth Logic Flow**tc "4.4.3.4.2.8 Filter Laser Azimuth Logic Flow"§

The logic flow of the Filter Laser Azimuth CSU is represented graphically by a flowchart in Figure 4.4.3.4.2.8-1. Execution of this CSU is initiated as needed by units within external CSCIs.

µ §

**Figure 4.4.3.4.2.8-1**
**Filter Laser Azimuth CSU Flowchart**tc "4.4.3.4.2.8-1 Filter Laser Azimuth CSU Flowchart" \f f§

**4.4.3.4.2.9 Filter Laser Azimuth Algorithms**tc "4.4.3.4.2.9 Filter Laser Azimuth Algorithms"§

The Filter Laser Azimuth CSU does not utilize any algorithms.

**4.4.3.4.2.10 Filter Laser Azimuth Limitations**tc "4.4.3.4.2.10 Filter Laser Azimuth Limitations"§

There are no limitations or unusual features in the Filter Laser Azimuth CSU.

**4.4.3.5 Filter Transmitter Azimuth CSU**tc "4.4.3.5 Filter Transmitter Azimuth CSU"§

The Filter Transmitter Azimuth CSU (DL-CSU-4.3.5) evaluates each PDU within an input list of Transmitter PDUs, and removes from the list those PDUs which are not within the specified azimuth range with respect to the heading of the reference entity. The following subparagraphs provide the design information for this CSU.

**4.4.3.5.1 Filter Transmitter Azimuth Design Specification/Constraints**tc "4.4.3.5.1 Filter Transmitter Azimuth Design Specification/Constraints"§

Table 4.4.3.5.1-1 identifies the requirements that are satisfied or partially satisfied by the Filter Transmitter Azimuth CSU. There are no design constraints for this CSU.

**Table 4.4.3.5.1-1**
**Requirements Allocated to the Filter Transmitter Azimuth CSU**tc " 4.4.3.5.1-1 Requirements Allocated to the Filter Transmitter Azimuth CSU" \f t§

| Requirement ID | DL SRS Paragraph |
|---|---|
| DL-R-23 | 3.2.4.3.5 |

**4.4.3.5.2 Filter Transmitter Azimuth Design**tc "4.4.3.5.2 Filter Transmitter Azimuth Design"§

The following subparagraphs specify the design of the Filter Transmitter Azimuth CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.4.3.5.2.1 Filter Transmitter Azimuth Input/Output Data Elements**tc "4.4.3.5.2.1 Filter Transmitter Azimuth Input/Output Data Elements"§

Table 4.4.3.5.2.1-1 identifies and states the purpose of each input and output data element to the Filter Transmitter Azimuth CSU.

**Table 4.4.3.5.2.1-1**
**Filter Transmitter Azimuth CSU I/O Data**tc " 4.4.3.5.2.1-1 Filter Transmitter Azimuth CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Transmitter PDU List Head | Input/Output | Points to a list of pointers to Transmitter PDUs. | See DL IRS, Table 3.2.2-2, DL-D-25 |
| Reference Entity State PDU | Input | Used as a reference entity to calculate the relative azimuth of each antenna in the list of Transmitter PDUs. | See DL IRS, Table 3.2.2-2, DL-D-01 |
| First Azimuth Threshold | Input | Specifies the first value in the azimuth threshold range. | See DL IRS, Table 3.2.2-2, DL-D-09 |
| Second Azimuth Threshold | Input | Specifies the second value in the azimuth threshold range. | See DL IRS, Table 3.2.2-2, DL-D-09 |

**4.4.3.5.2.2 Filter Transmitter Azimuth Local Data Elements**tc "4.4.3.5.2.2 Filter Transmitter Azimuth Local Data Elements"§

The Filter Transmitter Azimuth CSU does not utilize any local data elements.

**4.4.3.5.2.3 Filter Transmitter Azimuth Global Data Elements**tc "4.4.3.5.2.3 Filter Transmitter Azimuth Global Data Elements"§

The Filter Transmitter Azimuth CSU does not utilize any global data elements.

**4.4.3.5.2.4 Filter Transmitter Azimuth Local and Shared Data Structures**tc "4.4.3.5.2.4 Filter Transmitter Azimuth Local and Shared Data Structures"§

The Filter Transmitter Azimuth CSU does not implement any local or shared data structures.

**4.4.3.5.2.5 Filter Transmitter Azimuth Interrupts and Signals**tc "4.4.3.5.2.5 Filter Transmitter Azimuth Interrupts and Signals"§

The Filter Transmitter Azimuth CSU does not handle any interrupts or signals.

**4.4.3.5.2.6 Filter Transmitter Azimuth Error Handling**tc "4.4.3.5.2.6 Filter Transmitter Azimuth Error Handling"§

The Filter Transmitter Azimuth CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of FILT_TRANSMITTER_AZ to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Filter Transmitter Azimuth CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.4.3.5.2.7 Filter Transmitter Azimuth Use of Other Elements**tc "4.4.3.5.2.7 Filter Transmitter Azimuth Use of Other Elements"§

The Filter Transmitter Azimuth CSU does not use system service routines, global data files, or other global elements.

**4.4.3.5.2.8 Filter Transmitter Azimuth Logic Flow**tc "4.4.3.5.2.8 Filter Transmitter Azimuth Logic Flow"§

The logic flow of the Filter Transmitter Azimuth CSU is represented graphically by a flowchart in Figure 4.4.3.5.2.8-1. Execution of this CSU is initiated as needed by units within external CSCIs.

µ §

**Figure 4.4.3.5.2.8-1**
**Filter Transmitter Azimuth CSU Flowchart**tc "4.4.3.5.2.8-1 Filter Transmitter Azimuth CSU Flowchart" \f f§

**4.4.3.5.2.9 Filter Transmitter Azimuth Algorithms**tc "4.4.3.5.2.9 Filter Transmitter Azimuth Algorithms"§

The Filter Transmitter Azimuth CSU does not utilize any algorithms.

**4.4.3.5.2.10 Filter Transmitter Azimuth Limitations**tc "4.4.3.5.2.10 Filter Transmitter Azimuth Limitations"§

There are no limitations or unusual features in the Filter Transmitter Azimuth CSU.

**4.4.3.6 Filter Entity State Elevation CSU**tc "4.4.3.6 Filter Entity State Elevation CSU"§

The Filter Entity State Elevation CSU (DL-CSU-4.3.6) evaluates each PDU within an input list of Entity State PDUs, and removes from the list those PDUs which are not within the specified elevation range with respect to the heading of the reference entity. The following subparagraphs provide the design information for this CSU.

**4.4.3.6.1 Filter Entity State Elevation CSU Design Specification/Constraints**tc "4.4.3.6.1 Filter Entity State Elevation CSU Design Specification/Constraints"§

Table 4.4.3.6.1-1 identifies the requirements that are satisfied or partially satisfied by the Filter Entity State Elevation CSU. There are no design constraints for this CSU.

**Table 4.4.3.6.1-1**
**Requirements Allocated to the Filter Entity State Elevation CSU**tc " 4.4.3.6.1-1 Requirements Allocated to the Filter Entity State Elevation CSU" \f t§

| Requirement ID | DL SRS Paragraph |
|---|---|
| DL-R-28 | 3.2.4.3.6 |

**4.4.3.6.2 Filter Entity State Elevation CSU Design**tc "4.4.3.6.2 Filter Entity State Elevation CSU Design"§

The following subparagraphs specify the design of the Filter Entity State Elevation CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.4.3.6.2.1 Filter Entity State Elevation CSU Input/Output Data Elements**tc "4.4.3.6.2.1 Filter Entity State Elevation CSU Input/Output Data Elements"§

Table 4.4.3.6.2.1-1 identifies and states the purpose of each input and output data element to the Filter Entity State Elevation CSU.

**Table 4.4.3.6.2.1-1**
**Filter Entity State Elevation CSU I/O Data**tc " 4.4.3.6.2.1-1 Filter Entity State Elevation CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Entity State PDU List Head | Input/Output | Points to a list of pointers to Entity State PDUs. | See DL IRS, Table 3.2.2-2, DL-D-13 |
| Reference Entity State PDU | Input | Used as a reference entity to calculate the relative elevation of each entity in the list of Entity State PDUs. | See DL IRS, Table 3.2.2-2, DL-D-01 |
| First Elevation Threshold | Input | Specifies the first value in the elevation threshold range. | See DL IRS, Table 3.2.2-2, DL-D-09 |
| Second Elevation Threshold | Input | Specifies the second value in the elevation threshold range. | See DL IRS, Table 3.2.2-2, DL-D-09 |

**4.4.3.6.2.2 Filter Entity State Elevation CSU Local Data Elements**tc "4.4.3.6.2.2 Filter Entity State Elevation CSU Local Data Elements"§

The Filter Entity State Elevation CSU does not utilize any local data elements.

**4.4.3.6.2.3 Filter Entity State Elevation CSU Global Data Elements**tc "4.4.3.6.2.3 Filter Entity State Elevation CSU Global Data Elements"§

The Filter Entity State Elevation CSU does not utilize any global data elements.

**4.4.3.6.2.4 Filter Entity State Elevation CSU Local and Shared Data Structures**tc "4.4.3.6.2.4 Filter Entity State Elevation CSU Local and Shared Data Structures"§

The Filter Entity State Elevation CSU does not implement any local or shared data structures.

**4.4.3.6.2.5 Filter Entity State Elevation CSU Interrupts and Signals**tc "4.4.3.6.2.5 Filter Entity State Elevation CSU Interrupts and Signals"§

The Filter Entity State Elevation CSU does not handle any interrupts or signals.

**4.4.3.6.2.6 Filter Entity State Elevation CSU Error Handling**tc "4.4.3.6.2.6 Filter Entity State Elevation CSU Error Handling"§

The Filter Entity State Elevation CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of FILT_ES_AZ to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Filter Entity State Elevation CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.4.3.6.2.7 Filter Entity State Elevation CSU Use of Other Elements**tc "4.4.3.6.2.7 Filter Entity State Elevation CSU Use of Other Elements"§

The Filter Entity State Elevation CSU does not use system service routines, global data files, or other global elements.

**4.4.3.6.2.8 Filter Entity State Elevation CSU Logic Flow**tc "4.4.3.6.2.8 Filter Entity State Elevation CSU Logic Flow"§

The logic flow of the Filter Entity State Elevation CSU is represented graphically by a flowchart in Figure 4.4.3.6.2.8-1. Execution of this CSU is initiated as needed by units within external CSCIs. This CSU initiates the Filter Elevation CSU (DL-CSU-4.1.5).

µ §

**Figure 4.4.3.6.2.8-1**
**Filter Entity State Elevation CSU Flowchart**tc "4.4.3.6.2.8-1 Filter Entity State Elevation CSU Flowchart" \f f§

**4.4.3.6.2.9 Filter Entity State Elevation CSU Algorithms**tc "4.4.3.6.2.9 Filter Entity State Elevation CSU Algorithms"§

The Filter Entity State Elevation CSU does not utilize any algorithms.

**4.4.3.6.2.10 Filter Entity State Elevation CSU Limitations**tc "4.4.3.6.2.10 Filter Entity State Elevation CSU Limitations"§

There are no limitations or unusual features in the Filter Entity State Elevation CSU.

**4.4.3.7 Filter Entity State Azimuth and Elevation CSU**tc "4.4.3.7 Filter Entity State Azimuth and Elevation CSU"§

The Filter Entity State Azimuth and Elevation CSU (DL-CSU-4.3.7) evaluates each PDU within an input list of Entity State PDUs, and removes from the list those PDUs which are not within the specified azimuth range and the specified elevation range with respect to the reference entity. The following subparagraphs provide the design information for this CSU.

**4.4.3.7.1 Filter Entity State Azimuth and Elevation CSU Design Specification/Constraints**tc "4.4.3.7.1 Filter Entity State Azimuth and Elevation CSU Design Specification/Constraints"§

Table 4.4.3.7.1-1 identifies the requirements that are satisfied or partially satisfied by the Filter Entity State Azimuth and Elevation CSU. There are no design constraints for this CSU.

**Table 4.4.3.7.1-1**
**Requirements Allocated to the Filter Entity State Azimuth and Elevation CSU**tc " 4.4.3.7.1-1 Requirements Allocated to the Filter Entity State Azimuth and Elevation CSU" \f t§

| Requirement ID | DL SRS Paragraph |
|---|---|
| DL-R-29 | 3.2.4.3.7 |

**4.4.3.7.2 Filter Entity State Azimuth and Elevation CSU Design**tc "4.4.3.7.2 Filter Entity State Azimuth and Elevation CSU Design"§

The following subparagraphs specify the design of the Filter Entity State Azimuth and Elevation CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.4.3.7.2.1 Filter Entity State Azimuth and Elevation CSU Input/Output Data Elements**tc "4.4.3.7.2.1 Filter Entity State Azimuth and Elevation CSU Input/Output Data Elements"§

Table 4.4.3.7.2.1-1 identifies and states the purpose of each input and output data element to the Filter Entity State Azimuth and Elevation CSU.

**Table 4.4.3.7.2.1-1**
**Filter Entity State Azimuth and Elevation CSU I/O Data**tc " 4.4.3.7.2.1-1 Filter Entity State Azimuth and Elevation CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Entity State PDU List Head | Input/Output | Points to a list of pointers to Entity State PDUs. | See DL IRS, Table 3.2.2-2, DL-D-13 |
| Reference Entity State PDU | Input | Used as a reference entity to calculate the relative elevation of each entity in the list of Entity State PDUs. | See DL IRS, Table 3.2.2-2, DL-D-01 |
| First Azimuth Threshold | Input | Specifies the first value in the azimuth threshold range. | See DL IRS, Table 3.2.2-2, DL-D-09 |
| Second Azimuth Threshold | Input | Specifies the second value in the azimuth threshold range. | See DL IRS, Table 3.2.2-2, DL-D-09 |
| First Elevation Threshold | Input | Specifies the first value in the elevation threshold range. | See DL IRS, Table 3.2.2-2, DL-D-09 |
| Second Elevation Threshold | Input | Specifies the second value in the elevation threshold range. | See DL IRS, Table 3.2.2-2, DL-D-09 |

**4.4.3.7.2.2 Filter Entity State Azimuth and Elevation CSU Local Data Elements**tc "4.4.3.7.2.2 Filter Entity State Azimuth and Elevation CSU Local Data Elements"§

The Filter Entity State Azimuth and Elevation CSU does not utilize any local data elements.

**4.4.3.7.2.3 Filter Entity State Azimuth and Elevation CSU Global Data Elements**tc "4.4.3.7.2.3 Filter Entity State Azimuth and Elevation CSU Global Data Elements"§

The Filter Entity State Azimuth and Elevation CSU does not utilize any global data elements.

**4.4.3.7.2.4 Filter Entity State Azimuth and Elevation CSU Local and Shared Data Structures**tc "4.4.3.7.2.4 Filter Entity State Azimuth and Elevation CSU Local and Shared Data Structures"§

The Filter Entity State Azimuth and Elevation CSU does not implement any local or shared data structures.

**4.4.3.7.2.5 Filter Entity State Azimuth and Elevation CSU Interrupts and Signals**tc "4.4.3.7.2.5 Filter Entity State Azimuth and Elevation CSU Interrupts and Signals"§

The Filter Entity State Azimuth and Elevation CSU does not handle any interrupts or signals.

**4.4.3.7.2.6 Filter Entity State Azimuth and Elevation CSU Error Handling**tc "4.4.3.7.2.6 Filter Entity State Azimuth and Elevation CSU Error Handling"§

The Filter Entity State Azimuth and Elevation CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of FILT_ES_AZ_AND_EL to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Filter Entity State Azimuth and Elevation CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.4.3.7.2.7 Filter Entity State Azimuth and Elevation CSU Use of Other Elements**tc "4.4.3.7.2.7 Filter Entity State Azimuth and Elevation CSU Use of Other Elements"§

The Filter Entity State Azimuth and Elevation CSU does not use system service routines, global data files, or other global elements.

**4.4.3.7.2.8 Filter Entity State Azimuth and Elevation CSU Logic Flow**tc "4.4.3.7.2.8 Filter Entity State Azimuth and Elevation CSU Logic Flow"§

The logic flow of the Filter Entity State Azimuth and Elevation CSU is represented graphically by a flowchart in Figure 4.4.3.7.2.8-1. Execution of this CSU is initiated as needed by units within external CSCIs.

µ §

**Figure 4.4.3.7.2.8-1**
**Filter Entity State Azimuth and Elevation CSU Flowchart**tc "4.4.3.7.2.8-1 Filter Entity State Azimuth and Elevation CSU Flowchart" \f f§
**4.4.3.7.2.9 Filter Entity State Azimuth and Elevation CSU Algorithms**tc "4.4.3.7.2.9 Filter Entity State Azimuth and Elevation CSU Algorithms"§

The Filter Entity State Azimuth and Elevation CSU does not utilize any algorithms.

**4.4.3.7.2.10 Filter Entity State Azimuth and Elevation CSU Limitations**tc "4.4.3.7.2.10 Filter Entity State Azimuth and Elevation CSU Limitations"§

There are no limitations or unusual features in the Filter Entity State Azimuth and Elevation CSU.

**4.4.4 Filter List By Velocity CSC (DL-CSC-4.4)**tc "4.4.4 Filter List By Velocity CSC (DL-CSC-4.4)"§

The following subparagraphs identify and describe each of the CSUs of the Filter List By Velocity CSC (DL-CSC-4.4). Figure 4.4.4-1 identifies the CSUs of this CSC. Figure 4.4.4-2 shows the execution control flow relationships of the CSUs of this CSC. Figure 4.4.4-3 shows the data flow relationships of the CSUs of this CSC. (In these figures, solid lines with no arrows indicate a hierarchical relationship, solid lines with arrows indicate data flow, and dashed lines with arrows indicate control flow. Rectangles with solid borders represent units internal to the CSC, and rectangles with dashed borders or small square boxes indicate external CSCs and CSCIs.)

µ §

**Figure 4.4.4-1**
**Filter List By Velocity CSC Computer Software Units**tc "4.4.4-1 Filter List By Velocity CSC Computer Software Units" \f f§

μ §

**Figure 4.4.4-2**
**Execution Control of the Filter List By Velocity CSC**tc "4.4.4-2 Execution Control of the Filter List By Velocity CSC " \f f§

μ §

**Figure 4.4.4-3**
**Filter List By Velocity CSC Data Flow**tc "4.4.4-3 Filter List By Velocity CSC Data Flow" \f f §

### 4.4.4.1 Filter Entity State Minimum Velocity CSUtc "4.4.4.1 Filter Entity State Minimum Velocity CSU"§

The Filter Entity State Minimum Velocity CSU (DL-CSU-4.4.1) evaluates each PDU within an input list of Entity State PDUs, and removes from the list those PDUs which contain a velocity vector which has a magnitude that is less than the specified minimum velocity threshold. The following subparagraphs provide the design information for this CSU.

#### 4.4.4.1.1 Filter Entity State Minimum Velocity CSU Design Specification/Constraintstc "4.4.4.1.1 Filter Entity State Minimum Velocity CSU Design Specification/Constraints"§

Table 4.4.4.1.1-1 identifies the requirements that are satisfied or partially satisfied by the Filter Entity State Minimum Velocity CSU. There are no design constraints for this CSU.

