Interface Control Document MCCS to Science Instrument Software Interface (Functional) MCCS_SI_04

SOF-DA-ICD-SE03-052

Date: December 1, 2017
Revision: L

AFRC
Armstrong Flight Research Center
Edwards, CA 93523

ARC
Ames Research Center
Moffett Field, CA 94035

German Space Agency, DLR
Deutsches Zentrum für Luft und Raumfahrt

VERIFY THAT THIS IS THE CORRECT REVISION BEFORE USE

Interface Control Document
MCCS to Science Instrument Software Interface (Functional)
MCCS_SI_04

AUTHOR:

Victoria Moore (affiliate)
Digitally signed by Victoria Moore (affiliate)
Date: 2017.12.06 11:01:21 -08'00'

Critical Realm Corp/Victoria M. Moore, Software Systems Engineer — Date

Steven Culp (affiliate)
Digitally signed by Steven Culp (affiliate)
DN: c=US, o=U.S. Government, ou=NASA, ou=People, 0.9.2342.19200300.100.1.1=sculp, cn=Steven Culp (affiliate)
Date: 2017.12.06 11:56:17 -08'00'

SETI Institute/Steven Culp, Software Systems Engineer — Date

CONCURRENCE:

Elizabeth Moore
Digitally signed by Elizabeth Moore
DN: cn=Elizabeth Moore, o=USRA, ou=SOFIA, email=emmoore@sofia.usra.edu, c=US
Date: 2017.12.12 08:19:31 -07'00'

USRA/Elizabeth Moore, ISD Manager — Date

Bill Wohler
Digitally signed by Bill Wohler
Date: 2017.12.12 08:57:16 -08'00'

SETI/Bill Wohler, SI Software Interface Lead — Date

Nancy McKown
Digitally signed by Nancy McKown
Location: Palmdale, CA
Date: 2018.01.03 11:58:48 -08'00'

USRA/Nancy McKown, Lead Mission Director — Date

Holger Jakob (affiliate) 2017.12.13 09:03:57 +01'00'

DSI/Holger Jakob, TA Software and Improvement Lead — Date

MAY WINDREM
Digitally signed by MAY WINDREM
Date: 2017.12.13 09:33:36 -08'00'

NASA/May Windrem, Observing & Ground Systems Lead — Date

ALAN RHODES
Digitally signed by ALAN RHODES
Date: 2017.12.13 10:04:06 -08'00'

NASA/Alan Rhodes, SI Development Manager — Date

JONATHAN BROWN
Digitally signed by JONATHAN BROWN
Date: 2017.12.14 09:10:42 -08'00'

NASA/Jonathan Brown, SE&I Lead — Date

MATTHEW ENGA
Digitally signed by MATTHEW ENGA
DN: c=US, o=U.S. Government, ou=NASA, ou=People, cn=MATTHEW ENGA, 0.9.2342.19200300.100.1.1=menga
Date: 2017.12.18 19:19:10 +01'00'

NASA/Matt Enga, Observatory Systems Integration Lead — Date

EDWARD STANTON
Digitally signed by EDWARD STANTON
Date: 2017.12.19 07:55:41 -08'00'

NASA/Edward Stanton, Observatory Systems Director — Date

12/21/17

NASA/Michael Toberman, Operations Director — Date

Harold W Yorke
Digitally signed by Harold W Yorke
DN: cn=Harold W Yorke, o=USRA, ou=SOFIA Science Center, email=hyorke@usra.edu, c=US
Date: 2017.12.21 09:27:46 -08'00'

USRA/Harold Yorke, Science Mission Operations Director — Date

Digitally signed by Michael Hütwohl
DN: cn=Michael Hütwohl, o=IRS/DSI, ou=University of Stuttgart, email=huetwohl@dsi.uni-stuttgart.de, c=US
Date: 2017.12.22 09:29:04 +01'00'

DSI/Michael Hütwohl, Telescope Assembly Manager — Date

APPROVAL:

JEANETTE LE
Digitally signed by JEANETTE LE
Date: 2018.01.08 16:33:36 -08'00'

NASA/Jeanette Le, Chief Engineer — Date

Revision History

REV	DATE	DESCRIPTION	APPROVAL
A	5-6-04	L3 release (doc # 96145640-000, rev A). Incorporated miscellaneous changes. Completely replaced Appendix A and B.	N/A
B	9-21-10	Adapted for Early Science from 5/6/04 L3 version (doc # 96145640-000, rev A). Switched to current SOFIA document conventions. Major update to entire document. Details not available for Early Science are grayed out.	
C	5-2-11	Updated for Short Science 2 build. Specifically: 1) Rebuilt appendices A and B based on SS2 XML files (version 3.3.0). 2) Removed debug/diagnostic commands that should only be used by MCCS developers. 3) Removed instrument-specific SI command and data descriptions from appendices A and B. (These are input files to MCCS from SI teams and operations, not ICD specifications.) 4) Corrected typos and updated sample commands and responses in front matter to SS2 commands and data. 5) Showed EcRF and GalRF as available reference frames in Table 21 since L-3 indicates these are currently supported. 6) Removed references to x_bin & y_bin since these are not used by any SI teams (per L-3 and USRA review comments). 7) Added explanation of NeedDate field to front of appendix A. 8) Added KOSMA reference to sections 1-2. 9) Changed IMAGSN to IMAGE_SN in section 2.2 of Appendix C (per L-3 review comment). 10) Added Appendix E (Errata).	PMB
D	5-4-12	Updated for Segment 3 (see Appendix H for description of changes since rev C, and Appendix E for known errata in this version).	PMB
E	12-13-12	Updated to correct errata and match delivered MCCS for Segment 3 (primarily Appendices A	PMB

		& B). See Appendix H for a description of changes since rev D, and Appendix E for known errata in this version.	
E.1	1-16-13	Deleted “Low Limit” and “High Limit” numbers for nod.define pos angle value in Appendix B, pg B-30, per OCCB-CCR-043. Added 2 new coord.responses to Appendix B, pg B-9 per OCCB-CCR-044.	OCCB
F	3-10-14	Updated for MCCS’ first delivery for Phase 2 of Segment 3 (beginning of 2014). See Appendix H for a description of changes since rev E, and Appendix E for known errata in this version.	PMB
G	3-4-15	Document updated per CM approved template SOF-AR-FOR-PM93-2012B. See Appendix H for a description of changes since rev F, and Appendix E for known errata in this version.	OCCB
H	11-18-15	Updated for MCCS PIS/TAIPS release 40.2.0. See Appendix H for a description of changes since rev G, and Appendix E for known errata in this version.	OCCB
J	3-4-16	Updated for MCCS PIS/TAIPS release 40.3.0. See Appendix H for a description of changes since rev H, and Appendix E for known errata in this version.	OCCB
J.1	7-13-16	Updated Appendix E for known errata to coincide with a revised 40.3.0 release.	OCCB
K	7-24-17	Updated for MCCS PIS/TAIPS release 40.4.0 and 40.5.0. See Appendix H for a description of changes since rev J, and Appendix E for known errata in this version.	OCCB
L	12-01-17	Updated for MCCS PIS/TAIPS release 40.6.0. See Appendix H for a description of changes since rev K, and Appendix E for known errata in this version.	OCCB