**Table 4.4.4.1.1-1**
**Requirements Allocated to the Filter Entity State Minimum Velocity CSU**tc " 4.4.4.1.1-1 Requirements Allocated to the Filter Entity State Minimum Velocity CSU" \f t§

| Requirement ID | DL SRS Paragraph |
|---|---|
| DL-R-24 | 3.2.4.4.1 |

**4.4.4.1.2 Filter Entity State Minimum Velocity CSU Design**tc "4.4.4.1.2 Filter Entity State Minimum Velocity CSU Design"§

The following subparagraphs specify the design of the Filter Entity State Minimum Velocity CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.4.4.1.2.1 Filter Entity State Minimum Velocity CSU Input/Output Data Elements**tc "4.4.4.1.2.1 Filter Entity State Minimum Velocity CSU Input/Output Data Elements"§

Table 4.4.4.1.2.1-1 identifies and states the purpose of each input and output data element to the Filter Entity State Minimum Velocity CSU.

**Table 4.4.4.1.2.1-1**
**Filter Entity State Minimum Velocity CSU I/O Data**tc " 4.4.4.1.2.1-1 Filter Entity State Minimum Velocity CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Entity State PDU List Head | Input/Output | Points to a list of pointers to Entity State PDUs. | See DL IRS, Table 3.2.2-2, DL-D-13 |
| Velocity Threshold | Input | Specifies the minimum acceptable velocity, in meters per second, for an entity in the list. | See DL IRS, Table 3.2.2-2, DL-D-07 |

**4.4.4.1.2.2 Filter Entity State Minimum Velocity CSU Local Data Elements**tc "4.4.4.1.2.2 Filter Entity State Minimum Velocity CSU Local Data Elements"§

The Filter Entity State Minimum Velocity CSU does not utilize any local data elements.

**4.4.4.1.2.3 Filter Entity State Minimum Velocity CSU Global Data Elements**tc "4.4.4.1.2.3 Filter Entity State Minimum Velocity CSU Global Data Elements"§

The Filter Entity State Minimum Velocity CSU does not utilize any global data elements.

**4.4.4.1.2.4 Filter Entity State Minimum Velocity CSU Local and Shared Data Structures**tc "4.4.4.1.2.4 Filter Entity State Minimum Velocity CSU Local and Shared Data Structures"§

The Filter Entity State Minimum Velocity CSU does not implement any local or shared data structures.

**4.4.4.1.2.5 Filter Entity State Minimum Velocity CSU Interrupts and Signals**tc "4.4.4.1.2.5 Filter Entity State Minimum Velocity CSU Interrupts and Signals"§

The Filter Entity State Minimum Velocity CSU does not handle any interrupts or signals.

**4.4.4.1.2.6 Filter Entity State Minimum Velocity CSU Error Handling**tc "4.4.4.1.2.6 Filter Entity State Minimum Velocity CSU Error Handling"§

The Filter Entity State Minimum Velocity CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of FILT_MIN_VEL to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Filter Entity State Minimum Velocity CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.4.4.1.2.7 Filter Entity State Minimum Velocity CSU Use of Other Elements**tc "4.4.4.1.2.7 Filter Entity State Minimum Velocity CSU Use of Other Elements"§

The Filter Entity State Minimum Velocity CSU does not use system service routines, global data files, or other global elements.

**4.4.4.1.2.8 Filter Entity State Minimum Velocity CSU Logic Flow**tc "4.4.4.1.2.8 Filter Entity State Minimum Velocity CSU Logic Flow"§

The logic flow of the Filter Entity State Minimum Velocity CSU is represented graphically by a flowchart in Figure 4.4.4.1.2.8-1. Execution of this CSU is initiated as needed by units within external CSCIs.

µ §

**Figure 4.4.4.1.2.8-1**
**Filter Entity State Minimum Velocity CSU Flowchart**tc "4.4.4.1.2.8-1 Filter Entity State Minimum Velocity CSU Flowchart" \f f§

**4.4.4.1.2.9 Filter Entity State Minimum Velocity CSU Algorithms**tc "4.4.4.1.2.9 Filter Entity State Minimum Velocity CSU Algorithms"§

The Filter Entity State Minimum Velocity CSU does not utilize any algorithms.

**4.4.4.1.2.10 Filter Entity State Minimum Velocity CSU Limitations**tc "4.4.4.1.2.10 Filter Entity State Minimum Velocity CSU Limitations"§

There are no limitations or unusual features in the Filter Entity State Minimum Velocity CSU.

**4.4.4.2 Filter Entity State Maximum Velocity CSU**tc "4.4.4.2 Filter Entity State Maximum Velocity CSU"§

The Filter Entity State Maximum Velocity CSU (DL-CSU-4.5.1) evaluates each PDU within an input list of Entity State PDUs, and removes from the list those PDUs which contain a velocity vector which has a magnitude that is greater than the specified maximum velocity threshold. The following subparagraphs provide the design information for this CSU.

**4.4.4.2.1 Filter Entity State Maximum Velocity CSU Design Specification/Constraints**tc "4.4.4.2.1 Filter Entity State Maximum Velocity CSU Design Specification/Constraints"§

Table 4.4.4.2.1-1 identifies the requirements that are satisfied or partially satisfied by the Filter Entity State Maximum Velocity CSU. There are no design constraints for this CSU.

**Table 4.4.4.2.1-1**
**Requirements Allocated to the Filter Entity State Maximum Velocity CSU**tc " 4.4.4.2.1-1 Requirements Allocated to the Filter Entity State Maximum Velocity CSU" \f t§

| **Requirement ID** | **DL SRS Paragraph** |
|---|---|
| DL-R-25 | 3.2.4.5.1 |

**4.4.4.2.2 Filter Entity State Maximum Velocity CSU Design**tc "4.4.4.2.2 Filter Entity State Maximum Velocity CSU Design"§

The following subparagraphs specify the design of the Filter Entity State Maximum Velocity CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.4.4.2.2.1 Filter Entity State Maximum Velocity CSU Input/Output Data Elements**tc "4.4.4.2.2.1 Filter Entity State Maximum Velocity CSU Input/Output Data Elements"§

Table 4.4.4.2.2.1-1 identifies and states the purpose of each input and output data element to the Filter Entity State Maximum Velocity CSU.

**Table 4.4.4.2.2.1-1**
**Filter Entity State Maximum Velocity CSU I/O Data**tc " 4.4.4.2.2.1-1 Filter Entity State Maximum Velocity CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Entity State PDU List Head | Input/Output | Points to a list of pointers to Entity State PDUs. | See DL IRS, Table 3.2.2-2, DL-D-13 |
| Velocity Threshold | Input | Specifies the maximum acceptable velocity, in meters per second, for an entity in the list. | See DL IRS, Table 3.2.2-2, DL-D-07 |

**4.4.4.2.2.2 Filter Entity State Maximum Velocity CSU Local Data Elements**tc "4.4.4.2.2.2 Filter Entity State Maximum Velocity CSU Local Data Elements"§

The Filter Entity State Maximum Velocity CSU does not utilize any local data elements.

**4.4.4.2.2.3 Filter Entity State Maximum Velocity CSU Global Data Elements**tc "4.4.4.2.2.3 Filter Entity State Maximum Velocity CSU Global Data Elements"§

The Filter Entity State Maximum Velocity CSU does not utilize any global data elements.

**4.4.4.2.2.4 Filter Entity State Maximum Velocity CSU Local and Shared Data Structures**tc "4.4.4.2.2.4 Filter Entity State Maximum Velocity CSU Local and Shared Data Structures"§

The Filter Entity State Maximum Velocity CSU does not implement any local or shared data structures.

**4.4.4.2.2.5 Filter Entity State Maximum Velocity CSU Interrupts and Signals**tc "4.4.4.2.2.5 Filter Entity State Maximum Velocity CSU Interrupts and Signals"§

The Filter Entity State Maximum Velocity CSU does not handle any interrupts or signals.

**4.4.4.2.2.6 Filter Entity State Maximum Velocity CSU Error Handling**tc "4.4.4.2.2.6 Filter Entity State Maximum Velocity CSU Error Handling"§

The Filter Entity State Maximum Velocity CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of FILT_ES_MAX_VEL to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Filter Entity State Maximum Velocity CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.4.4.2.2.7 Filter Entity State Maximum Velocity CSU Use of Other Elements**tc "4.4.4.2.2.7 Filter Entity State Maximum Velocity CSU Use of Other Elements"§

The Filter Entity State Maximum Velocity CSU does not use system service routines, global data files, or other global elements.

**4.4.4.2.2.8 Filter Entity State Maximum Velocity CSU Logic Flow**tc "4.4.4.2.2.8 Filter Entity State Maximum Velocity CSU Logic Flow"§

The logic flow of the Filter Entity State Maximum Velocity CSU is represented graphically by a flowchart in Figure 4.4.4.2.2.8-1. Execution of this CSU is initiated as needed by units within external CSCIs.

µ §

**Figure 4.4.4.2.2.8-1**
**Filter Entity State Maximum Velocity CSU Flowchart**tc "4.4.4.2.2.8-1 Filter Entity State Maximum Velocity CSU Flowchart" \f f§

**4.4.4.2.2.9 Filter Entity State Maximum Velocity CSU Algorithms**tc "4.4.4.2.2.9 Filter Entity State Maximum Velocity CSU Algorithms"§

The Filter Entity State Maximum Velocity CSU does not utilize any algorithms.

**4.4.4.2.2.10 Filter Entity State Maximum Velocity CSU Limitations**tc "4.4.4.2.2.10 Filter Entity State Maximum Velocity CSU Limitations"§

There are no limitations or unusual features in the Filter Entity State Maximum Velocity CSU.

**4.5 Hashing Algorithms CSC (DL-CSC-5)**tc "4.5 Hashing Algorithms CSC (DL-CSC-5)"§

The following subparagraphs identify and describe each of the CSUs of the Hashing Algorithms CSC (DL-CSC-5). Figure 4.5-1 identifies the CSUs of this CSC. Figure 4.5-2 shows the execution control flow relationships of the CSUs of this CSC. Figure 4.5-3 shows the data flow relationships of the CSUs of this CSC. (In these figures, solid lines with no arrows indicate a hierarchical relationship, solid lines with arrows indicate data flow, and dashed lines with arrows indicate control flow. Rectangles with solid borders represent units internal to the CSC, and rectangles with dashed borders external CSCs and CSCIs.)

µ §

**Figure 4.5-1**
**Hashing Algorithms CSC Computer Software Units**tc "4.5-1 Hashing Algorithms CSC Computer Software Units " \f f§

µ §

**Figure 4.5-2**
**Hashing Algorithms CSC Execution Control**tc "4.6-2 Smoothing Algorithms CSC Execution Control" \f f§

µ **§**

**Figure 4.5-3**
**Hashing Algorithms CSC Data Flow**tc "4.6-3 Smoothing Algorithms CSC Data Flow" \f f §

### 4.5.1 Initialize Hash Table CSUtc "4.5.1 Initialize Hash Table CSU"§

The Initialize Hash Table CSU (DL-CSU-5.1) dynamically allocates a hash table, the length of which is specified by an input parameter, for the storage of the last processed Entity State PDU. If a hash table already exists, a new hash table will be allocated, using the new length that is passed in, and all current hash table entries will be re-hashed into the new structure.  The following subparagraphs provide the design information for this CSU.

#### **4.5.1.1 Initialize Hash Table Design Specification/Constraints**tc "4.5.1.1 Initialize Hash Table Design Specification/Constraints"§

There are no design constraints for this CSU.

#### **4.5.1.2 Initialize Hash Table Design**tc "4.5.1.2 Initialize Hash Table Design"§

The following subparagraphs specify the design of the Initialize Hash Table CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, data conversion, logic flow, algorithms, limitations, and use of other elements.

#### **4.5.1.2.1 Initialize Hash Table Input/Output Data Elements**tc "4.5.1.2.1 Initialize Hash Table Input/Output Data Elements"§

Table 4.5.1.2.1-1 identifies and states the purpose of each input and output data element to the Initialize Hash Table CSU.

**Table 4.5.1.2.1-1**
**Initialize Hash Table CSU I/O Data**tc " 4.5.1.2.1-1 Initialize Hash Table CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Hash Table Length | Input | Represents the number of hash table cells to be available. | See DL IRS, Table 3.2.2-1, DL-D-15 |
| Entity State History | Input/Output | A pointer to the hash table that contains the last processed Entity State PDU for each entity in the simulation. | See DL IRS, Table 3.2.2-1, |
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |

**4.5.1.2.2 Initialize Hash Table Local Data Elements**tc "4.5.1.2.2Initialize Hash Table Local Data Elements"§

Table 4.5.1.2.2-1 identifies and describes each data element that originates within the Initialize Hash Table CSU and which is not used by any other CSU.

**Table 4.6.2.2.2-1**
**Initialize Hash Table CSU Local Data Descriptions** tc " 4.6.2.2.2-1 Initialize Hash Table CSU Local Data Descriptions" \f t§

| Name | Description | Data Type |
|---|---|---|
| Temporary Node | Defines a temporary node for the hash table which will be used to rehash the old table into a new table of a larger size. | See DL IRS, Table 3.2.2-2, |

**4.5.1.2.3  Initialize Hash Table Global Data Elements**tc "4.5.1.2.3	Initialize Hash Table Global Data Elements"§

The Initialize Hash Table CSU does not utilize any global data elements.

**4.5.1.2.4  Initialize Hash Table Local and Shared Data Structures**tc "4.5.1.2.4	Initialize Hash Table Local and Shared Data Structures"§

The Initialize Hash Table CSU does not implement any local or shared data structures.

**4.5.1.2.5  Initialize Hash Table Interrupts and Signals**tc "4.5.1.2.5	Initialize Hash Table Interrupts and Signals"§

The Initialize Hash Table CSU does not handle any interrupts or signals.

**4.5.1.2.6  Initialize Hash Table Error Handling**tc "4.5.1.2.6	Initialize Hash Table Error Handling"§

The Initialize Hash Table CSU handles run-time errors via the use of a status code.  An Ada exception handler will trap error conditions which originate in this unit, and assign a value of HASH_INIT_FAILURE to the element called Status.  If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS.

**4.5.1.2.7  Initialize Hash Table Use of Other Elements**tc "4.5.1.2.7	Initialize Hash Table Use of Other Elements"§

The Initialize Hash Table CSU does not use system service routines, global data files, or other global elements.

**4.5.1.2.8  Initialize Hash Table Logic Flow**tc "4.5.1.2.8	Initialize Hash Table Logic Flow"§

The logic flow of the Initialize Hash Table CSU is represented graphically by a flowchart.  Figures 4.5.1.2.8-1 through 4.5.1.2.8-3 depict the sequence of logical operations comprising this CSU.  Execution of this CSU is initiated as needed by units within external CSCI's, however it should be invoked as sparingly as possible after the first time due to the amount of data processing involved in transferring the contents of the hash table into a new hash table.

**Figure 4.5.1.2.8-1**
**Initialize Hash Table CSU Flowchart (Part 1)**tc "4.5.1.2.8-1	Initialize Hash Table CSU Flowchart (Part 1)" \f f§

**Figure 4.5.1.2.8-2**
**Initialize Hash Table CSU Flowchart (Part 2)**tc "4.5.1.2.8-2	Initialize Hash Table CSU Flowchart (Part 2)" \f f§

**Figure 4.5.1.2.8-3**
**Initialize Hash Table CSU Flowchart (Part 3)**tc "4.5.1.2.8-3 Initialize Hash Table CSU Flowchart (Part 3)" \f f§

**4.5.1.2.9 Initialize Hash Table Algorithms**tc "4.5.1.2.9 Initialize Hash Table Algorithms"§

The Initialize Hash Table CSU does not utilize any algorithms.

**4.5.1.2.10 Initialize Hash Table Limitations**tc "4.5.1.2.10 Initialize Hash Table Limitations"§

There are no limitations or unusual features in the Initialize Hash Table CSU.

**4.5.2 Calculate Hash Table Address CSU**tc "4.5.2 Calculate Hash Table Address CSU"§

The Calculate Hash Table Address CSU (DL-CSU-5.2) calculates a hash table address based on the size of the hash table, using the Site ID, Application ID, and Entity ID that are associated with a DIS entity. The following subparagraphs provide the design information for this CSU.

#### 4.5.2.1 Calculate Hash Table Address Design Specification/Constraintstc "4.5.2.1 Calculate Hash Table Address Design Specification/Constraints"§

There are no design constraints for this CSU.

#### 4.5.2.2 Calculate Hash Table Address Designtc "4.5.2.2 Calculate Hash Table Address Design"§

The following subparagraphs specify the design of the Calculate Hash Table Address CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, data conversion, logic flow, algorithms, limitations, and use of other elements.

#### 4.5.2.2.1 Calculate Hash Table Address Input/Output Data Elementstc "4.5.2.2.1 Calculate Hash Table Address Input/Output Data Elements"§

Table 4.5.2.2.1-1 identifies and states the purpose of each input and output data element to the Calculate Hash Table Address CSU.

**Table 4.5.2.2.1-1**
**Calculate Hash Table Address CSU I/O Data**tc " 4.5.2.2.1-1 Calculate Hash Table Address CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Hash Table Length | Input | Represents the number of hash table cells to be available. | See DL IRS, Table 3.2.2-1, DL-D-15 |
| Entity Identifier | Input | Represents the unique identifier for a DIS entity, consisting of the elements Site ID, Application ID, and Entity ID. | See DL IRS, Table 3.2.2-1, DL-D-02 |
| Table Address | Output | Represents the cell within the hash table that corresponds to the entity (via the hash keys from the Entity Identifier). | See DL IRS, Table 3.2.2-1, DL-D-17 |
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |

**4.5.2.2.2 Calculate Hash Table Address Local Data Elements**tc "4.5.2.2.2 Calculate Hash Table Address Local Data Elements"§

The Calculate Hash Table Address CSU does not utilize any local data elements.

**4.5.2.2.3 Calculate Hash Table Address Global Data Elements**tc "4.5.2.2.3 Calculate Hash Table Address Global Data Elements"§

The Calculate Hash Table Address CSU does not utilize any global data elements.

**4.5.2.2.4 Calculate Hash Table Address Local and Shared Data Structures**tc "4.5.2.2.4 Calculate Hash Table Address Local and Shared Data Structures"§

The Calculate Hash Table Address CSU does not implement any local or shared data structures.

**4.5.2.2.5 Calculate Hash Table Address Interrupts and Signals**tc "4.5.2.2.5 Calculate Hash Table Address Interrupts and Signals"§

The Calculate Hash Table Address CSU does not handle any interrupts or signals.

**4.5.2.2.6 Calculate Hash Table Address Error Handling**tc "4.5.2.2.6 Calculate Hash Table Address Error Handling"§

The Calculate Hash Table Address CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of ADDRESS_FAILURE to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS.

**4.5.2.2.7 Calculate Hash Table Address Use of Other Elements**tc "4.5.2.2.7 Calculate Hash Table Address Use of Other Elements"§

The Calculate Hash Table Address CSU does not use system service routines, global data files, or other global elements.

**4.5.2.2.8 Calculate Hash Table Address Logic Flow**tc "4.5.2.2.8 Calculate Hash Table Address Logic Flow"§

The logic flow of the Calculate Hash Table Address CSU is represented graphically by a flowchart. Figure 4.5.2.2.8-1 depicts the sequence of logical operations comprising this CSU. Execution of this CSU is initiated by the Initialize Hash Table CSU (DL-CSU-5.1) in the event of transferring the contents of one hash into another hash table, by the Get Item CSU (DL-CSU-5.3) when retrieving information from the table and by the Put Item CSU (DL-CSU-5.4) when storing information to the table.