Interface Control Document
MCCS to Science Instrument Software Interface
(Functional) MCCS_SI_04

Table of Contents

1. INTRODUCTION 1

1.1 PURPOSE AND SCOPE 1
1.2 MISSION CONTROL AND COMMUNICATION SYSTEM 2
1.3 TELESCOPE ASSEMBLY DESCRIPTION 5
1.4 SCIENCE INSTRUMENT SYSTEM DESCRIPTION 7
1.5 STANDARD FLIGHT DATA PRODUCTS 8

2. APPLICABLE DOCUMENTS 9

3. INTERFACE DESCRIPTION 10

3.1 SOFIA COMMAND LANGUAGE NETWORK CONNECTION PROTOCOL 11
3.2 SESSIONS AND ROLES 11
3.3 COMMUNICATION AND DATA EXCHANGE 12
3.3.1 SOFIA Command Language Protocol 12
3.3.1.1 Command Syntax 14
3.3.1.2 ASCII Response Syntax (Legacy or Formatted) 15
3.3.1.3 Binary Response Syntax 20
3.3.1.4 ASCII Value Format 23
3.3.2 Integrated Command Log 25
3.3.3 User Comments and the Integrated Comment Log 26
3.3.4 Alert and Alarm Log 27
3.3.5 SI Data-Definition XML Files 28
3.3.6 Observatory Status and Metric Data 41
3.4 MCCS PERFORMANCE 45
3.5 REFERENCE FRAMES 45
3.6 OBSERVING EFFECTS 47

4. SOFIA COMMAND LANGUAGE DESCRIPTION 49

4.1 COMMAND AND HOUSEKEEPING DATA NAMES 49
4.1.1 Argument Lists and Modifiers 49
4.1.1.1 Per-Command Arguments 51
4.1.1.2 Pre-Defined Entries 52
4.1.1.3 Keyword Mutual-Exclusivity 54
4.1.1.4 Default Keyword Values 54
4.1.2 Aliases 55
4.1.2.1 Pre-defined Aliases 56
4.1.3 User-Defined Positions and Offsets 57
4.1.3.1 Inline Definitions 58
4.1.3.2 Position and Offset Attributes 58
4.1.4 Values and Escape Sequences 61
4.2 MONITORING HOUSEKEEPING DATA 61
4.2.1 Subscribing to Housekeeping Data 61
4.2.2 Data Timestamps 63
4.2.2.1 Simplified Access to Data Timestamps (planned) 65

4.2.3 Generated Data Items....65
4.2.4 Monitoring User-Specified Limits....66
4.3 ALERTS AND ALARMS....67
4.3.1 Subscribing to Alerts/Alarms....68
4.3.2 Confirming Alarms....70
4.3.3 Disabling/Enabling Alerts/Alarms....71

5. DEFINITIONS, ABBREVIATIONS, ACRONYMS....73

APPENDIX A. SCL COMMANDS....A-1

APPENDIX B. HOUSEKEEPING DATA....B-1
APPENDIX B.1 DATA ORGANIZATION....B-2
APPENDIX B.2 DATA LATENCY....B-4
APPENDIX B.3 PERIODIC DATA RATE....B-6
APPENDIX B.4 DATA ITEM LIST....B-12

APPENDIX C. DIGITAL VIDEO DISTRIBUTION SUBSYSTEM....C-1
APPENDIX C.1 TA IMAGER PROCESSING SUBSYSTEM (TAIPS)....C-2
APPENDIX C.2 VIDEO PROCESSING AND RECORDING SUBSYSTEM (VPARS)....C-6

APPENDIX D. MISSION AUDIO DISTRIBUTION SUBSYSTEM....D-1

APPENDIX E. ERRATA....E-1

APPENDIX F. MCCS FILE FORMATS....F-1
APPENDIX F.1 ARCHIVE FILE....F-1
APPENDIX F.2 ARCHIVE INDEX FILE....F-9
APPENDIX F.3 EPHEMERIS FILE....F-11

APPENDIX G. ALERT AND ALARM MESSAGES....G-1

APPENDIX H. RELEASE NOTES....H-1

List of Figures

Figure **Title** **Page**
Figure 1. Scope of MCCS_SI_04....2
Figure 2. Telescope Assembly....6
Figure 3. (X, Y, Z) SI Reference Frame (SIRF)....30
Figure 4 Simulated image with SIRF overlay....40

List of Tables

Table **Title** **Page**
Table 1. Available Network Services....5
Table 2. Flight Data Products....8
Table 3. Session Roles....12
Table 4. Syntax Notation....14
Table 5. Command/Response Tokens....14

Table 6. ASCII Command Responses 16
Table 7. Binary SCL Command Responses....20
Table 8. Binary Data Types 22
Table 9. ASCII Value Formats 23
Table 10. User-Accessible Reference Frames 45
Table 11. Command Use of Reference Frames 46
Table 12. Observing Effects and their Use in Position Definition, Publication, and Pointing....47
Table 13. Common MCCS Units and Abbreviations 51
Table 14. Pre-Defined Entries Supported in SCL....53
Table 15. Arguments which Permit Inline Position Definition 57
Table 16. Position and Offset Reference Frame Attributes 58
Table 17. Additional Position Attributes 59
Table 18. Escape Sequences 61
Table 19. Generated Data Items....66
Table 20. Alert/Alarm Severity Levels....68
Table 21. Alert/Alarm Subscribe Command Keywords....69
Table 22. Alert/Alarm Response Tokens....69
Table 23. Alarm Confirm Command Keywords....70
Table 24. Confirm Command Responses 71
Table 25. Disabling Alert/Alarm Keywords....71
Table 26. Disable Command Responses....72

1. INTRODUCTION

1.1 Purpose and Scope

This document defines the functional interfaces between the Mission Controls and Communications System (MCCS) and its users. 'Users' includes the Science Instrument team (SI), other flight participants, and ground support personnel who use MCCS, as well as any computers which they may choose to connect to MCCS.