**Figure 4.5.2.2.8-1**
**Calculate Hash Table Address CSU Flowchart**tc "4.5.2.2.8-1 Calculate Hash Table Address CSU Flowchart" \f f§

### 4.5.2.2.9 Calculate Hash Table Address Algorithms
tc "4.5.2.2.9Calculate Hash Table Address Algorithms"§

The Calculate Hash Table Address CSU uses the following algorithm to determine a hash table address:

Equation (a)

Table Address = ((((Site ID * 10) mod Hash Table Length ) +
((Application ID * 10) mod Hash Table Length)) +
((Entity ID) mod Hash Table Length))

where:

| | |
|---|---|
| Site ID | Entity State PUD DIS site identifier |
| Application ID | Entity State PDU DIS application identifier |
| Entity ID | Entity State PDU DIS entity identifier |
| Hash Table Length | The number of cells in the hash table |

#### 4.5.2.2.10 Calculate Hash Table Address Limitationstc "4.5.2.2.10 Calculate Hash Table Address Limitations"§

There are no limitations or unusual features in the Calculate Hash Table Address CSU.

### 4.5.3 Get Item CSUtc "4.5.3Get Item CSU"§

The Get Item CSU (DL-CSU-5.3) retrieves the pointer to the desired Entity State PDU in the hash table. The entity described by the Entity State PDU is identified by the combined elements Site ID, Application ID, and Entity ID (these three elements form the Entity Identifier). The following subparagraphs provide the design information for this CSU.

**4.5.3.1 Get Item Design Specification/Constraints**tc "4.5.3.1 Get Item Design Specification/Constraints"§

There are no design constraints for this CSU.

**4.5.3.2 Get Item Design**tc "4.5.3.2 Get Item Design"§

The following subparagraphs specify the design of the Get Item CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, data conversion, logic flow, algorithms, limitations, and use of other elements.

**4.5.3.2.1 Get Item Input/Output Data Elements**tc "4.5.3.2.1 Get Item Input/Output Data Elements"§

Table 4.5.3.2.1-1 identifies and states the purpose of each input and output data element to the Get Item CSU.

**Table 4.5.3.2.1-1**
**Get Item CSU I/O Data**tc " 4.5.3.2.1-1 Get Item CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Entity Identifier | Input | Represents the unique identifier for a DIS entity, consisting of the elements Site ID, Application ID, and Entity ID. | See DL IRS, Table 3.2.2-1, DL-D-02 |
| Hash Table Length | Input | The number of cells in the hash table | See DL IRS, Table 3.2.2-1, DL-D-02 |
| Last Entity State PDU | Output | A pointer to the desired Entity State PDU. | See DL IRS, Table 3.2.2-1, |
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |

**4.5.3.2.2 Get Item Local Data Elements**tc "4.5.3.2.2 Get Item Local Data Elements"§

Table 4.5.3.2.2-1 identifies and describes each data element that originates within the Get Item CSU and which is not used by any other CSU.

**Table 4.5.3.2.2-1**
**Get Item Local Data Descriptions**tc " 4.7.1.1.2.2-1 Local Data Descriptions for the Sort List CSU" \f t§

| Name | Description | Data Type |
|---|---|---|
| Temporary Pointer | A pointer that will be used to traverse the list of Entity State PDUs that hashed to the same address. | See DL IRS, Table 3.2.2-2, DL-D-11 |

#### **4.5.3.2.3 Get Item Global Data Elements**tc "4.5.3.2.3 Get Item Global Data Elements"§

The Get Item CSU does not utilize any global data elements.

#### **4.5.3.2.4 Get Item Local and Shared Data Structures**tc "4.5.3.2.4 Get Item Local and Shared Data Structures"§

The Get Item CSU does not implement any local or shared data structures.

#### **4.5.3.2.5 Get Item Interrupts and Signals**tc "4.5.3.2.5 Get Item Interrupts and Signals"§

The Get Item CSU does not handle any interrupts or signals.

#### **4.5.3.2.6 Get Item Error Handling**tc "4.5.3.2.6 Get Item Error Handling"§

The Get Item CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of GET_ITEM_FAILURE to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS.

#### **4.5.3.2.7 Get Item Use of Other Elements**tc "4.5.3.2.7 Get Item Use of Other Elements"§

The Get Item CSU does not use system service routines, global data files, or other global elements.

**4.5.3.2.8  Get Item Logic Flow**tc "4.5.3.2.8 Get Item Logic Flow"§

The logic flow of the Get Item CSU is represented graphically by a flowchart.  Figures 4.5.3.2.8-1 and 4.5.3.2.8-2 depict the sequence of logical operations comprising this CSU.  Execution of this CSU is initiated by the Rate Limiter CSU (DL-CSU-6.1) and the Alpha-Beta Filter CSU (DL-CSU-6.2) for the purpose of smoothing new values for entity position and velocity based on the time history of the entity.

**Figure 4.5.3.2.8-1**
**Get Item CSU Flowchart (Part 1)**tc "4.5.3.2.8-1    Get Item CSU Flowchart (Part 1)" \f f§

**Figure 4.5.3.2.8-2**
**Get Item CSU Flowchart (Part 2)**tc "4.5.3.2.8-2    Get Item CSU Flowchart (Part 2)" \f f§

**4.5.3.2.9 Get Item Algorithms**tc "4.5.3.2.9 Get Item Algorithms"§

The Get Item CSU does not utilize any algorithms.

**4.5.3.2.10 Get Item Limitations**tc "4.5.3.2.10 Get Item Limitations"§

There are no limitations or unusual features in the Get Item CSU.

**4.5.4 Put Item CSU**tc "4.5.4 Put Item CSU"§

The Put Item CSU (DL-CSU-5.5) inserts a DIS Entity State PDU into the hash table. The following subparagraphs provide the design information for this CSU.

**4.5.4.1 Put Item Design Specification/Constraints**tc "4.5.4.1 Put Item Design Specification/Constraints"§

There are no design constraints for this CSU.

**4.5.4.2 Put Item Design**tc "4.5.4.2 Put Item Design"§

The following subparagraphs specify the design of the Put Item CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, data conversion, logic flow, algorithms, limitations, and use of other elements.

**4.5.4.2.1 Put Item Input/Output Data Elements**tc "4.5.4.2.1 Put Item Input/Output Data Elements"§

Table 4.5.4.2.1-1 identifies and states the purpose of each input and output data element to the Put Item CSU.

**Table 4.5.4.2.1-1**
**Put Item CSU I/O Data**tc " 4.5.4.2.1-1 Put Item CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Entity State PDU | Input | Represents the information about an entity that is to be stored in the hash table (includes the Entity Identifier information needed for hashing). | See DL IRS, Table 3.2.2-1, DL-D-10 |
| Hash Table Length | Input | Represents the number of hash table cells to be available. | See DL IRS, Table 3.2.2-1, DL-D-15 |
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |

**4.5.4.2.2 Put Item Local Data Elements**tc "4.5.4.2.2 Put Item Local Data Elements"§

The Put Item CSU does not utilize any local data elements.

**4.5.4.2.3 Put Item Global Data Elements**tc "4.5.4.2.3 Put Item Global Data Elements"§

The Put Item CSU does not utilize any global data elements.

**4.5.4.2.4 Put Item Local and Shared Data Structures**tc "4.5.4.2.4 Put Item Local and Shared Data Structures"§

Table 4.5.4.2.2-1 identifies and describes each data element that originates within the Put Item CSU and which is not used by any other CSU.

**Table 4.5.4.2.2-1**
**Put Item Local Data Descriptions**tc " 4.7.1.1.2.2-1Local Data Descriptions for the Sort List CSU" \f t§

| Name | Description | Data Type |
|---|---|---|
| Temporary Pointer | A pointer that will be used to traverse the list of Entity State PDUs that hashed to the same address. | See DL IRS, Table 3.2.2-2, DL-D-11 |

**4.5.4.2.5 Put Item Interrupts and Signals**tc "4.5.4.2.5 Put Item Interrupts and Signals"§

The Put Item CSU does not handle any interrupts or signals.

**4.5.4.2.6 Put Item Error Handling**tc "4.5.4.2.6 Put Item Error Handling"§

The Get Item CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of PUT_ITEM_FAILURE to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS.

**4.5.4.2.7 Put Item Use of Other Elements**tc "4.5.4.2.8 Put Item Use of Other Elements"§

The Put Item CSU does not use system service routines, global data files, or other global elements.

**4.5.4.2.8 Put Item Logic Flow**tc "4.5.4.2.8 Put Item Logic Flow"§

The logic flow of the Put Item CSU is represented graphically by a flowchart. Figures 4.5.4.2.8-1 and 4.5.4.2.8-2 depict the sequence of logical operations comprising this CSU. Execution of this CSU is initiated by the Rate Limiter CSU (DL-CSU-6.1) and the Alpha-Beta Filter CSU (DL-CSU-6.2) for the purpose of smoothing new values for entity position and velocity based on the time history of the entity.

**Figure 4.5.4.2.8-1**
**Put Item CSU Flowchart (Part 1)**tc "4.5.4.2.8-1 Put Item CSU Flowchart (Part 1)" \f f§

**Figure 4.5.4.2.8-2**
**Put Item CSU Flowchart (Part 2)**tc "4.5.4.2.8-1 Put Item CSU Flowchart (Part 2)" \f f§

**4.5.4.2.9 Put Item Algorithms**tc "4.5.4.2.9 Put Item Algorithms"§

The Put Item CSU does not utilize any algorithms.

**4.5.4.2.10 Put Item Limitations**tc "4.5.4.2.10 Put Item Limitations"§

There are no limitations or unusual features in the Put Item CSU.

**4.6 Smoothing Algorithms CSC (DL-CSC-6)**tc "4.6 Smoothing Algorithms CSC (DL-CSC-6)"§

The following subparagraphs identify and describe each of the CSUs of the Smoothing Algorithms CSC (DL-CSC-6). Figure 4.6-1 identifies the CSUs of this CSC. Figure 4.6-2 shows the execution control flow relationships of the CSUs of this CSC. Figure 4.6-3 shows the data flow relationships of the CSUs of this CSC. (In these figures, solid lines with no arrows indicate a hierarchical relationship, solid lines with arrows indicate data flow, and dashed lines with arrows indicate control flow. Rectangles with solid borders represent units internal to the CSC, and rectangles with dashed borders or small square boxes indicate external CSCs and CSCIs.)

µ §

**Figure 4.6-1**
**Smoothing Algorithms CSC Computer Software Units**tc "4.6-1 Smoothing Algorithms CSC Computer Software Units" \f f§

µ §

**Figure 4.6-2**
**Smoothing Algorithms CSC Execution Control**tc "4.6-2 Smoothing Algorithms CSC Execution Control" \f f§

µ **§**

**Figure 4.6-3**
**Smoothing Algorithms CSC Data Flow**tc "4.6-3 Smoothing Algorithms CSC Data Flow" \f f §

### 4.6.1 Smooth Entity CSU (DL-CSU-6.1)tc "4.6.1 Smooth Entity CSU (DL-CSU-6.1)"§

The Smooth Entity CSU (DL-CSU-6.1) compensates for time discrepancies and network anomalies which affect the position and speed of an entity. The Smooth Entity CSU invokes smoothing units which have been activated. The following subparagraphs provide the design information for this CSU.

#### 4.6.1.1 Smooth Entity CSU Design Specification/Constraintstc "4.6.1.1Smooth Entity CSU Design Specification/Constraints"§

Table 4.6.1.1-1 identifies the requirements that are satisfied or partially satisfied by the Smooth Entity CSU. There are no design constraints for this CSU.

**Table 4.6.1.1-1**
**Requirements Allocated to the Smooth Entity CSU**tc " 4.6.1.1-1 Requirements Allocated to the Smooth Entity CSU" \f t§

| Requirement ID | DL SRS Paragraph |
|---|---|
| DL-R-26 | 3.2.6.1 |
| DL-R-27 | 3.2.6.2 |

#### 4.6.1.2 Smooth Entity CSU Designtc "4.6.1.2 Smooth Entity CSU Design"§

The following subparagraphs specify the design of the Smooth Entity CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

##### 4.6.1.2.1 Smooth Entity CSU Input/Output Data Elementstc "4.6.1.2.1 Smooth Entity CSU Input/Output Data Elements"§

Table 4.6.1.2.1-1 identifies and states the purpose of each input and output data element to the Smooth Entity CSU.

**Table 4.6.1.2.1-1**
**Smooth Entity CSU I/O Data**tc " 4.6.1.2.1-1 Smooth Entity CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Entity State PDU | Input/Output | Holds an Entity State PDU which requires smoothing operations. | See DL IRS, Table 3.2.2-2, DL-D-01 |

**4.6.1.2.2 Smooth Entity CSU Local Data Elements**tc "4.6.1.2.2 Smooth Entity CSU Local Data Elements"§

The Smooth Entity CSU does not use any local data elements.

**4.6.1.2.3 Smooth Entity CSU Global Data Elements**tc "4.6.1.2.3Smooth Entity CSU Global Data Elements"§

Table 4.6.1.2.3-1 identifies and describes each global data element that is used by the Smooth Entity CSU.

**Table 4.6.1.2.3-1**
**Smooth Entity CSU Global Data Elements**tc " 4.6.1.2.3-1 Smooth Entity CSU Global Data Elements " \f t§

| Name | Description | Reference |
|---|---|---|
| Alpha-Beta Filter Indicator | Indicates whether the Alpha-Beta Filter is currently selected to be active. | See DL IRS, Table 3.2.2-2, DL-D-34 |
| Rate Limiter Indicator | Indicates whether the Rate Limiter is currently selected to be active. | See DL IRS, Table 3.2.2-2, DL-D-34 |

**4.6.1.2.4 Smooth Entity CSU Local and Shared Data Structures**tc "4.6.1.2.4 Smooth Entity CSU Local and Shared Data Structures"§

The Smooth Entity CSU does not implement any local or shared data structures.

**4.6.1.2.5 Smooth Entity CSU Interrupts and Signals**tc "4.6.1.2.5 Smooth Entity CSU Interrupts and Signals"§

The Smooth Entity CSU does not handle any interrupts or signals.

**4.6.1.2.6 Smooth Entity CSU Error Handling**tc "4.6.1.2.6 Smooth Entity CSU Error Handling"§

The Smooth Entity CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of SMOOTH_ENTITY to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS.

**4.6.1.2.7 Smooth Entity CSU Use of Other Elements**tc "4.6.1.2.7 Smooth Entity CSU Use of Other Elements"§

The Smooth Entity CSU does not use system service routines, global data files, or other global elements.

**4.6.1.2.8 Smooth Entity CSU Logic Flow**tc "4.6.1.2.8 Smooth Entity CSU Logic Flow"§

The logic flow of the Smooth Entity CSU is represented graphically by a flowchart in Figure 4.6.1.2.8-1. Execution of this CSU is initiated by the Filtering Routines CSC (DL-CSC-4.1) and by external CSCIs. This CSU initiates the Rate Limiter CSU (DL-CSU-6.2), the Alpha-Beta Filter CSU (DL-CSU-6.3), and the Put Item CSU (DL-CSU-5.4).

**Figure 4.6.1.2.8-1**
**Smooth Entity CSU Flowchart (Part 1)**tc "4.6.1.2.8-1	Smooth Entity CSU Flowchart (Part 1)" \f f§

**Figure 4.6.1.2.8-2**
**Smooth Entity CSU Flowchart (Part 2)**tc "4.6.1.2.8-2	Smooth Entity CSU Flowchart (Part 2)" \f f§

µ §

**Figure 4.6.1.2.8-3**
**Smooth Entity CSU Flowchart (Part 3)**tc "4.6.1.2.8-3	Smooth Entity CSU Flowchart (Part 3)" \f f§

**4.6.1.2.9 Smooth Entity CSU Algorithms**tc "4.6.1.2.9 Smooth Entity CSU Algorithms"§

The Smooth Entity CSU does not utilize any algorithms.

**4.6.1.2.10 Smooth Entity CSU Limitations**tc "4.6.1.2.10 Smooth Entity CSU Limitations"§

There are no limitations or unusual features in the Smooth Entity CSU.

**4.6.2 Rate Limiter CSU (DL-CSU-6.2)**tc "4.6.2 Rate Limiter CSU (DL-CSU-6.2)"§

The Rate Limiter CSU (DL-CSU-6.2) ensures that the position and velocity of an entity do not change at a rate greater than the maximum possible change based on the elapsed time from the last entity state update and the last known Entity State PDU position, velocity, and acceleration. This is accomplished by dead-reckoning the entity to its next predicted position and velocity and then comparing this data to the current Entity State PDU data (i. e., position and velocity). If the current Entity State PDU data exceeds the predicted data, the current Entity State PDU data is reset to the predicted data. If the acceleration changes between entity state updates, the Rate Limiter position and velocity updates will lag behind the true position and velocity, therefore, a user defined coefficient ("fudge factor") will be used to compensate for this possible lag. The following subparagraphs provide the design information for this CSU.

**4.6.2.1 Rate Limiter CSU Design Specification/Constraints**tc "4.6.2.1 Rate Limiter CSU Design Specification/Constraints"§

Table 4.6.2.1-1 identifies the requirements that are satisfied or partially satisfied by the Rate Limiter CSU. There are no design constraints for this CSU.

**Table 4.6.2.1-1**
**Requirements Allocated to the Rate Limiter CSU**tc " 4.6.2.1-1 Requirements Allocated to the Rate Limiter CSU" \f t§

| Requirement ID | DL SRS Paragraph |
|---|---|
| DL-R-26 | 3.2.6.1 |

**4.6.2.2 Rate Limiter CSU Design**tc "4.6.2.2 Rate Limiter CSU Design"§

The following subparagraphs specify the design of the Rate Limiter CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.6.2.2.1 Rate Limiter CSU Input/Output Data Elements**tc "4.6.2.2.1 Rate Limiter CSU Input/Output Data Elements"§

Table 4.6.2.2.1-1 identifies and states the purpose of each input and output data element to the Rate Limiter CSU.

**Table 4.6.2.2.1-1**
**Rate Limiter CSU I/O Data**tc " 4.6.2.2.1-1 Rate Limiter CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Rate Limiter Coefficient | Input | Used to smooth the rate limiter position and velocity data. | See DL IRS, Table 3.2.2-2, |
| Entity State PDU | Input/Output | Holds an Entity State PDU which requires the rate limiting operation. | See DL IRS, Table 3.2.2-2, DL-D-01 |
| Previous Entity State PDU | Input | Contains the previous Entity State PDU for the entity described by the input Entity State PDU. | See DL IRS, Table 3.2.2-2, DL-D-01 |

**4.6.2.2.2 Rate Limiter CSU Local Data Elements**tc "4.6.2.2.2 Rate Limiter CSU Local Data Elements"§

Table 4.6.2.2.2-1 identifies and describes each data element that originates within the Rate Limiter CSU and which is not used by any other CSU.