The physical interface between MCCS and the Science Instrument (SI) is defined in MCCS_SI_05 ICD (SOF-AR-ICD-SE03-2029).

Throughout this document items identified in *gray italic text* are not currently available but are planned for later development.

Appendix H describes changes since the previous version (revision E, the Phase 1 version). Users who used the previous version to develop their interface to SOFIA systems are encouraged to review Appendix H to help them scope the changes and to identify which may impact their use of SOFIA.

Appendix A describes the SOFIA Command Language (SCL) commands.

Appendix B describes MCCS housekeeping data. In particular, B.1 describes housekeeping naming conventions, B.2 and B.3 describes latency and data rate requirements for MCCS distribution of housekeeping data, and B.4 lists the housekeeping data values. Note that since MCCS is still under development, the actual rates provided do not generally meet the required rates in B.3. The actual rates currently available are listed in the last column of the table in B.4.

This ICD covers all SCL commands and data, with one exception – the low-level SCL commands (sometimes referred to as TA "pass-through" commands) and data used to control and monitor the Telescope Assembly (TA). Generally, users are not expected to invoke TA pass-through commands directly since higher-level alternative commands are provided as defined in Appendix A of this ICD. Descriptions of TA pass-through commands and data are not included in this document; interested users may refer to the TA_MCCS_F ICD for descriptions of those commands and housekeeping data.

Appendix C describes the Digital Video Distribution Subsystem (DVDS).

Appendix D describes the Mission Audio Distribution Subsystem (MADS).

Appendix E lists errata known to exist in this ICD.

Appendix F describes the format of some key files generated or read by MCCS.

Appendix G lists the alert and alarm messages which may be generated by MCCS.

In addition to its use for user interaction with MCCS, the interfaces described in this document have been adopted by the MCCS Workstation Subsystem and some other observatory systems internal and external to MCCS for command and communication.

The scope of this interface between MCCS and its users is depicted by Figure 1. The vertical dashed line demarks the functional interface between MCCS and User, which is defined by this ICD.

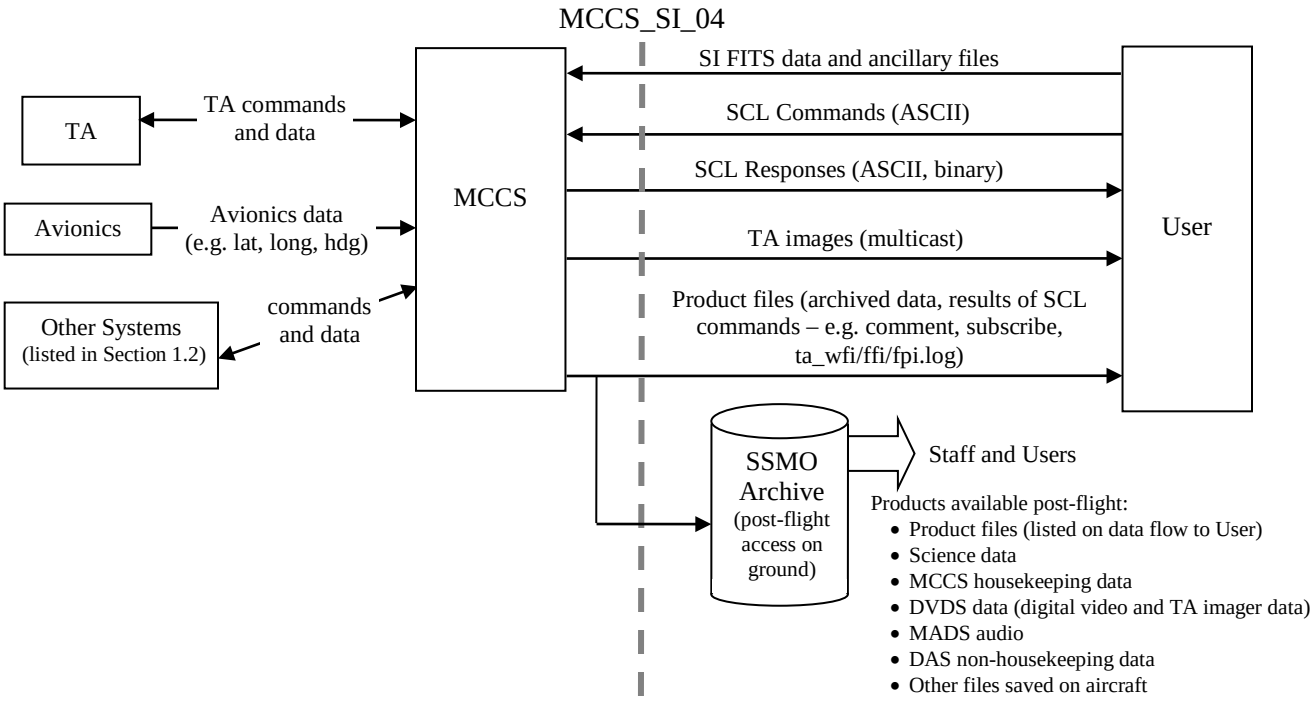

Figure 1. Scope of MCCS_SI_04

The arrows between MCCS and User represent interfaces that are available on the aircraft and in simulators of the aircraft systems. The arrow into the SSMO Archive represents the movement of SOFIA data products to ground systems for post-flight access by SOFIA staff, SI teams, and other users.

Files generated by MCCS, the SI team, or other SOFIA systems or users may be written to an NFS-mountable disk array. There they may be accessed in-flight and are moved post-flight to the SSMO Archive for access by SOFIA staff and users.

1.2 Mission Control and Communication System

The MCCS provides mission communication, power distribution, data acquisition, data archival, network services, and observatory crew interfaces.

In addition to the external interfaces to observatory uses described in this document, the MCCS supports external interfaces to the following observatory systems:

- Science Instrument: described in Section 1.4
- Telescope Assembly (TA): the telescope system, described in Section 1.3

- Cavity Environmental Control System (CECS): system used to cool down the cavity before flights and to keep the cavity dry
- Cavity Door Drive System (CDDS): control equipment to open/close the cavity door and control its movement
- External sensors: various measuring devices for routine system parameters (e.g., temperatures)
- Flight instrumentation: various measuring devices for non-routine system parameters (e.g., structural items); these are not part of the normal data flow
- Aircraft avionics: critical data regarding the aircraft state (e.g., altitude and location)
- Aircraft audio system: provides flight deck and other aircraft communication; MCCS interfaces to this system to allow communications between the MCCS audio subsystem (MADS) and the aircraft audio system.
- *Vacuum Pump System (VPS): provide and maintain vacuum in TA-SI tub*
- *Cryo-cooler System: electronic cooling system for SI*

MCCS itself is comprised of the following subsystems:

- Power Distribution Subsystem (PDS): distributes AC and DC power to observatory systems
- Mission Audio Data Subsystem (MADS): provides multi-channel voice communications throughout observatory
- Network Subsystem: provides Platform, Experimenter, and Video networks, with interconnection between the networks and connections around the aircraft for access by MCCS and user computers
- Platform Interface Subsystem (PIS): provides SOFIA Command Language (SCL) interface and associated algorithms
- Inter-Range Instrumentation Group/Network Time Protocol (IRIG/NTP) Time Distribution Subsystem: provides GPS-synchronized time to observatory systems
- Data Acquisition Subsystem (DAS): acquires data from sensors and other sources for use on the observatory
- Digital Video Distribution Subsystem (DVDS): MCCS subsystem to capture and distribute video data via two main components:
  - TA Image Processing Subsystem (TAIPS), which receives, processes, and distributes raw video image frames from TA imagers
  - Video Processing and Recording Subsystem (VPARS), which handles streaming video data from: workstation displays, video generators in VPARS, observatory cameras for situational awareness, and (optionally) from SI computers (to allow sharing of video with others onboard)
- Water Vapor Monitor (WVM): measures and provides water vapor burden information for science use
- Archiver Subsystem: stores and provides in-flight access to observatory data products; it is also the location of MCCS software configuration and data files

(/modstor); the Archiver subsystem is a server and high-reliability RAID disk array

- Workstation Subsystem: provides workstations, and associated MCCS GUI software for observatory monitoring and control, for the in-flight mission crew (this includes workstations at the MD and TO consoles, as well as workstations at the EPO console). All MCCS workstations connect to the Platform and Video networks.
- *Flight Manager Subsystem (FM): provides in-flight re-planning capabilities and automatic aircraft heading control.*
  (Note: Until the MCCS FM is implemented, in-flight flight re-planning capabilities are provided by the FMI software package running on the Science Flight Planner workstation. Heading control during observation legs will be performed automatically once FM is implemented, but is currently performed manually by the flight crew in coordination with the Mission Director and Science Flight Planner.)

MCCS networks are configured to permit the appropriate traffic between networks so that needed services (such as printing or access to the MCCS archival storage) are available where needed on any MCCS network. But each network has its own interconnections and primary roles as described below.

The Platform Network provides for intercommunication between MCCS subsystems.

The Experimenter Network supports the needs of visiting users with connections at:

- SI/PI Patch Panel (providing connections to SI/PI racks, Counter Weight Rack (CWR), as well as to the Science Instrument)
- Two conference tables, each of which supports four Experimenter network connections for user laptop computers
- Mission Director (MD), Science Flight Planner (SFP), and Telescope Operator (TO) consoles for mission crew laptop computers
- Connections for visitor laptop computers in forward EPO seating area (AKA 1st class section), on the upper deck, and user laptop computers at the EPO console.

The Video Network provides video data directly to selected locations (as identified below) and through the other MCCS networks for access throughout the Observatory:

- Direct connections to all MCCS workstations (at MD, TO, EPO consoles)
- Direct connections at SI/PI Patch Panel (to allow SI team access to video without loading video traffic onto the Experimenter network)
- To the Experimenter network to provide user access to all observatory services and data via a single network connection.

MCCS workstations are provided only at the six operations stations -- Mission Director (MD), Science Flight Planner (SFP), two Telescope Operators (TO1/TO2), and two EPO (EPO1/EPO2) -- for the core in-flight operations personnel. All seats, however, provide more general access and services for laptop computers. Except for the SI rack seats, which are discussed below, the MCCS directly provides each seat with the following:

- one power connection for a laptop (115V/60Hz)
- one network connection to the Experimenter network for a laptop
- one audio connection via MADS

The SI rack seats also have connectivity for audio, power, and network and associated services, but these are indirectly provided via the PI patch panel (a.k.a., doghouse). Physical connectivity for these seats is described in the associated ICD (MCCS_SI_05).

Table 1 lists the major services available to users connected to MCCS networks.

Table 1. Available Network Services

Service	Notes
File transfers	Subject to inter-network firewalls
DHCP	Automatically assigns IP address to user computer
NTP time	Sync'd by GPS
Printing	Access to printer(s)
Remote communications	Subject to inter-network firewalls
Typical OS utilities	Subject to general network permissions
Video/DVDS	Provided via video network
NFS connection to Archiver	Provides access to observatory data
SCL access	Subject to command permissions

1.3 Telescope Assembly Description

The Telescope Assembly (TA) is the main external interface to the MCCS, and can be partially controlled by the SI. As shown in Figure 2, the TA is comprised of four major subsystems:

- TAMCP (TA Master Computer Processor): This computer is a communications hub that connects to the MCCS and to the internal TA subsystems. It passes commands from the MCCS to the other parts of the TA, and it collects housekeeping data (HK) from the TA subsystems and forwards that data to the MCCS.
- TASCU (TA Servo Control Unit): This computer controls the telescope's pointing capability. It supervises the Coarse Drive and uses fiber optic gyroscopes to control the inertial attitude of the Fine Drive via electromagnetic torque motors.
- Tracker: The Tracker computer makes use of the three visible light charge-coupled device (CCD) imagers to monitor TA pointing on the sky, and to

correct the residual gyroscope drift via handshaking with the TASCU. The three CCD imagers are as follows:

- Focal Plane Imager (FPI): The FPI is a 1024x1024 pixel CCD camera with a 9'x9' field-of-view (FOV) that shares the telescope's focal plane with the SI via the telescope's dichroic tertiary mirror. Since it views nearly the same field of view as the SI, it is generally used for set-up, pointing, and tracking. The FPI mechanism includes a back-focus adjustment to make this imager parfocal with the SI.
- Fine Field Imager (FFI): The FFI consists of a headring-mounted separate telescope and camera with a 1024x1024 pixel CCD and 67'x67' FOV. The FFI can be used in addition to or instead of the FPI for pointing setup and tracking. When using the fully reflective telescope tertiary mirror the FPI becomes unavailable, so the FFI would become the primary tracking imager.
- Wide Field Imager (WFI): The WFI consists of a headring-mounted separate telescope and camera with a 1024x1024 pixel CCD and 6^{o}x6^{o} FOV. The WFI is primarily used for sky-field recognition and monitoring sky rotation stability.