**Table 4.6.2.2.2-1**
**Rate Limiter CSU Local Data Descriptions** tc " 4.6.2.2.2-1 Rate Limiter CSU Local Data Descriptions" \f t§

| Name | Description | Data Type |
|---|---|---|
| PDU History Pointer | Points to the beginning of the list of previous Entity State PDUs that are stored as a time history (Entity State PDU History) for a specific entity. (This CSU uses only the first PDU in the history.) | See DL IRS, Table 3.2.2-2, |
| Limited Entity State PDU | Contains the results of dead-reckoning the Previous Entity State PDU. This PDU contains the "limits" for position and velocity with respect to the entity. | See DL IRS, Table 3.2.2-2, DL-D-01 |

**4.6.2.2.3 Rate Limiter CSU Global Data Elements**tc "4.6.2.2.3 Rate Limiter CSU Global Data Elements"§

The Rate Limiter CSU does not utilize any global data elements.

**4.6.2.2.4 Rate Limiter CSU Local and Shared Data Structures**tc "4.6.2.2.4 Rate Limiter CSU Local and Shared Data Structures"§

The Rate Limiter CSU does not implement any local or shared data structures.

**4.6.2.2.5 Rate Limiter CSU Interrupts and Signals**tc "4.6.2.2.5 Rate Limiter CSU Interrupts and Signals"§

The Rate Limiter CSU does not handle any interrupts or signals.

**4.6.2.2.6 Rate Limiter CSU Error Handling**tc "4.6.2.2.6 Rate Limiter CSU Error Handling"§

The Rate Limiter CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of RATE_LIMITER to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS.

**4.6.2.2.7 Rate Limiter CSU Use of Other Elements**tc "4.6.2.2.7 Rate Limiter CSU Use of Other Elements"§

The Rate Limiter CSU does not use system service routines, global data files, or other global elements.

**4.6.2.2.8 Rate Limiter CSU Logic Flow**tc "4.6.2.2.8 Rate Limiter CSU Logic Flow"§

The logic flow of the Rate Limiter CSU is represented graphically by a flowchart in Figures 4.6.2.2.8-1 through 4.6.2.2.8-4. Execution of this CSU is initiated by the Smooth Entity CSU (DL-CSU-6.1) . This CSU initiates the Update Dead-Reckoned Position CSU (DL-CSU-2.1) and the Get Item CSU (DL-CSU-5.3).

**Figure 4.6.2.2.8-1**
**Rate Limiter CSU Flowchart (Part 1)**tc "4.6.2.2.8-1	Rate Limiter CSU Flowchart (Part 1)" \f f§

**Figure 4.6.2.2.8-2**
**Rate Limiter CSU Flowchart (Part 2)**tc "4.6.2.2.8-2 Rate Limiter CSU Flowchart (Part 2)" \f f§

**Figure 4.6.2.2.8-3**
**Rate Limiter CSU Flowchart (Part 3)**tc "4.6.2.2.8-3 Rate Limiter CSU Flowchart (Part 3)" \f f§

**Figure 4.6.2.2.8-4**
**Rate Limiter CSU Flowchart (Part 4)**tc "4.6.2.2.8-4 Rate Limiter CSU Flowchart (Part 4)" \f f§

#### 4.6.2.2.9 Rate Limiter CSU Algorithmstc "4.6.2.2.9 Rate Limiter CSU Algorithms"§

The Rate Limiter CSU requires comparisons to determine if the input Entity State PDU position and velocity vector components exceed the dead-reckoned Limited Entity State PDU position and velocity vector components. Each test corresponds to a specific operation in the logic flow (See Figures 4.6.2.2.8-2 and 4.6.2.2.8-3). The tests are designed to return a value of TRUE if the input Entity State PDU component is to be replaced by the Limited Entity State PDU value, and return a value of FALSE if no change is required. The "ABS" function is used in all of the tests to return the absolute value of the difference in positions or velocities.

**<u>Test</u>**

```
If
  ABS(component from the input Entity State PDU)>
  ABS(component from the dead-reckoned Limited Entity State PDU)
then
```

Equation (a)

Entity State PDU component =
Limited Entity State PDU component * Rate Limiter Coefficient

where:

| | |
|---|---|
| Rate Limiter Coefficient | is a user input value to compensate for the lag in position updates. |

and the components are defined to be the following record components (See Figure 4.6.2.2.8-2 and 46.2.2.8-3):

World Coordinates Record

| | |
|---|---|
| X | is the position along the X-axis in the world coordinate system |
| Y | is the position along the Y-axis in the world coordinate system |
| Z | is the position along the Z-axis in the world coordinate system |

Entity Linear Velocity Record

| | |
|---|---|
| X | is the velocity in the X-direction |
| Y | is the velocity in the Y-direction |
| Z | is the velocity in the Z-direction |

**4.6.2.2.10 Rate Limiter CSU Limitations**tc "4.6.2.2.10 Rate Limiter CSU Limitations"§

There are no limitations or unusual features in the Rate Limiter CSU.

**4.6.3 Alpha-Beta Filter CSU (DL-CSU-6.3)**tc "4.6.3 Alpha-Beta Filter CSU (DL-CSU-6.3)"§

The Alpha-Beta Filter CSU (DL-CSU-6.3) dampens the effects of rapid changes in the position of an entity. This is accomplished by weighing the new Entity State PDU position update and the last processed entity position update. The weight is provided by the user. The following subparagraphs provide the design information for this CSU.

**4.6.3.1  Alpha-Beta Filter CSU Design Specification/Constraints**tc "4.6.3.1 Alpha-Beta Filter CSU Design Specification/Constraints"§

Table 4.6.3.1-1 identifies the requirements that are satisfied or partially satisfied by the Alpha-Beta Filter CSU.  There are no design constraints for this CSU.

**Table 4.6.3.1-1**
**Requirements Allocated to the Alpha-Beta Filter CSU**tc " 4.6.3.1-1 Requirements Allocated to the Alpha-Beta Filter CSU" \f t§

| Requirement ID | DL SRS Paragraph |
|---|---|
| DL-R-27 | 3.2.6.2 |

### 4.6.3.2 Alpha-Beta Filter CSU Design

tc "4.6.3.2   Alpha-Beta Filter CSU Design"§

The following subparagraphs specify the design of the Alpha-Beta Filter CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

#### 4.6.3.2.1 Alpha-Beta Filter CSU Input/Output Data Elements

tc "4.6.3.2.1       Alpha-Beta Filter CSU Input/Output Data Elements"§

Table 4.6.3.2.1-1 identifies and states the purpose of each input and output data element to the Alpha-Beta Filter CSU.

**Table 4.6.3.2.1-1**
**Alpha-Beta Filter CSU I/O Data**tc " 4.6.3.2.1-1     Alpha-Beta Filter CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Entity State PDU | Input/Output | Entity State PDU which requires the dampening operation. | See DL IRS, Table 3.2.2-2, DL-D-11 |
| Previous Entity State PDU | Input | Contains the previous Entity State PDU for the entity described by the input Entity State PDU. | See DL IRS, Table 3.2.2-2, DL-D-01 |

**4.6.3.2.2 Alpha-Beta Filter CSU Local Data Elements**tc "4.6.3.2.2 Alpha-Beta Filter CSU Local Data Elements"§

The Alpha Beta Filter CSU does not implement any local or shared data structures.

**4.6.3.2.3 Alpha-Beta Filter CSU Global Data Elements**tc "4.6.3.2.3 Alpha-Beta Filter CSU Global Data Elements"§

Table 4.6.3.2.3-1 identifies and describes each global data element that is used by the Alpha-Beta Filter CSU.

**Table 4.6.3.2.3-1**
**Alpha-Beta Filter CSU Global Data Elements**tc " 4.6.3.2.3-1 Alpha-Beta Filter CSU Global Data Elements" \f t§

| **Name** | **Description** | **Reference** |
|---|---|---|
| Alpha-Beta Filter Coefficient | Contains the value to be used by the Alpha-Beta Filter during smoothing | See DL IRS, Table 3.2.2-2, DL-D-36 |

**4.6.3.2.4 Alpha-Beta Filter CSU Local and Shared Data Structures**tc "4.6.3.2.4 Alpha-Beta Filter CSU Local and Shared Data Structures"§

The Alpha-Beta Filter CSU does not implement any local or shared data structures.

**4.6.3.2.5 Alpha-Beta Filter CSU Interrupts and Signals**tc "4.6.3.2.5 Alpha-Beta Filter CSU Interrupts and Signals"§

The Alpha-Beta Filter CSU does not handle any interrupts or signals.

**4.6.3.2.6 Alpha-Beta Filter CSU Error Handling**tc "4.6.3.2.6 Alpha-Beta Filter CSU Error Handling"§

The Alpha-Beta Filter CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of AB_FILTER to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS.

**4.6.3.2.7 Alpha-Beta Filter CSU Use of Other Elements**tc "4.6.3.2.7 Alpha-Beta Filter CSU Use of Other Elements"§

The Alpha-Beta Filter CSU does not use system service routines, global data files, or other global elements.

**4.6.3.2.8 Alpha-Beta Filter CSU Logic Flow**tc "4.6.3.2.8 Alpha-Beta Filter CSU Logic Flow"§

The logic flow of the Alpha-Beta Filter CSU is represented graphically by a flowchart in Figure 4.6.3.2.8-1. Execution of this CSU is initiated by the Smooth Entity CSU (DL-CSU-6.1).

**Figure 4.6.3.2.8-1**
**Alpha-Beta Filter CSU Flowchart (Part 1)**tc "4.6.3.2.8-1 Alpha-Beta Filter CSU Flowchart (Part 1)" \f f§

**4.6.3.2.9 Alpha-Beta Filter CSU Algorithms**tc "4.6.3.2.9 Alpha-Beta Filter CSU Algorithms"§

The Alpha-Beta Filter CSU utilizes the following algorithms:

Equation (a)

µ §

where:

µ § Alpha is a percentage which represents the weight of the new position

µ § Beta is a percentage which represents the weight of the last position

Equation (b)

µ §
µ §
µ §

where:

are the components of the new position (Entity State PDU)

µ § are the components of the last position (Previous Entity State PDU)

µ § Alpha is a percentage which represents the weight of the new position

µ § Beta is a percentage which represents the weight of the last position

**4.6.3.2.10 Alpha-Beta Filter CSU Limitations**tc "4.6.3.2.10 Alpha-Beta Filter CSU Limitations"§

There are no limitations or unusual features in the Alpha-Beta Filter CSU.

**4.6.4 Set Alpha-Beta Filter Coefficient CSU (DL-CSU-6.4)**tc "4.6.4 Set Alpha-Beta Filter Coefficient CSU (DL-CSU-6.4)"§

The Set Alpha-Beta Filter Coefficient CSU (DL-CSU-6.4) establishes the Alpha-Beta Filter Coefficient value. this coefficient value represents a multiplier that is used to calculate a "smoothed" position update by using the coefficient to "weight" the position data from the current and previous Entity State PDU. This will allow the current position to be de-emphasized according to the past history of the entities' position update behavior. This value is input by the user. The following subparagraphs provide the design information for this CSU.

**4.6.4.1 Set Alpha-Beta Filter Coefficient CSU Design Specification/Constraints**tc "4.6.4.1 Set Alpha-Beta Filter Coefficient CSU Design Specification/Constraints"§

Table 4.6.4.1-1 identifies the requirements that are satisfied or partially satisfied by the Set Alpha-Beta Filter Coefficient CSU. There are no design constraints for this CSU.

**Table 4.6.4.1-1**
**Requirements Allocated to the Set Alpha-Beta Filter Coefficient CSU**tc " 4.6.4.1-1 Requirements Allocated to the Set Alpha-Beta Filter Coefficient CSU" \f t§

| **Requirement ID** | **DL SRS Paragraph** |
|---|---|
| DL-R-27 | 3.2.6.2 |

**4.6.4.2 Set Alpha-Beta Filter Coefficient CSU Design**tc "4.6.4.2 Set Alpha-Beta Filter Coefficient CSU Design"§

The following subparagraphs specify the design of the Set Alpha-Beta Filter Coefficient CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.6.4.2.1 Set Alpha-Beta Filter Coefficient CSU Input/Output Data Elements**tc "4.6.4.2.1 Set Alpha-Beta Filter Coefficient CSU Input/Output Data Elements"§

Table 4.6.4.2.1-1 identifies and states the purpose of each input and output data element to the Set Alpha-Beta Filter Coefficient CSU.

**Table 4.6.4.2.1-1**
**Set Alpha-Beta Filter Coefficient CSU I/O Data**tc " 4.6.4.2.1-1 Set Alpha-Beta Filter Coefficient CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Alpha-Beta Coefficient | Input | Contains the multiplier to be used "dampened" a new position update. | See DL IRS, Table 3.2.2-2, DL-D-33 |

**4.6.4.2.2 Set Alpha-Beta Filter Coefficient CSU Local Data Elements**tc "4.6.4.2.2 Set Alpha-Beta Filter Coefficient CSU Local Data Elements"§

The Set Alpha-Beta Filter Coefficient does not use any local data elements.

**4.6.4.2.3 Set Alpha-Beta Filter Coefficient CSU Global Data Elements**tc "4.6.4.2.3 Set Alpha-Beta Filter Coefficient CSU Global Data Elements"§

Table 4.6.4.2.3-1 identifies and describes each global data element that is used by the Set Alpha-Beta Filter Coefficient CSU.

**Table 4.6.4.2.3-1**
**Set Alpha-Beta Filter Coefficient CSU Global Data Elements**tc " 4.6.4.2.3-1 Set Alpha-Beta Filter Coefficient CSU Global Data Elements" \f t§

| Name | Description | Reference |
|---|---|---|
| Alpha-Beta Filter Coefficient Data | Contains the value to be used by the Alpha-Beta Filter during smoothing | See DL IRS, Table 3.2.2-2, DL-D-36 |

**4.6.4.2.4 Set Alpha-Beta Filter Coefficient CSU Local and Shared Data Structures**tc "4.6.4.2.4 Set Alpha-Beta Filter Coefficient CSU Local and Shared Data Structures"§

The Set Alpha-Beta Coefficient CSU does not implement any local or shared data structures.

**4.6.4.2.5 Set Alpha-Beta Filter Coefficient CSU Interrupts and Signals**tc "4.6.4.2.5 Set Alpha-Beta Filter Coefficient CSU Interrupts and Signals"§

The Set Alpha-Beta Filter Coefficient CSU does not handle any interrupts or signals.

**4.6.4.2.6 Set Alpha-Beta Filter Coefficient CSU Error Handling**tc "4.6.4.2.6 Set Alpha-Beta Filter Coefficient CSU Error Handling"§

The Set Alpha-Beta Filter Coefficient CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of SET_AB_FILTER_COEFF to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS.

**4.6.4.2.7 Set Alpha-Beta Filter Coefficient CSU Use of Other Elements**tc "4.6.4.2.7 Set Alpha-Beta Filter Coefficient CSU Use of Other Elements"§

The Set Alpha-Beta Filter Coefficient CSU does not use system service routines, global data files, or other global elements.

**4.6.4.2.8 Set Alpha-Beta Filter Coefficient CSU Logic Flow**tc "4.6.4.2.8 Set Alpha-Beta Filter Coefficient CSU Logic Flow"§

The logic flow of the Set Alpha-Beta Filter Coefficient CSU is represented graphically via a flowchart in Figure 4.6.4.2.8-1. Execution of this CSU is initiated by external CSCIs. This CSU does not initiate any other units.

**Figure 4.6.4.2.8-1**
**Set Alpha-Beta Filter Coefficient CSU Flowchart**tc "4.6.4.2.8-1 Set Alpha-Beta Filter Coefficient CSU Flowchart" \f f§

**4.6.4.2.9 Set Alpha-Beta Filter Coefficient CSU Algorithms**tc "4.6.4.2.9 Set Alpha-Beta Filter Coefficient CSU Algorithms"§

The Set Alpha-Beta Filter Coefficient CSU does not utilize any algorithms.

**4.6.4.2.10 Set Alpha-Beta Filter Coefficient CSU Limitations**tc "4.6.4.2.10 Set Alpha-Beta Filter Coefficient CSU Limitations"§

There are no limitations or unusual features in the Set Alpha-Beta Filter Coefficient CSU.

**4.6.5 Get Alpha-Beta Filter Coefficient CSU (DL-CSU-6.5)**tc "4.6.5 Get Alpha-Beta Filter Coefficient CSU (DL-CSU-6.5)"§

The Get Alpha-Beta Filter Coefficient CSU (DL-CSU-6.5) retrieves the Alpha-Beta Filter Coefficient value. This coefficient value represents a multiplier that is used to calculate a "smoothed" position update by using the coefficient to "weight" the position data from the current and previous Entity State PDUs. This will allow the current position to be de-emphasized according to the past history of the entities' position update behavior. The following subparagraphs provide the design information for this CSU.

**4.6.5.1 Get Alpha-Beta Filter Coefficient CSU Design Specification/Constraints**tc "4.6.5.1 Get Alpha-Beta Filter Coefficient CSU Design Specification/Constraints"§

Table 4.6.5.1-1 identifies the requirements that are satisfied or partially satisfied by the Get Alpha-Beta Filter Coefficient CSU. There are no design constraints for this CSU.

**Table 4.6.5.1-1**
**Requirements Allocated to the Get Alpha-Beta Filter Coefficient CSU**tc " 4.6.5.1-1 Requirements Allocated to the Get Alpha-Beta Filter Coefficient CSU" \f t§

| **Requirement ID** | **DL SRS Paragraph** |
|---|---|
| DL-R-27 | 3.2.6.2 |

**4.6.5.2 Get Alpha-Beta Filter Coefficient CSU Design**tc "4.6.5.2 Get Alpha-Beta Filter Coefficient CSU Design"§

The following subparagraphs specify the design of the Get Alpha-Beta Filter Coefficient CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.6.5.2.1 Get Alpha-Beta Filter Coefficient CSU Input/Output Data Elements**tc "4.6.5.2.1 Get Alpha-Beta Filter Coefficient CSU Input/Output Data Elements"§

Table 4.6.5.2.1-1 identifies and states the purpose of each input and output data element to the Get Alpha-Beta Filter Coefficient CSU.

**Table 4.6.5.2.1-1**
**Get Alpha-Beta Filter Coefficient CSU I/O Data**tc " 4.6.5.2.1-1 Get Alpha-Beta Filter Coefficient CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Alpha-Beta Coefficient Array | Output | Contains the multipliers to be used with the current PDU and the PDU History | See DL IRS, Table 3.2.2-2, DL-D-36 |

**4.6.5.2.2 Get Alpha-Beta Filter Coefficient CSU Local Data Elements**tc "4.6.5.2.2 Get Alpha-Beta Filter Coefficient CSU Local Data Elements"§

The Get Alpha-Beta Filter Coefficient CSU does not use any local data elements.

**4.6.5.2.3 Get Alpha-Beta Filter Coefficient CSU Global Data Elements**tc "4.6.5.2.3 Get Alpha-Beta Filter Coefficient CSU Global Data Elements"§

Table 4.6.5.2.3-1 identifies and describes each global data element that is used by the Get Alpha-Beta Filter Coefficient CSU.

**Table 4.6.5.2.3-1**
**Get Alpha-Beta Filter Coefficient CSU Global Data Elements**tc " 4.6.5.2.3-1 Get Alpha-Beta Filter Coefficient CSU Global Data Elements" \f t§

| Name | Description | Reference |
|---|---|---|
| Alpha-Beta Filter Coefficient Data | Contains the value to be used by the Alpha-Beta Filter during smoothing | See DL IRS, Table 3.2.2-2, DL-D-36 |

**4.6.5.2.4 Get Alpha-Beta Filter Coefficient CSU Local and Shared Data Structures**tc "4.6.5.2.4 Get Alpha-Beta Filter Coefficient CSU Local and Shared Data Structures"§

The Get Alpha-Beta Filter Coefficient CSU does not implement any local or shared data structures.