- SMCU (Secondary Mirror Control Unit): This computer controls the Focus Centering Mechanism (FCM) for TA collimation and focus, and also the Tip Chopping Mechanism (TCM) of the Secondary Mirror Assembly (SMA). The observer can control the SMCU through the MCCS via SCL. The SI can also control the secondary mirror tilt/tip position via analog voltages presented to the PI Patch Panel (see TA_SI_04). The chopper frequency may be controlled via a synchronizing transistor-transistor logic (TTL) signal from the SI.

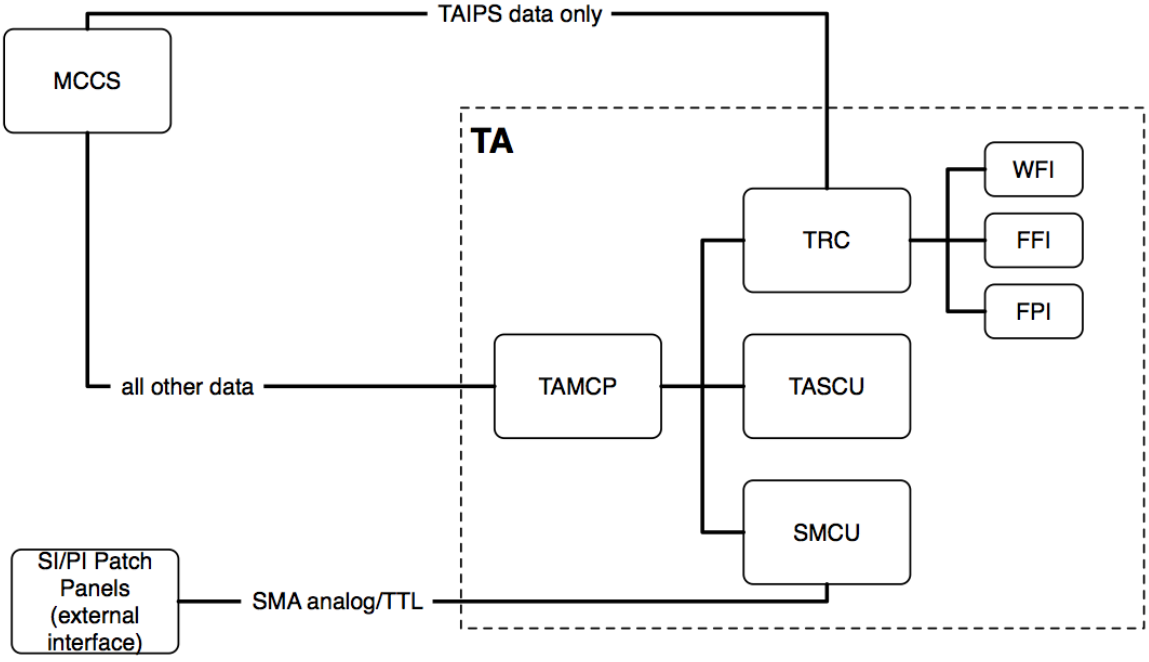

Figure 2. Telescope Assembly

1.4 Science Instrument System Description

For the purposes of this document, the SI system is considered to be one or more computer systems and associated equipment coupled to the SI detection devices and electronics controlled by the scientists using GUI interfaces, scripts, other computational command entry, and mechanical/electrical controls. The SI interfaces are comprised of the following:

- SI: the science instrument to be used for observing (e.g., FORCAST)
- Counterweight rack (CWR): rack of science equipment and/or simple weights used to counterbalance the telescope
- Principal Investigator (PI) racks: one or more racks of science equipment for use with the SI

The functional interface between the SI and the MCCS may be over one or multiple physical network connections. Science instrument equipment may be at any assigned node (IP address) on the Experimenter Network. All the generally available utilities on the network (e.g., printing and access to the MCCS Archiver) will be available to SI computers anywhere on the network. SI computers can interface with MCCS to monitor and control observatory functions via SCL sessions as described later in this document.

The MCCS provides a high-reliability disk array, the MCCS Archiver Subsystem, for storing observatory data products throughout flights. SIs are expected to transfer stored science data, and may transfer any ancillary data, during the flight to the MCCS Archiver. Science data should be written in FITS format as described in the DCS-SI-01 ICD. All SI-related files should be written in the /archive/si directory on the MCCS Archiver, which is a disk array which may be mounted to the user's computer with NFS or Samba. Any files and subdirectories written in this area can be organized as the SI Team sees fit. Note, however, that due to a DCS database limitation the full path plus filename for any given file should not exceed 200 total characters relative to the above "root". For example, a full path+file of /archive/si/mode1/file001.fits requires 30 characters, but only 19 are "charged" under the 200-character limit since those are relative to the /archive/si "root". If requested in advance of their flight series, the Archiver may also be configured to support other standard file-transfer protocols (e.g. sftp).

Based on data volume estimates for 1st generation instruments, 703 Gbytes of disk space is available on the MCCS Archiver per flight for SI teams to store their data. This equates to ~20 Mbytes per second continuous for almost 10 hours. If the SI team may need more than this they should contact observatory staff before their flight series to determine whether additional space may be made available.

The science data along with other observatory data collected during the flight are transferred to the ground-based data archive after each flight to ensure that no essential data is lost and to provide quick-look access shortly after each flight.

1.5 Standard Flight Data Products

Each flight of SOFIA produces a number of standard data products that are then saved in the archive. These standard products are listed in Table 2. For each product the table shows its source, format and file suffix, any associated standard, and accompanying notes; any standards noted as "custom" are defined in this ICD.

Table 2. Flight Data Products

Data Product / Source	Format / file suffix	Standard	Notes
Audio / MADS	*MP3 / .mp3*	*ISO 11172-3 & ISO 13818-3; www.iso.org*	*Recorded audio files from MADS. See Appendix D for further information.*
Housekeeping Data / MCCS	ark / .ark	custom	Housekeeping data via PIS. See Appendix F for further information.
Non-Housekeeping Data / MCCS	ark / .ark	custom	Data collected by DAS or other MCCS subsystems available from the MCCS Archiver (for example high-speed special-purpose data). See Appendix F for further information.
TA Images (raw) / DVDS	ark / .ark	custom	WFI/FFI/FPI imager data from DVDS/TAIPS. See Appendix F for further information.
TA Images (logged) / DVDS	FITS / .fits	fits.gsfc.nasa.gov	WFI/FFI/FPI images saved with the ta_wfi/ffi/fpi.log SCL commands. The FITS tool library is managed by NASA. See Appendix C for further information.
Video / DVDS	H.264 / .mp4	H.264/ MPEG-4	Streaming video data from DVDS/VPARS
SI data / SI	FITS / .fits	fits.gsfc.nasa.gov	SI-produced data; the format is defined in ICD DCS-SI-01. The FITS tool library is managed by NASA.
Log files / MCCS	Text / .txt	ASCII	MCCS log files and ancillary files from GUI & SI
Integrated command log file / MCCS	Text / .txt	ASCII	A single file containing all SCL commands & the final response to each command for a flight (each entry includes time, user & role, along with SCL command or final response). Refer to Section 3.3.2.
Integrated comment log file / MCCS	*Text / .txt*	*ASCII*	*A single file containing comments from all users for a flight (each entry includes time, user & role, along with comment)*
Alert & Alarm Log / MCCS	*Text / .txt*	*ASCII*	*A single file containing every alert and alarm message generated by MCCS.*

2. APPLICABLE DOCUMENTS

APP-AR-SPE-SW02-2012	MCCS Coordinate Transform Software Architectural Design: coord.convert
APP-DF-HWCI-SE51-2020	Mission Audio Distribution System (MADS) System Description Document
APP-DF-ICD-SE03-2011	ICD MCCS Network
APP-DF-ICD-SE03-2041	Intra-MCCS Functional ICD
APP-DF-ICD-SE03-2043	Intra-MCCS Physical ICD
PD-2003	Interface Reference Document
SOF-DA-MAN-OP02-2181	SOFIA Command Language (SCL) User's Manual
SCI-US-ICD-SE03-2023	ICD DCS-SI-01, Data Cycle System (DCS) ICD
SOF-AR-ICD-SE03-2029	ICD MCCS_SI_05, Principal Investigator Patch Panel to Principal Investigator Equipment Rack(s)
SOF-AR-SPE-SE01-2028	SOFIA Science Instrument System Specification
SOF-DA-ICD-SOF-1030	SOFIA Systems Interface Requirements Specification
SOF-DF-ICD-SE03-047	ICD TA_MCCS_F, TA to MCCS Functional
SOF-L3-ICD-SE03-048	ICD TA_MCCS_P TA to MCCS Physical
TA_SI_04	TA Chopper Processor / Principal Investigator Computer Direct Analog Interface

3. INTERFACE DESCRIPTION

The MCCS to SI physical interface is defined in MCCS_SI_05 ICD (SOF-AR-ICD-SE03-2029).

A primary functional interface provided by MCCS is the SOFIA Command Language (SCL). SCL is an ASCII protocol for users to monitor and control observatory systems. The definition of SCL is the subject of the bulk of this ICD.

The MCCS also provides a standard network environment with a range of services and available data to support science operations on the observatory. The following list summarizes the functional interfaces provided by MCCS:

(a) TCP/IP Ethernet network; allows user computers to intercommunicate and communicate with MCCS computers. Network services such as sftp, scp, http client and server, and X Windows client and server are permitted traffic on all observatory networks. In addition, all traffic is permitted on the Experimenter Network, where user computers normally reside and interconnect with other computers.

(b) Dynamic Host Configuration Protocol (DHCP) allows computers to receive an IP address automatically from the MCCS DHCP server.

(c) Monitor and control of observatory systems via the SOFIA Command Language (SCL)

(d) Color Printing

(e) Synchronization to standard observatory time (Universal Time synchronized to GPS satellites); synchronization is available to user systems via:

- Network Time Protocol (NTP) on all onboard networks
- IRIG-B time code interface at SI patch panel (see MCCS_SI_05 ICD)

(f) TA images, multicast in near real-time (see Appendix C)

(g) TA image archive (see Appendix C)

(h) MCCS Housekeeping data archive (see Appendices B and F)

(i) Mission audio distribution and recording (see Appendix D)

(j) Read/write access to MCCS Archiver RAID system for storage of SI data, and read access to other observatory data. Support is provided for NFS (Network File System), and for Samba which provides interoperability with Windows computers.

Before a computer can be connected to an MCCS network, it must undergo a security scan. In addition, in order to make use of the above MCCS services, a user must obtain one or more of the following:

1) An IP address for the user's computer (assigned by observatory staff). (This is needed if the user wishes to connect to MCCS networks and chooses not to use DHCP to automatically acquire a connection.)

2) An SCL account to allow direct monitoring and control of observatory functions.

3) User/group permissions to allow writing to or reading from the MCCS Archiver Subsystem.

Development and testing of observatory control from user computers must be done in collaboration with Mission Operations personnel prior to its use on the observatory. Such testing must include use of ground simulation system(s) to confirm proper commanding of mission-critical systems (such as the TA) before it may be used on the observatory.

The following sections provide details of the SOFIA Command Language.

3.1 SOFIA Command Language Network Connection Protocol

In order to carry on an SCL session, either one bidirectional socket or a combination of a "write-only" and a "read-only" socket must exist between MCCS and an external system. Thus, in order for an SCL session to be started, one of the following must occur:

(a) External system calls for a single "bidirectional" socket: The external system connects via TCP/IP to a well-known MCCS IP address and port (port # 6555). When the external system issues a normal login command, the MCCS will return responses over the same TCP/IP connection. Thus, the single TCP/IP socket is used for bidirectional communication.

(b) External system calls for two “one-way” sockets: The external system connects via TCP/IP to the well-known MCCS IP address and port (as defined above). The external system then issues a login command which includes the "port" keyword to tell MCCS which port on the originating external system to connect to for all output from MCCS. MCCS then connects to the indicated port and sends the response to the login command. Note that both sockets are in fact capable of bi-directional communication, but that by convention the communications only occur in one direction on each socket.

While the simpler single-socket connection protocol ((a) above) is likely to be adequate for most uses, the two-socket connection is provided for users who prefer it, for performance or other reasons.

If MCCS detects a lost socket connection, MCCS closes all sockets related to that connection and terminates any active SCL commands which originated from that connection. The external system is responsible for establishing a new connection to MCCS if/when it so chooses. In addition, MCCS issues an informational alert message to notify users and provide a record of the lost connection.