**4.6.5.2.5 Get Alpha-Beta Filter Coefficient CSU Interrupts and Signals**tc "4.6.5.2.5 Get Alpha-Beta Filter Coefficient CSU Interrupts and Signals"§

The Get Alpha-Beta Filter Coefficient CSU does not handle any interrupts or signals.

**4.6.5.2.6 Get Alpha-Beta Filter Coefficient CSU Error Handling**tc "4.6.5.2.6 Get Alpha-Beta Filter Coefficient CSU Error Handling"§

The Get Alpha-Beta Filter Coefficient CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of GET_AB_FILTER_COEFF to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS.

**4.6.5.2.7 Get Alpha-Beta Filter Coefficient CSU Use of Other Elements**tc "4.6.5.2.7 Get Alpha-Beta Filter Coefficient CSU Use of Other Elements"§

The Get Alpha-Beta Filter Coefficient CSU does not use system service routines, global data files, or other global elements.

**4.6.5.2.8 Get Alpha-Beta Filter Coefficient CSU Logic Flow**tc "4.6.5.2.8 Get Alpha-Beta Filter Coefficient CSU Logic Flow"§

The logic flow of the Get Alpha-Beta Filter Coefficient CSU is represented graphically via a flowchart in Figure 4.6.5.2.8-1. Execution of this CSU is initiated by external CSCIs. This CSU does not initiate any other units.

**Figure 4.6.5.2.8-1**
**Get Alpha-Beta Filter Coefficient CSU Flowchart**tc "4.6.5.2.8-1 Get Alpha-Beta Filter Coefficient CSU Flowchart" \f f§

**4.6.5.2.9 Get Alpha-Beta Filter Coefficient CSU Algorithms**tc "4.6.5.2.9 Get Alpha-Beta Filter Coefficient CSU Algorithms"§

The Get Alpha-Beta Filter Coefficient CSU does not utilize any algorithms.

**4.6.5.2.10 Get Alpha-Beta Filter Coefficient CSU Limitations**tc "4.6.5.2.10 Get Alpha-Beta Filter Coefficient CSU Limitations"§

There are no limitations or unusual features in the Get Alpha-Beta Filter Coefficient CSU.

**4.7 Sorting Algorithms CSC (DL-CSC-7)**tc "4.7 Sorting Algorithms CSC (DL-CSC-7)"§

The Sorting Algorithms CSC consists of three sub-level CSCs: the Sort Routines CSC (DL-CSC-7.1), the Sort By Distance CSC (DL-CSC-7.2), and the Sort By Velocity CSC (DL-CSC-7.3). The following subparagraphs describe the design for each sub-level CSC.

**4.7.1 Sort Routines CSC (DL-CSC-4.7.1)**tc "4.7.1Sort Routines CSC (DL-CSC-4.7.1)"§

The following subparagraphs identify and describe each of the CSUs of the Sort Routines CSC. Figure 4.7.1-1 identifies the CSUs of this CSC. Figure 4.7.1-2 shows the execution control flow relationships of the CSUs of this CSC. Figure 4.7.1-3 shows the data flow relationships of the CSUs of this CSC. (In these figures, solid lines with no arrows indicate a hierarchical relationship, solid lines with arrows indicate data flow, and dashed lines with arrows indicate control flow. Rectangles with solid borders represent units internal to the CSC, and rectangles with dashed borders indicate external CSCs and CSCIs.)

µ §

**Figure 4.7.1-1**
**Sort Routines CSC Computer Software Units**tc "4.7.1-1  Sort Routines CSC Computer Software Units" \f f§

µ §

**Figure 4.7.1-2**
**Sort Routines CSC Execution Control**tc "4.7.1-2  Sort Routines CSC Execution Control" \f f§

µ §

**Figure 4.7.1-3**
**Sort Routines CSC Data Flow**tc "4.7.1-3    Sort Routines CSC Data Flow" \f f §

The Sort Routines CSC represents a generic software component; it may be instantiated with appropriate data types and operators to sort a variety of array structures. Table 4.7.1-1 describes the generic data types which are used in the design of this CSC. Table 4.7.1-2 describes the generic functions required to sort an object of the structure type used in the instantiation.

**Table 4.7.1-1**
**Sort Routines CSC Generic Data Types** tc " 4.7.1.-1    Sort Routines CSC Generic Data Types " \f t§

| **Name** | **Description** | **Data Type** |
|---|---|---|
| Item Type | The type of individual objects being sorted. | Private |
| Index Type | The size of the array being sorted. | Unconstrained Discrete Type |
| Items Type | The array type containing the objects to be sorted. | Array of Item, constrained by Index Type |

**Table 4.7.1-2**
**Sort Routines CSC Generic Functions**tc " 4.7.1-2 Sort Routines CSC Generic Functions" \f t§

| Name | Description |
|---|---|
| Belongs After | This function is used to determine the order of the items in the array. It returns the Boolean result of a comparison of two objects of type Item Type (see Table 4.7.1-1). This function must be defined outside the scope of the generic to specify the desired sequence (ascending or descending), and to perform comparisons of record components that could not be done within the generic routine, |

**4.7.1.1 Sort List CSU (DL-CSU-4.7.1.1)**tc "4.7.1.1 Sort List CSU (DL-CSU-4.7.1.1)"§

The Sort List CSU (DL-CSU-7.1.1) sorts a list of pointers to PDUs. The PDUs are stored in shared memory and contain the information to be used as sort criteria, including position data that is used to sort by distance. This information is used to organize the PDU pointers such that they are in either ascending order or descending order with respect to the sort criteria. The method of comparing elements in the data set is defined by a function, which is defined outside the scope of the generic, and is used in the instantiation of the Sort Routines CSC. The following subparagraphs provide the design information for this CSU.

**4.7.1.1.1 Sort List CSU Design Specification/Constraints**tc "4.7.1.1.1 Sort List CSU Design Specification/Constraints"§

Table 4.7.1.1.1-1 identifies the requirements that are satisfied or partially satisfied by the Sort List CSU. This CSU is available via the external interface of the DIS Library; however, it is compatible only with other units which are implemented using Ada.

**Table 4.7.1.1.1-1**
**Requirements Allocated to the Sort List CSU**tc " 4.7.1.1.1-1 Requirements Allocated to the Sort List CSU" \f t§

| **Requirement ID** | **DL SRS Paragraph** |
|---|---|
| DL-R-28 | 3.2.7.1.1 |

**4.7.1.1.2 Sort List CSU Design**tc "4.7.1.1.2 Sort List CSU Design"§

The Sort List CSU is designed as a generic unit which implements a binary insertion sort algorithm. The following subparagraphs specify the design of the Sort List CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.7.1.1.2.1 Sort List CSU Input/Output Data Elements**tc "4.7.1.1.2.1 Sort List CSU Input/Output Data Elements"§

Table 4.7.1.1.2.1-1 identifies and states the purpose of each input and output data element to the Sort List CSU.

**Table 4.7.1.1.2.1-1**
**Sort List CSU I/O Data**tc " 4.7.1.1.2.1-1 Sort List CSU I/O Data" \f t§

| **Data Element** | **Input/Output** | **Purpose** | **Data Type** |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Items | Input/Output | Contains the objects to be sorted. | See Table 4.7.1.-1 |

**4.7.1.1.2.2 Sort List CSU Local Data Elements**tc "4.7.1.1.2.2 Sort List CSU Local Data Elements"§

Table 4.7.1.1.2.2-1 identifies and describes each data element that originates within the Sort List CSU and which is not used by any other CSU.

**Table 4.7.1.1.2.2-1**
**Sort List CSU Local Data Descriptions**tc " 4.7.1.1.2.2-1 Sort List CSU Local Data Descriptions" \f t§

| Name | Description | Data Type |
|---|---|---|
| Temporary Item | An object of the same data type as those within the array that is to be sorted, which is used to exchange the places of two objects in the array. | Item Type (see Table 4.7.1-1) |
| Left Index | Leftmost partitioning index used in the binary search during the sort. | Index Type (see Table 4.7.1-1) |
| Middle Index | Resulting index from the division by 2 used in the binary search within the sorting algorithm. | Index Type (see Table 4.7.1-1) |
| Right Index | Rightmost partitioning index used in the binary search during the sort. | Index Type (see Table 4.7.1-1) |
| Outer Index | A looping index used to analyze each element in the array to be sorted | Index Type (see Table 4.7.1-1) |

**4.7.1.1.2.3 Sort List CSU Global Data Elements**tc "4.7.1.1.2.3 Sort List CSU Global Data Elements"§

The Sort List CSU does not utilize any global data elements.

**4.7.1.1.2.4 Sort List CSU Local and Shared Data Structures**tc "4.7.1.1.2.4 Sort List CSU Local and Shared Data Structures"§

The Sort List CSU does not implement any local or shared data structures.

**4.7.1.1.2.5 Sort List CSU Interrupts and Signals**tc "4.7.1.1.2.5 Sort List CSU Interrupts and Signals"§

The Sort List CSU does not handle any interrupts or signals.

**4.7.1.1.2.6 Sort List CSU Error Handling**tc "4.7.1.1.2.6 Sort List CSU Error Handling"§

The Sort List CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of SORT_LIST to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS.

**4.7.1.1.2.7 Sort List CSU Use of Other Elements**tc "4.7.1.1.2.7 Sort List CSU Use of Other Elements"§

The Sort List CSU does not use system service routines, global data files, or other global elements.

**4.7.1.1.2.8 Sort List CSU Logic Flow**tc "4.7.1.1.2.8 Sort List CSU Logic Flow"§

The logic flow of the Sort List CSU is represented graphically by a flowchart in Figures 4.7.1.1.2.8-1 through 4.7.1.1.2.8-3 Execution of this CSU is initiated by the Sort By Distance CSC (DL-CSC-7.2), the Sort By Velocity CSC (DL-CSC-7.3) and by external CSCIs.

**Figure 4.7.1.1.2.8-1**
**Sort List CSU Flowchart (Part 1)** tc "4.7.1.1.2.8-1 Sort List CSU Flowchart (Part 1)" \f f§

**Figure 4.7.1.1.2.8-2**
**Sort List CSU Flowchart (Part 2)** tc "4.7.1.1.2.8-2 Sort List CSU Flowchart (Part 2)" \f f§

**Figure 4.7.1.1.2.8-3**
**Sort List CSU Flowchart (Part 3)** tc "4.7.1.1.2.8-3 Sort List CSU Flowchart (Part 3)" \f f§

**4.7.1.1.2.9 Sort List CSU Algorithms**tc "4.7.1.1.2.9 Sort List CSU Algorithms"§

The Sort List CSU implements a binary insertion sort algorithm which is a variation of the simple insertion sorting algorithm. The difference is that a binary search is used to locate the insertion point when swapping elements instead of a sequential search. This binary search is accomplished by dividing the list into two parts and then determining into which part the item will be inserted. This process if repeated until the placement point is found. This implementation requires managing three indexes into the list of items: a starting point, ending point and mid point in the list. The following is the equation for calculating the mid point in the list:

Equation (a)

$$\text{Middle Index} = (\text{Left Index} + \text{Right Index})/2$$

where:

| | |
|---|---|
| Middle Index | is the mid point in the list (the item being sorted will be compared to this item to determine in which half of the list it belongs) |
| Left Index | starting point for the search of insertion point |
| Right Index | ending point for the search of insertion point |

**4.7.1.1.2.10 Sort List CSU Limitations**tc "4.7.1.1.2.10 Sort List CSU Limitations"§

There are no limitations or unusual features in the Sort List CSU.

**4.7.2 Sort By Distance CSC (DL-CSC-4.7.2)**tc "4.7.2 Sort By Distance CSC (DL-CSC-4.7.2)"§

The following subparagraphs identify and describe each of the CSUs of the Sort By Distance CSC. Figure 4.7.2-1 identifies the CSUs of this CSC. Figure 4.7.2-2 shows the execution control flow relationships of the CSUs of this CSC. Figure 4.7.2-3 shows the data flow relationships of the CSUs of this CSC. (In these figures, solid lines with no arrows indicate a hierarchical relationship, solid lines with arrows indicate data flow, and dashed lines with arrows indicate control flow. Rectangles with solid borders represent units internal to the CSC, and rectangles with dashed borders indicate external CSCs and CSCIs.)

µ §

**Figure 4.7.2-1**
**Sort By Distance CSC Computer Software Units**tc "4.7.2-1 Sort By Distance CSC Computer Software Units" \f f§
µ §

**Figure 4.7.2-2**
**Sort By Distance CSC Execution Control**tc "4.7.2-2 Sort By Distance CSC Execution Control" \f f§

µ §

**Figure 4.7.2-3**
**Sort By Distance CSC Data Flow**tc "4.7.2-3 Sort By Distance CSC Data Flow" \f f §

**4.7.2.1 Sort Entity State Distance CSU (DL-CSU-7.2.1)**tc "4.7.2.1 Sort Entity State Distance CSU (DL-CSU-7.2.1)"§

The Sort Entity State Distance CSU (DL-CSU-7.2.1) sorts a list of pointers to Entity State PDUs into ascending order based on the distance between the position within the PDU and the input reference position. The following subparagraphs provide the design information for this CSU.

**4.7.2.1.1 Sort Entity State Distance CSU Design Specification/Constraints**tc "4.7.2.1.1 Sort Entity State Distance CSU Design Specification/Constraints"§

Table 4.7.2.1.1-1 identifies the requirements that are satisfied or partially satisfied by the Sort Entity State Distance CSU. There are no design constraints for this CSU.

**Table 4.7.2.1.1-1**
**Requirements Allocated to the Sort Entity State Distance CSU**tc " 4.7.2.1.1-1 Requirements Allocated to the Sort Entity State Distance CSU" \f t§

| Requirement ID | DL SRS Paragraph |
|---|---|
| DL-R-29 | 3.2.7.2.1 |

**4.7.2.1.2 Sort Entity State Distance CSU Design**tc "4.7.2.1.2 Sort Entity State Distance CSU Design"§

The following subparagraphs specify the design of the Sort Entity State Distance CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.7.2.1.2.1 Sort Entity State Distance CSU Input/Output Data Elements**tc "4.7.2.1.2.1 Sort Entity State Distance CSU Input/Output Data Elements"§

Table 4.7.2.1.2.1-1 identifies and states the purpose of each input and output data element to the Sort Entity State Distance CSU.

**Table 4.7.2.1.2.1-1**
**Sort Entity State Distance CSU I/O Data**tc " 4.7.2.1.2.1-1 Sort Entity State Distance CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Entity State PDU List Head | Input/Output | Points to a list of pointers to Entity State PDUs. | See DL IRS, Table 3.2.2-2, DL-D-13 |

#### 4.7.2.1.2.2 Sort Entity State Distance CSU Local Data Elementstc "4.7.2.1.2.2 Sort Entity State Distance CSU Local Data Elements"§

Table 4.7.2.1.2.2-1 identifies and describes each data element that originates within the Sort Entity State CSU and which is not used by any other CSU. Table 4.7.2.1.2.2-2 provides the size, units, and range for each of the local data elements. Table 4.7.2.1.2.2-3 describes the local function required by this CSU. Table 4.7.2.1.2.2-4 identifies the scope of each local data element and function.

**Table 4.7.2.1.2.2-1**
**Sort Entity State Distance CSU Local Data Descriptions**tc " 4.7.2.1.2.2-1 Sort Entity State Distance CSU Local Data Descriptions" \f t§

| Name | Description | Data Type |
|---|---|---|
| Item Count | The element which is used to store the result of counting the number of nodes in the list of Entity State PDU pointers. | See Table 4.7.2.1.2.2-2 |
| Loop Index | The element used as an index for the array processing loop. | See Table 4.7.2.1.2.2-2 |
| Entity State Pointer Array | An array containing all Entity State PDU pointers and an associated distance value. The distance is measured between the world coordinates position within each Entity State PDU and the input reference position. | See Table 4.7.2.1.2.2-2 |

**Table 4.7.2.1.2.2-2**
**Size, Units, and Range of Local Data for the Sort Entity State Distance CSU**tc " 4.7.2.1.2.2-2 Size, Units, and Range of Local Data for the Sort Entity State Distance CSU" \f t§

| Name | Size (Bits) | Units | Range |
|---|---|---|---|
| Item Count | 16 | N/A | 0..65535 |
| Loop Index | 16 | N/A | 0..65535 |
| Entity State Pointer Array | 96 * Item Count | See Table 4.7.2.1.2.4-1 | |

**Table 4.7.2.1.2.2-3**
**Sort Entity State Distance CSU Local Function Description**tc " 4.7.2.1.2.2-3 Sort Entity State Distance CSU Local Function Description" \f t§

| Name | Description | Algorithm |
|---|---|---|
| Distance From Reference Is Greater | Returns the Boolean result of comparing two distances using the "greater than" operator. | See paragraph 4.7.2.1.2.9 |

**Table 4.7.2.1.2.2-4**
**Scope Level of Local Declarations for the Sort Entity State Distance CSU**tc " 4.7.2.1.2.2-4 Scope Level of Local Declarations for the Sort Entity State Distance CSU" \f t§

| Name | Scope Level |
|---|---|
| Item Count | Sort Entity State Distance unit |
| Loop Index | Sort Pointer Array block |
| Entity State Pointer Array | Sort Pointer Array block |
| Distance From Reference Is Greater | Sort Pointer Array block |

#### 4.7.2.1.2.3 Sort Entity State Distance CSU Global Data Elements tc "4.7.2.1.2.3 Sort Entity State Distance CSU Global Data Elements"§

The Sort Entity State Distance CSU does not utilize any global data elements.

#### 4.7.2.1.2.4 Sort Entity State Distance CSU Local and Shared Data Structures tc "4.7.2.1.2.4 Sort Entity State Distance CSU Local and Shared Data Structures"§

Table 4.7.2.1.2.4-1 describes local data structures implemented by the Sort Entity State Distance CSU. This CSU does not utilize any shared data structures.

**Table 4.7.2.1.2.4-1**
**Entity State Pointer Array Data Structure**tc " 4.7.2.1.2.4-1 Entity State Pointer Array Data Structure" \f t§

| Array Bounds | Array Element Type | Units | Range |
|---|---|---|---|
| 1 .. Item Count | Sort Record | See Table 4.7.2.1.2.4-2 | |

**Table 4.7.2.1.2.4-2**
**Sort Record Data Structure**tc " 4.7.2.1.2.4-2 Sort Record Data Structure" \f t§

| Data Element | Data Type | Size (Bits) | Units | Range |
|---|---|---|---|---|
| Distance | See DL IRS, Table 3.2.2-2, DL-D-06 | | | |
| Entity State PDU List Pointer Node | See DL IRS, Table 3.2.2-2, DL-D-13 | | | |

#### 4.7.2.1.2.5 Sort Entity State Distance CSU Interrupts and Signalstc "4.7.2.1.2.5 Sort Entity State Distance CSU Interrupts and Signals"§

The Sort Entity State Distance CSU does not handle any interrupts or signals.