3.2 Sessions and Roles

In order to initiate an “SCL session” with MCCS each user will log in using an assigned name and role. Command access will be restricted according to this role.

For example, for a science instrument computer to login in, it may use the command:

```
1 login user=<instrument> role=instrument password=<pwd> <CR>
```

where:

```
<instrument> is the name assigned to the instrument from the
    users.xml file in MCCS
<pwd> is the password for this user in this role
```

MCCS supports multiple (at least 40) simultaneous SCL sessions, thereby allowing SI teams, mission operations staff, and other onboard users to monitor and control observatory subsystems as permitted by their roles.

Table 3 lists common user roles.

Table 3. Session Roles

Name of Role	Description of Role
ifd	In-Flight Director session *(to become 'md' in a future release)*
md	*Mission Director session*
to	Telescope Operator session
instrument	Instrument computer session
pi	SI team sessions
gi	General Investigator sessions
facil	EPO Facilitator sessions
epo	EPO class/guest sessions
visitor	VIP visitor session
fp	Flight Planner session
wvm_system	Water Vapor Monitor session

3.3 Communication and Data Exchange

SOFIA Command Language (SCL) allows scientists and other observatory personnel to control the science experiment and to access other MCCS software functions, controls, and services from onboard PI computers and observatory workstations. Commands are entered by activating previously composed scripts (XML files) and through command-line interfaces and GUI selections. These SCL commands are passed to MCCS where validation and translation into the appropriate direct commands to the TA or other subsystems occur.

3.3.1 SOFIA Command Language Protocol

This section describes the SCL command and response protocol. Note that a separate document, the SOFIA Command Language (SCL) User's Manual, provides an overview of SCL and how it may be used to observe on SOFIA.

SCL commands are ASCII and are not case sensitive – the MCCS converts all commands and keywords to lower case before interpreting them. Note however that string values which are quoted (i.e. enclosed in double-quote marks) are case sensitive. For example, `name="FooBar"` within a command would treat the value of name as *FooBar*, whereas the same command with `name=FooBar` (i.e. without quotes) would treat the value of name as *foobar*.

By default, SCL responses are in ASCII format - however, a binary SCL response format is also available (see section 3.3.1.3, Binary Responses).

All SCL commands from the SI or other users external to MCCS begin with a user designated message ID, which is a positive 32-bit integer value. This ID will be included on all responses to the command so that the user knows which responses are associated with which command (since the user may have multiple commands running at the same time). For this reason, the user should specify a different ID with each command they issue, since otherwise it may become difficult to know which responses are associated with a specific command. Of course, a user may re-use a previously specified ID if the command associated with that ID has completed. Also, two different users may issue commands with the same ID without any problems since each user session has its own command environment (i.e., user sessions do not share commands – each session has its own set of commands that execute independently from commands issued by other user sessions). This means that all command responses sent to a particular user session will be associated with a command issued by that session.

SCL command responses take one of three formats:

1) BINARY (pure binary): binary header, data, and footer (defined in Section 3.3.1.3);
2) LEGACY (pure ASCII): no header, new line indicates end of message; and
3) FORMATTED (formatted ASCII): no header, new lines may be embedded, will be terminated by a new line.

The default type is LEGACY. Telnet users can use any one of the three. To override the LEGACY format, a user can do one of the following:

1) In the user's XML file, set the default format (attribute resp_format) to BINARY or FORMATTED. The user's XML file is named [user name].xml and it resides in the '/modstor/mccs/config' directory. This is an optional file for each user which contains initial SCL commands to be run whenever that user logs in.
2) Override the default set in the user's XML file by specifying the format at the end of a command: "`resp_format=<type>`" where `<type>` is LEGACY, BINARY, or FORMATTED". (Applies only to that command.)
3) Issue a set command to change the format so that all commands issued after that point receive responses in the specified format: "`set resp_format=<type>`" where `<type>` is LEGACY, BINARY, or FORMATTED".

3.3.1.1 Command Syntax

The notation used in describing the command syntax for the SOFIA Command Language is shown in Table 4.

Table 4. Syntax Notation

Symbol	Description
<>	Encloses a symbolic token.
{}	Encloses a token which may be repeated zero or more times.
[]	Encloses an optional token.

An SCL command takes the following form.

<msg id> <command> { <keyword> = <value> [({ <keyword> = <value> [({ <keyword> = <value> […] })] })] } <term>

The terms in the above are defined in Table 5 and are separated by white space. White space is any combination of spaces and/or tabs. For formatted output, white space also includes carriage returns and line feeds.

SCL commands have a maximum length of 8,191 characters, including the terminator at the end of the command.

In addition, an SCL command can have a maximum of 1000 tokens.

To abort an active command the user may issue the "cancel" command.

<msg id> cancel cmdid=<msg id of command to cancel> <term>

The system will respond with a '<msg id of cancel command> #' response, which indicates the specified command was aborted. Note that many commands complete rapidly, so that there may be nothing to cancel by the time the command is received and processed.

Table 5. Command/Response Tokens

Token	Description
<msg id>	An arbitrary positive number used to identify a specific instance of a command. This number could be incremented by 1 for each subsequent command and be used for easy tracking of the sequence of commands. The value must be a positive 32 bit integer (i.e. from 1 to 2147483647). When responses are sent for the command, this number is used for the response message ID (see the next two sections).
<command>	Unique command name, including routing if necessary.
<keyword>	A keyword defined for the given command.
=	Equal sign (required after every keyword).
<value>	A value of a type compatible with the keyword it is assigned to. Note that a value may also contain the following: […] Defines lists of values, such as list=[height width

Token	Description
	depth], or bounds an ordered set or array type value command parameter (e.g., ta_tsc.tsc_fd_pos_abs fd_des_pos_abs_quat=[0.0 0.0 0.0 1.0] for its quaternion position parameter). [[…][…]…] Defines nested lists/arrays such as lists of points. Note that each sub-list counts as a token. These nested lists can be nested to any depth and for arbitrary lengths, which allows for possible future support for objects like 3x3 matrices (e.g., Direction Cosine Matrices (DCM)).
( )	Parentheses delineate keyword-value pairs, in which the values represent attribute(s) of the value preceding the parentheses.
…	Indicates that the pattern "({<keyword>=<value>})" may be repeated indefinitely. Since parenthesized keyword-value pairs represent attributes, this expresses the idea that attributes may have attributes, which theoretically (though not practically) could go on ad infinitum.
<term>	A character indicating end of a command or response. The MCCS will return <LF> at the end of all responses. It will accept <CR>, <LF>, or any sequence of <CR> and <LF> characters for commands.