#### 4.7.2.1.2.6 Sort Entity State Distance CSU Error Handlingtc "4.7.2.1.2.6 Sort Entity State Distance CSU Error Handling"§

The Sort Entity State Distance CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of SORT_ES_DIST to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Sort Entity State Distance CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.7.2.1.2.7 Sort Entity State Distance CSU Use of Other Elements**tc "4.7.2.1.2.7 Sort Entity State Distance CSU Use of Other Elements"§

The Sort Entity State Distance CSU does not use system service routines, global data files, or other global elements.

**4.7.2.1.2.8 Sort Entity State Distance CSU Logic Flow**tc "4.7.2.1.2.8 Sort Entity State Distance CSU Logic Flow"§

The logic flow of the Sort Entity State Distance CSU is represented graphically by a flowchart in Figures 4.7.2.1.2.8-1 through 4.7.2.1.2.8-5. Execution of this CSU is initiated by external CSCIs. The Sort Entity State Distance CSU initiates the Calculate Distance CSU (DL-CSU-3.1). This CSU also instantiates the generic Sort List CSU (DL-CSU-7.1.1) and initiates the new instance of the Sort List CSU.

µ §

**Figure 4.7.2.1.2.8-1**
**Sort Entity State Distance CSU Flowchart (Part 1)**tc "4.7.2.1.2.8-1 Sort Entity State Distance CSU Flowchart (Part 1)" \f f§
µ **§**

**Figure 4.7.2.1.2.8-2**
**Sort Entity State Distance CSU Flowchart (Part 2)**tc "4.7.2.1.2.8-2 Sort Entity State Distance CSU Flowchart (Part 2)" \f f§

**Figure 4.7.2.1.2.8-3**
**Sort Entity State Distance CSU Flowchart (Part 3)**tc "4.7.2.1.2.8-3	Sort Entity State Distance CSU Flowchart (Part 3)" \f f§

µ §

**Figure 4.7.2.1.2.8-4**
**Sort Entity State Distance CSU Flowchart (Part 4)**tc "4.7.2.1.2.8-4 Sort Entity State Distance CSU Flowchart (Part 4)" \f f§

μ §

**Figure 4.7.2.1.2.8-5**
**Sort Entity State Distance CSU Flowchart (Part 5)**tc "4.7.2.1.2.8-5 Sort Entity State Distance CSU Flowchart (Part 5)" \f f§

**4.7.2.1.2.9 Sort Entity State Distance CSU Algorithms**tc "4.7.2.1.2.9 Sort Entity State Distance CSU Algorithms"§

The Sort Entity State Distance CSU contains an algorithm to compare two distance values which are components of a Sort Record structure (see Table 4.7.2.1.2.4-2). The algorithm is designed as a function, which returns the Boolean result of the comparison which uses the "greater than" operator. This function will be used in the Sort List CSU (DL-CSU-7.1.1) to sort the array of pointers to PDUs into ascending order with respect to the distance from the reference position.
**Function (A)**

```
function Distance_From_Reference_Is_Greater (Left  : in Sort_Record,
                                             Right : in Sort_Record)
  return BOOLEAN is

begin
   return (Left.Distance > Right.Distance);
end Distance_From_Reference_Is_Greater;
```

**4.7.2.1.2.10 Sort Entity State Distance CSU Limitations**tc "4.7.2.1.2.10 Sort Entity State Distance CSU Limitations"§

There are no limitations or unusual features in the Sort Entity State Distance CSU.

**4.7.2.2 Sort Fire Distance CSU (DL-CSU-7.2.2)**tc "4.7.2.2 Sort Fire Distance CSU (DL-CSU-7.2.2)"§

The Sort Fire Distance CSU (DL-CSU-7.2.2) sorts a list of pointers to Fire PDUs into ascending order based on the distance between the position within the PDU and the input reference position. The following subparagraphs provide the design information for this CSU.

**4.7.2.2.1 Sort Fire Distance CSU Design Specification/Constraints**tc "4.7.2.2.1 Sort Fire Distance CSU Design Specification/Constraints"§

Table 4.7.2.2.1-1 identifies the requirements that are satisfied or partially satisfied by the

Sort Fire Distance CSU.  There are no design constraints for this CSU.

**Table 4.7.2.2.1-1**
**Requirements Allocated to the Sort Fire Distance CSU**tc " 4.7.2.2.1-1   Requirements Allocated to the Sort Fire Distance CSU" \f t§

| **Requirement ID** | **DL SRS Paragraph** |
|---|---|
| DL-R-30 | 3.2.7.2.2 |

**4.7.2.2.2  Sort Fire Distance CSU Design**tc "4.7.2.2.2	Sort Fire Distance CSU Design"§

The following subparagraphs specify the design of the Sort Fire Distance CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.7.2.2.2.1  Sort Fire Distance CSU Input/Output Data Elements**tc "4.7.2.2.2.1	Sort Fire Distance CSU Input/Output Data Elements"§

Table 4.7.2.2.2.1-1 identifies and states the purpose of each input and output data element to the Sort Fire Distance CSU.

**Table 4.7.2.2.2.1-1**
**Sort Fire Distance CSU I/O Data**tc " 4.7.2.2.2.1-1 Sort Fire Distance CSU I/O Data" \f t§

| **Data Element** | **Input/Output** | **Purpose** | **Data Type** |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Fire PDU List Head | Input/Output | Points to a list of pointers to Fire PDUs. | See DL IRS, Table 3.2.2-2, DL-D-16 |

**4.7.2.2.2.2 Sort Fire Distance CSU Local Data Elements**tc "4.7.2.2.2.2 Sort Fire Distance CSU Local Data Elements"§

Table 4.7.2.2.2.2-1 identifies and describes each data element that originates within the Sort Fire CSU and which is not used by any other CSU. Table 4.7.2.2.2.2-2 provides the size, units, and range for each of the local data elements. Table 4.7.2.2.2.2-3 describes the local function required by this CSU. Table 4.7.2.2.2.2-4 identifies the scope of each local data element and function.

**Table 4.7.2.2.2.2-1**
**Sort Fire Distance CSU Local Data Descriptions**tc " 4.7.2.2.2.2-1 Sort Fire Distance CSU Local Data Descriptions" \f t§

| Name | Description | Data Type |
|---|---|---|
| Item Count | The element which is used to store the result of counting the number of nodes in the list of Fire PDU pointers. | See Table 4.7.2.2.2.2-2 |
| Loop Index | The element used as an index for the array processing loop. | See Table 4.7.2.2.2.2-2 |
| Fire Pointer Array | An array containing all Fire PDU pointers and an associated distance value. The distance is measured between the world coordinates position within each Fire PDU and the input reference position. | See Table 4.7.2.2.2.2-2 |

**Table 4.7.2.2.2.2-2**
**Size, Units, and Range of Local Data for the Sort Fire Distance CSU**tc " 4.7.2.2.2.2-2 Size, Units, and Range of Local Data for the Sort Fire Distance CSU" \f t§

| Name | Size (Bits) | Units | Range |
|---|---|---|---|
| Item Count | 16 | N/A | 0..65535 |
| Loop Index | 16 | N/A | 0..65535 |
| Fire Pointer Array | 96 * Item Count | See Table 4.7.2.2.2.4-1 | |

**Table 4.7.2.2.2.2-3**
**Sort Fire Distance CSU Local Function Description**tc " 4.7.2.2.2.2-3 Sort Fire Distance CSU Local Function Description" \f t§

| Name | Description | Algorithm |
|---|---|---|
| Distance From Reference Is Greater | Returns the Boolean result of comparing two distances using the "greater than" operator. | See paragraph 4.7.2.2.2.9 |

**Table 4.7.2.2.2.2-4**
**Scope Level of Local Declarations for the Sort Fire Distance CSU**tc " 4.7.2.2.2.2-4 Scope Level of Local Declarations for the Sort Fire Distance CSU" \f t§

| Name | Scope Level |
|---|---|
| Item Count | Sort Fire Distance unit |
| Loop Index | Sort Pointer Array block |
| Fire Pointer Array | Sort Pointer Array block |
| Distance From Reference Is Greater | Sort Pointer Array block |

**4.7.2.2.2.3 Sort Fire Distance CSU Global Data Elements**tc "4.7.2.2.2.3 Sort Fire Distance CSU Global Data Elements"§

The Sort Fire Distance CSU does not utilize any global data elements.

**4.7.2.2.2.4 Sort Fire Distance CSU Local and Shared Data Structures**tc "4.7.2.2.2.4 Sort Fire Distance CSU Local and Shared Data Structures"§

Table 4.7.2.2.2.4-1 describes local data structures implemented by the Sort Fire Distance CSU. This CSU does not utilize any shared data structures.

**Table 4.7.2.2.2.4-1**
**Fire Pointer Array Data Structure**tc " 4.7.2.2.2.4-1 Fire Pointer Array Data Structure" \f t§

| Array Bounds | Array Element Type | Units | Range |
|---|---|---|---|
| 1 .. Item Count | Sort Record | See Table 4.7.2.2.2.4-2 | |

**Table 4.7.2.2.2.4-2**
**Sort Record Data Structure**tc " 4.7.2.2.2.4-2 Sort Record Data Structure" \f t§

| Data Element | Data Type | Size (Bits) | Units | Range |
|---|---|---|---|---|
| Distance | See DL IRS, Table 3.2.2-2, DL-D-06 | | | |
| Fire PDU List Pointer Node | See DL IRS, Table 3.2.2-2, DL-D-16 | | | |

**4.7.2.2.2.5 Sort Fire Distance CSU Interrupts and Signals**tc "4.7.2.2.2.5 Sort Fire Distance CSU Interrupts and Signals"§

The Sort Fire Distance CSU does not handle any interrupts or signals.

**4.7.2.2.2.6 Sort Fire Distance CSU Error Handling**tc "4.7.2.2.2.6 Sort Fire Distance CSU Error Handling"§

The Sort Fire Distance CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of SORT_FIRE_DIST to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Sort Fire Distance CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.7.2.2.2.7 Sort Fire Distance CSU Use of Other Elements**tc "4.7.2.2.2.7 Sort Fire Distance CSU Use of Other Elements"§

The Sort Fire Distance CSU does not use system service routines, global data files, or other global elements.

**4.7.2.2.2.8 Sort Fire Distance CSU Logic Flow**tc "4.7.2.2.2.8 Sort Fire Distance CSU Logic Flow"§

The logic flow of the Sort Fire Distance CSU is represented graphically by a flowchart in Figures 4.7.2.2.2.8-1 through 4.7.2.2.2.8-5. Execution of this CSU is initiated by external CSCIs. The Sort Fire Distance CSU initiates the Calculate Distance CSU (DL-CSU-3.1). This CSU also instantiates the generic Sort List CSU (DL-CSU-7.1.1) and initiates the new instance of the Sort List CSU.

**Figure 4.7.2.2.2.8-1**
**Sort Fire Distance CSU Flowchart (Part 1)**tc "4.7.2.2.2.8-1 Sort Fire Distance CSU Flowchart (Part 1)" \f f§
µ **§**

**Figure 4.7.2.2.2.8-2**
**Sort Fire Distance CSU Flowchart (Part 2)**tc "4.7.2.2.2.8-2 Sort Fire Distance CSU Flowchart (Part 2)" \f f§

µ §

**Figure 4.7.2.2.2.8-3**
**Sort Fire Distance CSU Flowchart (Part 3)**tc "4.7.2.2.2.8-3 Sort Fire Distance CSU Flowchart (Part 3)" \f f§

µ §

**Figure 4.7.2.2.2.8-4**
**Sort Fire Distance CSU Flowchart (Part 4)**tc "4.7.2.2.2.8-4 Sort Fire Distance CSU Flowchart (Part 4)" \f f§

µ §

**Figure 4.7.2.2.2.8-5**

**Sort Fire Distance CSU Flowchart (Part 5)**tc "4.7.2.2.2.8-5 Sort Fire Distance CSU Flowchart (Part 5)" \f f§

**4.7.2.2.2.9 Sort Fire Distance CSU Algorithms**tc "4.7.2.2.2.9 Sort Fire Distance CSU Algorithms"§

The Sort Fire Distance CSU contains an algorithm to compare two distance values which are components of a Sort Record structure (see Table 4.7.2.2.2.4-2). The algorithm is designed as a function, which returns the Boolean result of the comparison which uses the "greater than" operator. This function will be used in the Sort List CSU (DL-CSU-7.1.1) to sort the array of pointers to PDUs into ascending order with respect to the distance from the reference position.

**<u>Function (A)</u>**

```
function Distance_From_Reference_Is_Greater (Left  : in Sort_Record,
                                             Right : in Sort_Record)
 return BOOLEAN is

begin
  return (Left.Distance > Right.Distance);
end Distance_From_Reference_Is_Greater;
```

**4.7.2.2.2.10 Sort Fire Distance CSU Limitations**tc "4.7.2.2.2.10 Sort Fire Distance CSU Limitations"§

There are no limitations or unusual features in the Sort Fire Distance CSU.

**4.7.2.3 Sort Detonation Distance CSU (DL-CSU-7.2.3)**tc "4.7.2.3 Sort Detonation Distance CSU (DL-CSU-7.2.3)"§

The Sort Detonation Distance CSU (DL-CSU-7.2.3) sorts a list of pointers to Detonation PDUs into ascending order based on the distance between the position within the PDU and the input reference position. The following subparagraphs provide the design information for this CSU.

**4.7.2.3.1 Sort Detonation Distance CSU Design Specification/Constraints**tc "4.7.2.3.1 Sort Detonation Distance CSU Design Specification/Constraints"§

Table 4.7.2.3.1-1 identifies the requirements that are satisfied or partially satisfied by the Sort Detonation Distance CSU. There are no design constraints for this CSU.

**Table 4.7.2.3.1-1**
**Requirements Allocated to the Sort Detonation Distance CSU**tc " 4.7.2.3.1-1 Requirements Allocated to the Sort Detonation Distance CSU" \f t§

| Requirement ID | DL SRS Paragraph |
|---|---|
| DL-R-31 | 3.2.7.2.3 |

**4.7.2.3.2 Sort Detonation Distance CSU Design**tc "4.7.2.3.2 Sort Detonation Distance CSU Design"§

The following subparagraphs specify the design of the Sort Detonation Distance CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.7.2.3.2.1 Sort Detonation Distance CSU Input/Output Data Elements**tc "4.7.2.3.2.1 Sort Detonation Distance CSU Input/Output Data Elements"§

Table 4.7.2.3.2.1-1 identifies and states the purpose of each input and output data element to the Sort Detonation Distance CSU.

**Table 4.7.2.3.2.1-1**
**Sort Detonation Distance CSU I/O Data**tc " 4.7.2.3.2.1-1 Sort Detonation Distance CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Detonation PDU List Head | Input/Output | Points to a list of pointers to Detonation PDUs. | See DL IRS, Table 3.2.2-2, DL-D-19 |

**4.7.2.3.2.2 Sort Detonation Distance CSU Local Data Elements**tc "4.7.2.3.2.2 Sort Detonation Distance CSU Local Data Elements"§

Table 4.7.2.3.2.2-1 identifies and describes each data element that originates within the Sort Detonation CSU and which is not used by any other CSU. Table 4.7.2.3.2.2-2 provides the size, units, and range for each of the local data elements. Table 4.7.2.3.2.2-3 describes the local function required by this CSU. Table 4.7.2.3.2.2-4 identifies the scope of each local data element and function.

**Table 4.7.2.3.2.2-1**
**Sort Detonation Distance CSU Local Data Descriptions**tc " 4.7.2.3.2.2-1 Sort Detonation Distance CSU Local Data Descriptions" \f t§

| Name | Description | Data Type |
|---|---|---|
| Item Count | The element which is used to store the result of counting the number of nodes in the list of Detonation PDU pointers. | See Table 4.7.2.3.2.2-2 |
| Loop Index | The element used as an index for the array processing loop. | See Table 4.7.2.3.2.2-2 |
| Detonation Pointer Array | An array containing all Detonation PDU pointers and an associated distance value. The distance is measured between the world coordinates position within each Detonation PDU and the input reference position. | See Table 4.7.2.3.2.2-2 |

**Table 4.7.2.3.2.2-2**
**Size, Units, and Range of Local Data for the Sort Detonation Distance CSU**tc " 4.7.2.3.2.2-2 Size, Units, and Range of Local Data for the Sort Detonation Distance CSU" \f t§

| Name | Size (Bits) | Units | Range |
|---|---|---|---|
| Item Count | 16 | N/A | 0..65535 |
| Loop Index | 16 | N/A | 0..65535 |
| Detonation Pointer Array | 96 * Item Count | See Table 4.7.2.3.2.4-1 | |

**Table 4.7.2.3.2.2-3**
**Sort Detonation Distance CSU Local Function Description**tc " 4.7.2.3.2.2-3 Sort Detonation Distance CSU Local Function Description" \f t§

| Name | Description | Algorithm |
|---|---|---|
| Distance From Reference Is Greater | Returns the Boolean result of comparing two distances using the "greater than" operator. | See paragraph 4.7.2.3.2.9 |

**Table 4.7.2.3.2.2-4**
**Scope Level of Local Declarations for the Sort Detonation Distance CSU**tc " 4.7.2.3.2.2-4 Scope Level of Local Declarations for the Sort Detonation Distance CSU" \f t§

| Name | Scope Level |
|---|---|
| Item Count | Sort Detonation Distance unit |
| Loop Index | Sort Pointer Array block |
| Detonation Pointer Array | Sort Pointer Array block |
| Distance From Reference Is Greater | Sort Pointer Array block |

**4.7.2.3.2.3 Sort Detonation Distance CSU Global Data Elements**tc "4.7.2.3.2.3Sort Detonation Distance CSU Global Data Elements"§

The Sort Detonation Distance CSU does not utilize any global data elements.

**4.7.2.3.2.4 Sort Detonation Distance CSU Local and Shared Data Structures**tc "4.7.2.3.2.4 Sort Detonation Distance CSU Local and Shared Data Structures"§

Table 4.7.2.3.2.4-1 describes local data structures implemented by the Sort Detonation Distance CSU. This CSU does not utilize any shared data structures.

**Table 4.7.2.3.2.4-1**
**Detonation Pointer Array Data Structure**tc " 4.7.2.3.2.4-1 Detonation Pointer Array Data Structure" \f t§

| Array Bounds | Array Element Type | Units | Range |
|---|---|---|---|
| 1 .. Item Count | Sort Record | See Table 4.7.2.3.2.4-2 | |

**Table 4.7.2.3.2.4-2**
**Sort Record Data Structure**tc " 4.7.2.3.2.4-2 Sort Record Data Structure" \f t§

| Data Element | Data Type | Size (Bits) | Units | Range |
|---|---|---|---|---|
| Distance | See DL IRS, Table 3.2.2-2, DL-D-06 | | | |
| Detonation PDU List Pointer Node | See DL IRS, Table 3.2.2-2, DL-D-19 | | | |

**4.7.2.3.2.5 Sort Detonation Distance CSU Interrupts and Signals**tc "4.7.2.3.2.5 Sort Detonation Distance CSU Interrupts and Signals"§

The Sort Detonation Distance CSU does not handle any interrupts or signals.

**4.7.2.3.2.6 Sort Detonation Distance CSU Error Handling**tc "4.7.2.3.2.6 Sort Detonation Distance CSU Error Handling"§

The Sort Detonation Distance CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of SORT_DETONATION_DIST to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Sort Detonation Distance CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.7.2.3.2.7 Sort Detonation Distance CSU Use of Other Elements**tc "4.7.2.3.2.7 Sort Detonation Distance CSU Use of Other Elements"§

The Sort Detonation Distance CSU does not use system service routines, global data files, or other global elements.

**4.7.2.3.2.8 Sort Detonation Distance CSU Logic Flow**tc "4.7.2.3.2.8 Sort Detonation Distance CSU Logic Flow"§

The logic flow of the Sort Detonation Distance CSU is represented graphically by a flowchart in Figures 4.7.2.3.2.8-1 through 4.7.2.3.2.8-5. Execution of this CSU is initiated by external CSCIs. The Sort Detonation Distance CSU initiates the Calculate Distance CSU (DL-CSU-3.1). This CSU also instantiates the generic Sort List CSU (DL-CSU-7.1.1) and initiates the new instance of the Sort List CSU.

µ §

**Figure 4.7.2.3.2.8-1**
**Sort Detonation Distance CSU Flowchart (Part 1)**tc "4.7.2.3.2.8-1 Sort Detonation Distance CSU Flowchart (Part 1)" \f f§
µ **§**

**Figure 4.7.2.3.2.8-2**
**Sort Detonation Distance CSU Flowchart (Part 2)**tc "4.7.2.3.2.8-2 Sort Detonation Distance CSU Flowchart (Part 2)" \f f§

µ §

**Figure 4.7.2.3.2.8-3**
**Sort Detonation Distance CSU Flowchart (Part 3)**tc "4.7.2.3.2.8-3 Sort Detonation Distance CSU Flowchart (Part 3)" \f f§

µ §

**Figure 4.7.2.3.2.8-4**
**Sort Detonation Distance CSU Flowchart (Part 4)**tc "4.7.2.3.2.8-4 Sort Detonation Distance CSU Flowchart (Part 4)" \f f§

µ §

**Figure 4.7.2.3.2.8-5**
**Sort Detonation Distance CSU Flowchart (Part 5)**tc "4.7.2.3.2.8-5 Sort Detonation Distance CSU Flowchart (Part 5)" \f f§

**4.7.2.3.2.9 Sort Detonation Distance CSU Algorithms**tc "4.7.2.3.2.9 Sort Detonation Distance CSU Algorithms"§

The Sort Detonation Distance CSU contains an algorithm to compare two distance values which are components of a Sort Record structure (see Table 4.7.2.3.2.4-2). The algorithm is designed as a function, which returns the Boolean result of the comparison which uses the "greater than" operator. This function will be used in the Sort List CSU (DL-CSU-7.1.1) to sort the array of pointers to PDUs into ascending order with respect to the distance from the reference position.

**Function (A)**

```
function Distance_From_Reference_Is_Greater (Left  : in Sort_Record,
                                             Right : in Sort_Record)
  return BOOLEAN is

begin
  return (Left.Distance > Right.Distance);
end Distance_From_Reference_Is_Greater;
```

**4.7.2.3.2.10 Sort Detonation Distance CSU Limitations**tc "4.7.2.3.2.10 Sort Detonation Distance CSU Limitations"§

There are no limitations or unusual features in the Sort Detonation Distance CSU.

**4.7.2.4 Sort Laser Distance CSU (DL-CSU-7.2.4)**tc "4.7.2.4 Sort Laser Distance CSU (DL-CSU-7.2.4)"§

The Sort Laser Distance CSU (DL-CSU-7.2.4) sorts a list of pointers to Laser PDUs into ascending order based on the distance between the position within the PDU and the input reference position. The following subparagraphs provide the design information for this CSU.

**4.7.2.4.1 Sort Laser Distance CSU Design Specification/Constraints**tc "4.7.2.4.1 Sort Laser Distance CSU Design Specification/Constraints"§

Table 4.7.2.4.1-1 identifies the requirements that are satisfied or partially satisfied by the Sort Laser Distance CSU. There are no design constraints for this CSU.

**Table 4.7.2.4.1-1**
**Requirements Allocated to the Sort Laser Distance CSU**tc " 4.7.2.4.1-1 Requirements Allocated to the Sort Laser Distance CSU" \f t§

| Requirement ID | DL SRS Paragraph |
|---|---|
| DL-R-32 | 3.2.7.2.4 |

**4.7.2.4.2 Sort Laser Distance CSU Design**tc "4.7.2.4.2 Sort Laser Distance CSU Design"§

The following subparagraphs specify the design of the Sort Laser Distance CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.7.2.4.2.1 Sort Laser Distance CSU Input/Output Data Elements**tc "4.7.2.4.2.1 Sort Laser Distance CSU Input/Output Data Elements"§

Table 4.7.2.4.2.1-1 identifies and states the purpose of each input and output data element to the Sort Laser Distance CSU.

**Table 4.7.2.4.2.1-1**
**Sort Laser Distance CSU I/O Data**tc " 4.7.2.4.2.1-1 Sort Laser Distance CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Laser PDU List Head | Input/Output | Points to a list of pointers to Laser PDUs. | See DL IRS, Table 3.2.2-2, DL-D-22 |

**4.7.2.4.2.2 Sort Laser Distance CSU Local Data Elements**tc "4.7.2.4.2.2 Sort Laser Distance CSU Local Data Elements"§

Table 4.7.2.4.2.2-1 identifies and describes each data element that originates within the Sort Laser CSU and which is not used by any other CSU. Table 4.7.2.4.2.2-2 provides the size, units, and range for each of the local data elements. Table 4.7.2.4.2.2-3 describes the local function required by this CSU. Table 4.7.2.4.2.2-4 identifies the scope of each local data element and function.

**Table 4.7.2.4.2.2-1**
**Sort Laser Distance CSU Local Data Descriptions**tc " 4.7.2.4.2.2-1 Sort Laser Distance CSU Local Data Descriptions" \f t§

| Name | Description | Data Type |
|---|---|---|
| Item Count | The element which is used to store the result of counting the number of nodes in the list of Laser PDU pointers. | See Table 4.7.2.4.2.2-2 |
| Loop Index | The element used as an index for the array processing loop. | See Table 4.7.2.4.2.2-2 |
| Laser Pointer Array | An array containing all Laser PDU pointers and an associated distance value. The distance is measured between the world coordinates position within each Laser PDU and the input reference position. | See Table 4.7.2.4.2.2-2 |

**Table 4.7.2.4.2.2-2**
**Size, Units, and Range of Local Data for the Sort Laser Distance CSU**tc " 4.7.2.4.2.2-2 Size, Units, and Range of Local Data for the Sort Laser Distance CSU" \f t§

| Name | Size (Bits) | Units | Range |
|---|---|---|---|
| Item Count | 16 | N/A | 0..65535 |
| Loop Index | 16 | N/A | 0..65535 |
| Laser Pointer Array | 96 * Item Count | See Table 4.7.2.4.2.4-1 | |

**Table 4.7.2.4.2.2-3**
**Sort Laser Distance CSU Local Function Description**tc " 4.7.2.4.2.2-3 Sort Laser Distance CSU Local Function Description" \f t§

| Name | Description | Algorithm |
|---|---|---|
| Distance From Reference Is Greater | Returns the Boolean result of comparing two distances using the "greater than" operator. | See paragraph 4.7.2.4.2.9 |

**Table 4.7.2.4.2.2-4**
**Scope Level of Local Declarations for the Sort Laser Distance CSU**tc " 4.7.2.4.2.2-4 Scope Level of Local Declarations for the Sort Laser Distance CSU" \f t§

| Name | Scope Level |
|---|---|
| Item Count | Sort Laser Distance unit |
| Loop Index | Sort Pointer Array block |
| Laser Pointer Array | Sort Pointer Array block |
| Distance From Reference Is Greater | Sort Pointer Array block |

**4.7.2.4.2.3 Sort Laser Distance CSU Global Data Elements**tc "4.7.2.4.2.3 Sort Laser Distance CSU Global Data Elements"§

The Sort Laser Distance CSU does not utilize any global data elements.

**4.7.2.4.2.4 Sort Laser Distance CSU Local and Shared Data Structures**tc "4.7.2.4.2.4 Sort Laser Distance CSU Local and Shared Data Structures"§

Table 4.7.2.4.2.4-1 describes local data structures implemented by the Sort Laser Distance CSU. This CSU does not utilize any shared data structures.

**Table 4.7.2.4.2.4-1**
**Laser Pointer Array Data Structure**tc " 4.7.2.4.2.4-1 Laser Pointer Array Data Structure" \f t§

| Array Bounds | Array Element Type | Units | Range |
|---|---|---|---|
| 1 .. Item Count | Sort Record | See Table 4.7.2.4.2.4-2 | |

**Table 4.7.2.4.2.4-2**
**Sort Record Data Structure**tc " 4.7.2.4.2.4-2 Sort Record Data Structure" \f t§

| Data Element | Data Type | Size (Bits) | Units | Range |
|---|---|---|---|---|
| Distance | See DL IRS, Table 3.2.2-2, DL-D-06 | | | |
| Laser PDU List Pointer Node | See DL IRS, Table 3.2.2-2, DL-D-22 | | | |

**4.7.2.4.2.5 Sort Laser Distance CSU Interrupts and Signals**tc "4.7.2.4.2.5 Sort Laser Distance CSU Interrupts and Signals"§

The Sort Laser Distance CSU does not handle any interrupts or signals.

**4.7.2.4.2.6 Sort Laser Distance CSU Error Handling**tc "4.7.2.4.2.6 Sort Laser Distance CSU Error Handling"§

The Sort Laser Distance CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of SORT_LASER_DIST to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Sort Laser Distance CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.7.2.4.2.7 Sort Laser Distance CSU Use of Other Elements**tc "4.7.2.4.2.7 Sort Laser Distance CSU Use of Other Elements"§

The Sort Laser Distance CSU does not use system service routines, global data files, or other global elements.

**4.7.2.4.2.8 Sort Laser Distance CSU Logic Flow**tc "4.7.2.4.2.8 Sort Laser Distance CSU Logic Flow"§

The logic flow of the Sort Laser Distance CSU is represented graphically by a flowchart in Figures 4.7.2.4.2.8-1 through 4.7.2.4.2.8-5. Execution of this CSU is initiated by external CSCIs. The Sort Laser Distance CSU initiates the Calculate Distance CSU (DL-CSU-3.1). This CSU also instantiates the generic Sort List CSU (DL-CSU-7.1.1) and initiates the new instance of the Sort List CSU.

µ §

**Figure 4.7.2.4.2.8-1**
**Sort Laser Distance CSU Flowchart (Part 1)**tc "4.7.2.4.2.8-1 Sort Laser Distance CSU Flowchart (Part 1)" \f f§
µ **§**

**Figure 4.7.2.4.2.8-2**
**Sort Laser Distance CSU Flowchart (Part 2)**tc "4.7.2.4.2.8-2 Sort Laser Distance CSU Flowchart (Part 2)" \f f§

µ §

**Figure 4.7.2.4.2.8-3**
**Sort Laser Distance CSU Flowchart (Part 3)**tc "4.7.2.4.2.8-3 Sort Laser Distance CSU Flowchart (Part 3)" \f f§

**Figure 4.7.2.4.2.8-4**
**Sort Laser Distance CSU Flowchart (Part 4)**tc "4.7.2.4.2.8-4 Sort Laser Distance CSU Flowchart (Part 4)" \f f§

**Figure 4.7.2.4.2.8-5**
**Sort Laser Distance CSU Flowchart (Part 5)**tc "4.7.2.4.2.8-5 Sort Laser Distance CSU Flowchart (Part 5)" \f f§

**4.7.2.4.2.9 Sort Laser Distance CSU Algorithms**tc "4.7.2.4.2.9 Sort Laser Distance CSU Algorithms"§

The Sort Laser Distance CSU contains an algorithm to compare two distance values which are components of a Sort Record structure (see Table 4.7.2.4.2.4-2). The algorithm is designed as a function, which returns the Boolean result of the comparison which uses the "greater than" operator. This function will be used in the Sort List CSU (DL-CSU-7.1.1) to sort the array of pointers to PDUs into ascending order with respect to the distance from the reference position.

**Function (A)**

```
function Distance_From_Reference_Is_Greater (Left  : in Sort_Record,
                                             Right : in Sort_Record)
  return BOOLEAN is

begin
  return (Left.Distance > Right.Distance);
end Distance_From_Reference_Is_Greater;
```

**4.7.2.4.2.10 Sort Laser Distance CSU Limitations**tc "4.7.2.4.2.10 Sort Laser Distance CSU Limitations"§

There are no limitations or unusual features in the Sort Laser Distance CSU.

**4.7.2.5 Sort Transmitter Distance CSU (DL-CSU-7.2.5)**tc "4.7.2.5 Sort Transmitter Distance CSU (DL-CSU-7.2.5)"§

The Sort Transmitter Distance CSU (DL-CSU-7.2.5) sorts a list of pointers to Transmitter PDUs into ascending order based on the distance between the position within the PDU and the input reference position. The following subparagraphs provide the design information for this CSU.

**4.7.2.5.1 Sort Transmitter Distance CSU Design Specification/Constraints**tc "4.7.2.5.1 Sort Transmitter Distance CSU Design Specification/Constraints"§

Table 4.7.2.5.1-1 identifies the requirements that are satisfied or partially satisfied by the Sort Transmitter Distance CSU. There are no design constraints for this CSU.

**Table 4.7.2.5.1-1**
**Requirements Allocated to the Sort Transmitter Distance CSU**tc " 4.7.2.5.1-1 Requirements Allocated to the Sort Transmitter Distance CSU" \f t§

| Requirement ID | DL SRS Paragraph |
|---|---|
| DL-R-33 | 3.2.7.2.5 |

**4.7.2.5.2 Sort Transmitter Distance CSU Design**tc "4.7.2.5.2 Sort Transmitter Distance CSU Design"§

The following subparagraphs specify the design of the Sort Transmitter Distance CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

**4.7.2.5.2.1 Sort Transmitter Distance CSU Input/Output Data Elements**tc "4.7.2.5.2.1 Sort Transmitter Distance CSU Input/Output Data Elements"§

Table 4.7.2.5.2.1-1 identifies and states the purpose of each input and output data element to the Sort Transmitter Distance CSU.

**Table 4.7.2.5.2.1-1**
**Sort Transmitter Distance CSU I/O Data**tc " 4.7.2.5.2.1-1 Sort Transmitter Distance CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Transmitter PDU List Head | Input/Output | Points to a list of pointers to Transmitter PDUs. | See DL IRS, Table 3.2.2-2, DL-D-25 |

**4.7.2.5.2.2 Sort Transmitter Distance CSU Local Data Elements**tc "4.7.2.5.2.2Sort Transmitter Distance CSU Local Data Elements"§

Table 4.7.2.5.2.2-1 identifies and describes each data element that originates within the Sort Transmitter CSU and which is not used by any other CSU. Table 4.7.2.5.2.2-2 provides the size, units, and range for each of the local data elements. Table 4.7.2.5.2.2-3 describes the local function required by this CSU. Table 4.7.2.5.2.2-4 identifies the scope of each local data element and function.

**Table 4.7.2.5.2.2-1**
**Sort Transmitter Distance CSU Local Data Descriptions**tc " 4.7.2.5.2.2-1 Sort Transmitter Distance CSU Local Data Descriptions" \f t§

| Name | Description | Data Type |
|---|---|---|
| Item Count | The element which is used to store the result of counting the number of nodes in the list of Transmitter PDU pointers. | See Table 4.7.2.5.2.2-2 |
| Loop Index | The element used as an index for the array processing loop. | See Table 4.7.2.5.2.2-2 |
| Transmitter Pointer Array | An array containing all Transmitter PDU pointers and an associated distance value. The distance is measured between the world coordinates position within each Transmitter PDU and the input reference position. | See Table 4.7.2.5.2.2-2 |

**Table 4.7.2.5.2.2-2**
**Size, Units, and Range of Local Data for the Sort Transmitter Distance CSU**tc " 4.7.2.5.2.2-2 Size, Units, and Range of Local Data for the Sort Transmitter Distance CSU" \f t§

| Name | Size (Bits) | Units | Range |
|---|---|---|---|
| Item Count | 16 | N/A | 0..65535 |
| Loop Index | 16 | N/A | 0..65535 |
| Transmitter Pointer Array | 96 * Item Count | See Table 4.7.2.5.2.4-1 | |

**Table 4.7.2.5.2.2-3**
**Sort Transmitter Distance CSU Local Function Description**tc " 4.7.2.5.2.2-3 Sort Transmitter Distance CSU Local Function Description" \f t§

| Name | Description | Algorithm |
|---|---|---|
| Distance From Reference Is Greater | Returns the Boolean result of comparing two distances using the "greater than" operator. | See paragraph 4.7.2.5.2.9 |

**Table 4.7.2.5.2.2-4**
**Scope Level of Local Declarations for the Sort Transmitter Distance CSU**tc " 4.7.2.5.2.2-4 Scope Level of Local Declarations for the Sort Transmitter Distance CSU" \f t§

| **Name** | **Scope Level** |
|---|---|
| Item Count | Sort Transmitter Distance unit |
| Loop Index | Sort Pointer Array block |
| Transmitter Pointer Array | Sort Pointer Array block |
| Distance From Reference Is Greater | Sort Pointer Array block |

#### 4.7.2.5.2.3 Sort Transmitter Distance CSU Global Data Elementstc "4.7.2.5.2.3 Sort Transmitter Distance CSU Global Data Elements"§

The Sort Transmitter Distance CSU does not utilize any global data elements.

#### 4.7.2.5.2.4 Sort Transmitter Distance CSU Local and Shared Data Structurestc "4.7.2.5.2.4 Sort Transmitter Distance CSU Local and Shared Data Structures"§

Table 4.7.2.5.2.4-1 describes local data structures implemented by the Sort Transmitter Distance CSU. This CSU does not utilize any shared data structures.

**Table 4.7.2.5.2.4-1**
**Transmitter Pointer Array Data Structure**tc " 4.7.2.5.2.4-1 Transmitter Pointer Array Data Structure" \f t§

| **Array Bounds** | **Array Element Type** | **Units** | **Range** |
|---|---|---|---|
| 1 .. Item Count | Sort Record | See Table 4.7.2.5.2.4-2 | |

**Table 4.7.2.5.2.4-2**
**Sort Record Data Structure**tc " 4.7.2.5.2.4-2 Sort Record Data Structure" \f t§

| Data Element | Data Type | Size (Bits) | Units | Range |
|---|---|---|---|---|
| Distance | See DL IRS, Table 3.2.2-2, DL-D-06 | | | |
| Transmitter PDU List Pointer Node | See DL IRS, Table 3.2.2-2, DL-D-25 | | | |

**4.7.2.5.2.5 Sort Transmitter Distance CSU Interrupts and Signals**tc "4.7.2.5.2.5 Sort Transmitter Distance CSU Interrupts and Signals"§

The Sort Transmitter Distance CSU does not handle any interrupts or signals.

**4.7.2.5.2.6 Sort Transmitter Distance CSU Error Handling**tc "4.7.2.5.2.6 Sort Transmitter Distance CSU Error Handling"§

The Sort Transmitter Distance CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of SORT_TRANSMITTER_DIST to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Sort Transmitter Distance CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.7.2.5.2.7 Sort Transmitter Distance CSU Use of Other Elements**tc "4.7.2.5.2.7 Sort Transmitter Distance CSU Use of Other Elements"§

The Sort Transmitter Distance CSU does not use system service routines, global data files, or other global elements.

**4.7.2.5.2.8 Sort Transmitter Distance CSU Logic Flow**tc "4.7.2.5.2.8 Sort Transmitter Distance CSU Logic Flow"§

The logic flow of the Sort Transmitter Distance CSU is represented graphically by a flowchart in Figures 4.7.2.5.2.8-1 through 4.7.2.5.2.8-5. Execution of this CSU is initiated by external CSCIs. The Sort Transmitter Distance CSU initiates the Calculate Distance CSU (DL-CSU-3.1). This CSU also instantiates the generic Sort List CSU (DL-CSU-7.1.1) and initiates the new instance of the Sort List CSU.

µ §

**Figure 4.7.2.5.2.8-1**
**Sort Transmitter Distance CSU Flowchart (Part 1)**tc "4.7.2.5.2.8-1 Sort Transmitter Distance CSU Flowchart (Part 1)" \f f§

µ **§**

**Figure 4.7.2.5.2.8-2**
**Sort Transmitter Distance CSU Flowchart (Part 2)**tc "4.7.2.5.2.8-2	Sort Transmitter Distance CSU Flowchart (Part 2)" \f f§

**Figure 4.7.2.5.2.8-3**
**Sort Transmitter Distance CSU Flowchart (Part 3)**tc "4.7.2.5.2.8-3 Sort Transmitter Distance CSU Flowchart (Part 3)" \f f§

**Figure 4.7.2.5.2.8-4**
**Sort Transmitter Distance CSU Flowchart (Part 4)**tc "4.7.2.5.2.8-4 Sort Transmitter Distance CSU Flowchart (Part 4)" \f f§

µ §

**Figure 4.7.2.5.2.8-5**
**Sort Transmitter Distance CSU Flowchart (Part 5)**tc "4.7.2.5.2.8-5 Sort Transmitter Distance CSU Flowchart (Part 5)" \f f§

**4.7.2.5.2.9 Sort Transmitter Distance CSU Algorithms**tc "4.7.2.5.2.9 Sort Transmitter Distance CSU Algorithms"§

The Sort Transmitter Distance CSU contains an algorithm to compare two distance values which are components of a Sort Record structure (see Table 4.7.2.5.2.4-2). The algorithm is designed as a function, which returns the Boolean result of the comparison which uses the "greater than" operator. This function will be used in the Sort List CSU (DL-CSU-7.1.1) to sort the array of pointers to PDUs into ascending order with respect to the distance from the reference position.

**Function (A)**

```
function Distance_From_Reference_Is_Greater (Left  : in Sort_Record,
                                             Right : in Sort_Record)
  return BOOLEAN is

begin
   return (Left.Distance > Right.Distance);
end Distance_From_Reference_Is_Greater;
```

**4.7.2.5.2.10 Sort Transmitter Distance CSU Limitations**tc "4.7.2.5.2.10 Sort Transmitter Distance CSU Limitations"§

There are no limitations or unusual features in the Sort Transmitter Distance CSU.

**4.7.3 Sort By Velocity CSC (DL-CSC-4.7.3)**tc "4.7.3 Sort By Velocity CSC (DL-CSC-4.7.3)"§

The following subparagraphs identify and describe each of the CSUs of the Sort By Velocity CSC. Figure 4.7.3-1 identifies the CSUs of this CSC. Figure 4.7.3-2 shows the execution control flow relationships of the CSUs of this CSC. Figure 4.7.3-3 shows the data flow relationships of the CSUs of this CSC. (In these figures, solid lines with no arrows indicate a hierarchical relationship, solid lines with arrows indicate data flow, and dashed lines with arrows indicate control flow. Rectangles with solid borders represent units internal to the CSC, and rectangles with dashed borders indicate external CSCs and CSCIs.)

µ §

**Figure 4.7.3-1**
**Sort By Velocity CSC Computer Software Units**tc "4.7.3-1	Sort By Velocity CSC Computer Software Units" \f f§

µ §

**Figure 4.7.3-2**
**Sort By Velocity CSC Execution Control**tc "4.7.3-2	Sort By Velocity CSC Execution Control" \f f§

µ §

**Figure 4.7.3-3**
**Sort By Velocity CSC Data Flow**tc "4.7.3-3 Sort By Velocity CSC Data Flow" \f f §

#### 4.7.3.1 Sort Entity State Velocity CSU (DL-CSU-7.3.1)tc "4.7.3.1 Sort Entity State Velocity CSU (DL-CSU-7.3.1)"§

The Sort Entity State Velocity CSU (DL-CSU-7.3.1) sorts a list of pointers to Entity State PDUs into descending order based on the velocity magnitude of each entity. The following subparagraphs provide the design information for this CSU.

##### 4.7.3.1.1 Sort Entity State Velocity CSU Design Specification/Constraintstc "4.7.3.1.1 Sort Entity State Velocity CSU Design Specification/Constraints"§

Table 4.7.3.1.1-1 identifies the requirements that are satisfied or partially satisfied by the Sort Entity State Velocity CSU. There are no design constraints for this CSU.

**Table 4.7.3.1.1-1**
**Requirements Allocated to the Sort Entity State Velocity CSU**tc " 4.7.3.1.1-1 Requirements Allocated to the Sort Entity State Velocity CSU" \f t§

| Requirement ID | DL SRS Paragraph |
|---|---|
| DL-R-34 | 3.2.7.3.1 |

##### 4.7.3.1.2 Sort Entity State Velocity CSU Designtc "4.7.3.1.2 Sort Entity State Velocity CSU Design"§

The following subparagraphs specify the design of the Sort Entity State Velocity CSU, including input/output data, local data, local and shared data structures, interrupts, error handling, logic flow, algorithms, limitations, and use of other elements.

###### 4.7.3.1.2.1 Sort Entity State Velocity CSU Input/Output Data Elementstc "4.7.3.1.2.1 Sort Entity State Velocity CSU Input/Output Data Elements"§

Table 4.7.3.1.2.1-1 identifies and states the purpose of each input and output data element to the Sort Entity State Velocity CSU.

**Table 4.7.3.1.2.1-1**
**Sort Entity State Velocity CSU I/O Data**tc " 4.7.3.1.2.1-1 Sort Entity State Velocity CSU I/O Data" \f t§

| Data Element | Input/Output | Purpose | Data Type |
|---|---|---|---|
| Status | Output | Indicates whether this unit encountered an error condition. | See DL IRS, Table 3.2.2-2, DL-D-26 |
| Entity State PDU List Head | Input/Output | Points to a list of pointers to Entity State PDUs. | See DL IRS, Table 3.2.2-2, DL-D-13 |

**4.7.3.1.2.2 Sort Entity State Velocity CSU Local Data Elements**tc "4.7.3.1.2.2 Sort Entity State Velocity CSU Local Data Elements"§

Table 4.7.3.1.2.2-1 identifies and describes each data element that originates within the Sort Entity State CSU and which is not used by any other CSU. Table 4.7.3.1.2.2-2 provides the size, units, and range for each of the local data elements. Table 4.7.3.1.2.2-3 describes the local function required by this CSU. Table 4.7.3.1.2.2-4 identifies the scope of each local data element and function.

**Table 4.7.3.1.2.2-1**
**Sort Entity State Velocity CSU Local Data Descriptions**tc " 4.7.3.1.2.2-1 Sort Entity State Velocity CSU Local Data Descriptions" \f t§

| Name | Description | Data Type |
|---|---|---|
| Item Count | The element which is used to store the result of counting the number of nodes in the list of Entity State PDU pointers. | See Table 4.7.3.1.2.2-2 |
| Loop Index | The element used as an index for the array processing loop. | See Table 4.7.3.1.2.2-2 |
| Entity State Pointer Array | An array containing all Entity State PDU pointers and an associated velocity value. The velocity represents the magnitude of the Entity Linear Velocity record within each Entity State PDU. | See Table 4.7.3.1.2.2-2 |

**Table 4.7.3.1.2.2-2**
**Size, Units, and Range of Local Data for the Sort Entity State Velocity CSU**tc " 4.7.3.1.2.2-2 Size, Units, and Range of Local Data for the Sort Entity State Velocity CSU" \f t§

| Name | Size (Bits) | Units | Range |
|---|---|---|---|
| Item Count | 16 | N/A | 0..65535 |
| Loop Index | 16 | N/A | 0..65535 |
| Entity State Pointer Array | 96 * Item Count | See Table 4.7.3.1.2.4-1 | |

**Table 4.7.3.1.2.2-3**
**Sort Entity State Velocity CSU Local Function Description**tc " 4.7.3.1.2.2-3 Sort Entity State Velocity CSU Local Function Description" \f t§

| Name | Description | Algorithm |
|---|---|---|
| Velocity Magnitude Is Less | Returns the Boolean result of comparing two velocity vectors using the "less than" operator. | See paragraph 4.7.3.1.2.9 |

**Table 4.7.3.1.2.2-4**
**Scope Level of Local Declarations for the Sort Entity State Velocity CSU**tc " 4.7.3.1.2.2-4 Scope Level of Local Declarations for the Sort Entity State Velocity CSU" \f t§

| Name | Scope Level |
|---|---|
| Item Count | Sort Entity State Velocity unit |
| Loop Index | Sort Pointer Array block |
| Entity State Pointer Array | Sort Pointer Array block |
| Velocity Magnitude Is Less | Sort Pointer Array block |

**4.7.3.1.2.3 Sort Entity State Velocity CSU Global Data Elements**tc "4.7.3.1.2.3 Sort Entity State Velocity CSU Global Data Elements"§

The Sort Entity State Velocity CSU does not utilize any global data elements.

**4.7.3.1.2.4 Sort Entity State Velocity CSU Local and Shared Data Structures**tc "4.7.3.1.2.4 Sort Entity State Velocity CSU Local and Shared Data Structures"§

Table 4.7.3.1.2.4-1 describes local data structures implemented by the Sort Entity State Velocity CSU. This CSU does not utilize any shared data structures.

**Table 4.7.3.1.2.4-1**
**Entity State Pointer Array Data Structure**tc " 4.7.3.1.2.4-1 Entity State Pointer Array Data Structure" \f t§

| Array Bounds | Array Element Type | Units | Range |
|---|---|---|---|
| 1 .. Item Count | Sort Record | See Table 4.7.3.1.2.4-2 | |

**Table 4.7.3.1.2.4-2**
**Sort Record Data Structure**tc " 4.7.3.1.2.4-2 Sort Record Data Structure" \f t§

| Data Element | Data Type | Size (Bits) | Units | Range |
|---|---|---|---|---|
| Velocity | See DL IRS, Table 3.2.2-2, DL-D-07 | | | |
| Entity State PDU List Pointer Node | See DL IRS, Table 3.2.2-2, DL-D-13 | | | |

**4.7.3.1.2.5 Sort Entity State Velocity CSU Interrupts and Signals**tc "4.7.3.1.2.5 Sort Entity State Velocity CSU Interrupts and Signals"§

The Sort Entity State Velocity CSU does not handle any interrupts or signals.

**4.7.3.1.2.6 Sort Entity State Velocity CSU Error Handling**tc "4.7.3.1.2.6 Sort Entity State Velocity CSU Error Handling"§

The Sort Entity State Velocity CSU handles run-time errors via the use of a status code. An Ada exception handler will trap error conditions which originate in this unit, and assign a value of SORT_ES_VEL to the element called Status. If no errors occur as a result of executing this unit, the Status will contain a value of SUCCESS. If an error occurs in a unit which is called by the Sort Entity State Velocity CSU, then the error code for that unit will be passed back to the caller of this unit via the element Status.

**4.7.3.1.2.7 Sort Entity State Velocity CSU Use of Other Elements**tc "4.7.3.1.2.7 Sort Entity State Velocity CSU Use of Other Elements"§

The Sort Entity State Velocity CSU does not use system service routines, global data files, or other global elements.

**4.7.3.1.2.8 Sort Entity State Velocity CSU Logic Flow**tc "4.7.3.1.2.8 Sort Entity State Velocity CSU Logic Flow"§

The logic flow of the Sort Entity State Velocity CSU is represented graphically by a flowchart in Figures 4.7.3.1.2.8-1 through 4.7.3.1.2.8-5. Execution of this CSU is initiated by external CSCIs. The Sort Entity State Velocity CSU initiates the Calculate Velocity CSU (DL-CSU-3.3). This CSU also instantiates the generic Sort List CSU (DL-CSU-7.1.1) and initiates the new instance of the Sort List CSU.

µ §

**Figure 4.7.3.1.2.8-1**
**Sort Entity State Velocity CSU Flowchart (Part 1)**tc "4.7.3.1.2.8-1 Sort Entity State Velocity CSU Flowchart (Part 1)" \f f§

µ **§**

**Figure 4.7.3.1.2.8-2**
**Sort Entity State Velocity CSU Flowchart (Part 2)**tc "4.7.3.1.2.8-2 Sort Entity State Velocity CSU Flowchart (Part 2)" \f f§

µ §

**Figure 4.7.3.1.2.8-3**
**Sort Entity State Velocity CSU Flowchart (Part 3)**tc "4.7.3.1.2.8-3 Sort Entity State Velocity CSU Flowchart (Part 3)" \f f§

µ §

**Figure 4.7.3.1.2.8-4**
**Sort Entity State Velocity CSU Flowchart (Part 4)**tc "4.7.3.1.2.8-4 Sort Entity State Velocity CSU Flowchart (Part 4)" \f f§

µ §

**Figure 4.7.3.1.2.8-5**
**Sort Entity State Velocity CSU Flowchart (Part 5)**tc "4.7.3.1.2.8-5 Sort Entity State Velocity CSU Flowchart (Part 5)" \f f§

**4.7.3.1.2.9 Sort Entity State Velocity CSU Algorithms**tc "4.7.3.1.2.9 Sort Entity State Velocity CSU Algorithms"§

The Sort Entity State Velocity CSU contains an algorithm to compare two velocity magnitude values which are components of a Sort Record structure (see Table 4.7.3.1.2.4-2). The algorithm is designed as a function, which returns the Boolean result of the comparison which uses the "less than" operator. This function will be used in the Sort List CSU (DL-CSU-7.1.1) to sort the array of pointers to PDUs into descending order with respect to velocity magnitude.

**<u>Function (A)</u>**

```
function Velocity_Magnitude_Is_Less (Left  : in Sort_Record,
                                     Right : in Sort_Record)
  return BOOLEAN is

begin
   return (Left.Velocity < Right.Velocity);
end Velocity_Magnitude_Is_Less;
```

**4.7.3.1.2.10 Sort Entity State Velocity CSU Limitations**tc "4.7.3.1.2.10 Sort Entity State Velocity CSU Limitations"§

There are no limitations or unusual features in the Sort Entity State Velocity CSU.

# 5 CSCI Datatc "5	CSCI Data"§

Table 5-1 describes the global data elements within the DL CSCI. Table 5-2 documents the CSU(s) where the data element is set or calculated, and the CSU(s) where the data element is used. Data elements of the CSCI's external interfaces have been identified and referenced at the point of usage.

**Table 5-1**
**Global Data Element Descriptions**tc "5-1   Global Data Element Descriptions" \f t§

| Identifier | Name | Description | Source CSU(s) | Destination CSU(s) |
|---|---|---|---|---|
| DL-GD-1 | Delta Time | Time elapsed since last position update. | DL-CSU-2.1<br>DL-CSU-2.4<br>DL-CSU-2.5<br>DL-CSU-2.6 | DL-CSU-2.1<br>DL-CSU-2.3<br>DL-CSU-2.4<br>DL-CSU-2.5<br>DL-CSU-2.6 |
| DL-GD-2 | Entity Angular Velocity | Rate at which an entity's orientation is changing, using the entity's own coordinate system. | DL-CSU-2.1 | DL-CSU-2.5<br>DL-CSU-2.6 |
| DL-GD-3 | Entity Linear Acceleration | Linear acceleration of entity in geocentric (world) coordinates. | DL-CSU-2.1 | DL-CSU-2.4<br>DL-CSU-2.6 |
| DL-GD-4 | Entity Linear Velocity | Linear velocity of entity in geocentric (world) coordinates. | DL-CSU-2.1<br>DL-CSU-2.4<br>DL-CSU-2.5<br>DL-CSU-2.6 | DL-CSU-2.1<br>DL-CSU-2.3<br>DL-CSU-2.4<br>DL-CSU-2.5<br>DL-CSU-2.6 |
| DL-GD-5 | Entity Location | Position of entity in geocentric (world) coordinates. | DL-CSU-2.1<br>DL-CSU-2.3<br>DL-CSU-2.4<br>DL-CSU-2.5<br>DL-CSU-2.6 | DL-CSU-2.1<br>DL-CSU-2.3<br>DL-CSU-2.4<br>DL-CSU-2.5<br>DL-CSU-2.6 |
| DL-GD-6 | Entity Orientation | Orientation of entity using Euler angles (which form a rotational transformation matrix). | DL-CSU-2.1<br>DL-CSU-2.5<br>DL-CSU-2.6 | DL-CSU-2.1<br>DL-CSU-2.5<br>DL-CSU-2.6 |

**Table 5-2**
**Global Data Element Definitions**tc "5-2 Global Data Element Definitions" \f t§

| Identifier | Name | Units | Range | Size (Bits) | Type |
|---|---|---|---|---|---|
| DL-GD-1 | Delta Time | µs | n/a | 32 | Unsigned Integer |
| DL-GD-2 | Entity Angular Velocity | See IST-CR-93-15, Paragraph 5.3.2, Angular Velocity Vector Record | | | |
| DL-GD-3 | Entity Linear Acceleration | See IST-CR-93-15, Paragraph 5.3.20.2, Linear Acceleration Vector | | | |
| DL-GD-4 | Entity Linear Velocity | See IST-CR-93-15, Paragraph 5.3.20.3, Linear Velocity Vector | | | |
| DL-GD-5 | Entity Location | See IST-CR-93-15, Paragraph 5.3.21, World Coordinates Record | | | |
| DL-GD-6 | Entity Orientation | See IST-CR-93-15, Paragraph 5.3.11, Euler Angles Record | | | |

## 6 DL CSCI Data Filestc "6  DL CSCI Data Files"§
The DL CSCI has no shared data files.

## 7 Notestc "7   Notes"§

The following subparagraphs contain additional information to assist in understanding this design document, such as system considerations, meanings of acronyms and abbreviations, and McCabe complexity measurements for the CSUs described in this document.

### 7.1 Acronyms and Abbreviationstc "7.1    Acronyms and Abbreviations"§

Table 8.3-1 contains a list of all acronyms and abbreviations used in this SDD, and their meanings as used in this document.

**Table 8.3-1**
**Acronyms and Abbreviations**tc " 8.3-1      Acronyms and Abbreviations" \f t§

| **Acronym/ Abbreviation** | **Meaning** |
|---|---|
| CDRL | Contract Data Requirements List |
| DL | DIS Library |
| CSC | Computer Software Component |
| CSU | Computer Software Unit |
| DID | Data Item Description |
| DIS | Distributed Interactive Simulation |

| | |
|---|---|
| DOD | Department of Defense |
| I/O | Input/Output |
| IRS | Interface Requirements Specification |
| IST | Institute for Simulation and Training |
| m | Meter |
| msec | Millisecond |
| N/A | Not Applicable |
| NAWCAD | Naval Air Warfare Center, Aircraft Division |
| PDU | Protocol Data Unit |
| sec | Second |
| SRS | Software Requirements Specification |