Examples:

```
101 coord.position action="ignore"
    name="ref_pos"(ra=02h16m46.39s dec=57d01m45.65s
    centroid=trc_aoi1 rof1=trc_aoi2 rof2=trc_aoi3 inertial=yes
    equinox=J2000)<CR>
102 nod.define amplitude=90.00 coord_sys="erf" pos_angle=45.00
    profile=2 track_pos_a="nodtracka"
    track_pos_b="nodtrackb"<CR>
103 nod.goto pos=a<CR>
104 get list=[sibs.ra sibs.dec]<CRLF>
105 nod.next<CRLF>
106 sma.chop frequency=5 start=no<LF>
107 coord.set comment="ready to observe now"
    file="log1.txt"<CRLF>
108 ta_pos.pattern extra_points=2 start_delay=5 rate=100
    coord_sys=tarf points=[[0 0][50 0][100 0]] <CRLF>
```

3.3.1.2 ASCII Response Syntax (Legacy or Formatted)

Each command will have one or more responses. There will be zero or more intermediate responses followed by a single final response. The response to a command takes the following form (see also Table 5 for definitions). Each response has a message ID and a response type as described in Table 6.

<msg id> <respType> [{<keyword>=<value> [({<keyword>=<value>})] }]<term>

Table 6. ASCII Command Responses

Token	Description
<msgID>	Numeric ID from the command that this is a response to.
<respType>	Single character describing the type of response **: (**final response) success. **E (**final response) error in one or more data values: One or more command arguments or attributes has failed a limit check, failed unit conversion, or does not match the expected representation, e.g., a number was expected, but the user supplied a name string. Command not processed. **S** (final response) syntax error: The command does not follow the syntax given in Section 3.3.1.1, or the command destination or command name is not valid or cannot be found, or a given argument or attribute name is not valid for the specified command name. Command not processed. **F** (final response) A command failed at the destination. **#** (final response) command aborted by the user. **A**(intermediate response) command acknowledgement: Command has passed all basic syntax and data checks. Note that following this check, the command is routed to its destination for execution, where further syntax or data checks on the command may be performed, so an E or S response (see above) may follow an A response. Also, note that an E or S response may be sent without an A response. **I** (intermediate response) for a normal intermediate response (e.g. continuing responses to a subscribe command). **W** (intermediate response) for a warning. **D** (intermediate response) for developer debug information messages. **N** (intermediate response) for developer debug warning messages (i.e. something is wrong).

MCCS will always generate upper case respType.

After a given command has sent an ‘A’ response, any number of intermediate responses may follow, concluded by a final response (i.e., ‘:’, ‘E’, ‘S’, ‘F’, or ‘#’).

Commands that do not pass their basic syntax check will return an ‘S’ final response. Those that do not pass their data checks will return an ‘E’ final response. As is mentioned in the table above, an ‘S’ or ‘E’ response may also be returned after an ‘A’ response if the command fails any specific syntax or data checks performed at the destination. Specific syntax or data checks involve those things that only the destination knows about the command, e.g., if the given arguments make sense together (even though they may be syntactically valid) or how to perform units conversions for items with dynamic conversion factors (e.g., converting from pixels to millimeters when pixels are specified according to the plate scale of a given SI).

The ‘D’ and ‘N’ responses are provided for programmer convenience. They contain response messages that give the programmer more insight into what is happening internally with the software.

In order to allow the user to control the verbosity of intermediate responses, a ‘verbose’ environment variable/global argument allows the user to specify one of three levels for intermediate responses:

1) NONE: Final responses and ‘A’ responses will be sent to the user. No other intermediate responses will be sent to the user, except for intermediate responses from the session subscribe and watch commands.
2) NORMAL: Final responses and ‘A’, ‘I’ and ‘W’ intermediate responses will be sent to the user. (The intermediate response types included are those of likely interest to most users.)
3) ALL: All responses will be sent to the user (same as NORMAL but with addition of ‘D’ and ‘N’ responses).

The default verbosity is NONE. Telnet users will probably want to use the NORMAL level. To override the NONE verbosity, a user can do one of the following:

1) In the user’s XML file, set the default verbosity (attribute verbose) to NORMAL or ALL, or
2) Override the default by specifying the verbosity at the end of a command: "verbose=<verbosity>”, where <verbosity> is NONE, NORMAL, or ALL. (Applies only to that command.) Or
3) Issue a set command to change the verbosity so that all commands issued after this point will receive intermediate responses according to the specified verbosity:

```
"set verbose=<verbosity>”, where <verbosity> is NONE, NORMAL, or
    ALL
```

Note that the ‘A’ response type cannot be disabled.

The keywords in the response syntax are actually labels for the values. They may be turned off by setting the showLabels keyword to ‘no’. When this is done, only the values are shown.

In all responses with returned values, if showLabels is set to yes, the string “keyword=” will precede any returned value, where “keyword” is replaced with a relevant label. This label will vary depending on the command issued, but as a general rule, if the command is a session subscribe, get, or set command, the labels will correspond to the data elements specified in the command. For example, if the user issues the following subscribe command:

```
5 subscribe list=[das.ins_1_12hz.hybrid_lat
    das.ins_1_12hz.hybrid_lon] showLabels=yes<CR>
```

Then the response will contain the labels “latitude=” and “longitude=” before the values for those data elements:

```
5 I das.ins_1_12hz.hybrid_lat=34.6135523790000
    das.ins_1_12hz.hybrid_lon=-118.0754739010000<LF>
```

Thus, an SI may parse responses for information of interest by scanning for the relevant keywords. Conversely, if showLabels were set to no, then the above response would take the following form:

```
5 I 34.6135523790000 -118.0754739010000<LF>
```

For commands that send general response messages back to the user (i.e., responses that do not involve data elements), the label will typically be the string "message=<string>". This response is unaffected by the setting of showLabels. For example:

```
7 : message=″Command completed″<LF>
```

Finally, please note that some command responses may not have any values associated with them, i.e., they may only have a message ID and response type. For example:

```
8 :<LF>
```

The following are examples of responses to commands in Section 3.3.1.1 (with the session defaults set to showLabels=yes and verbose=all). Note that responses do not necessarily come in the order that the commands were issued.

```
101 A <LF>
102 A <LF>
102 : <LF>
101 : <LF>
103 A <LF>
104 A <LF>
104 : sibs.ra=5h46m53.93s sibs.dec=-9d23m43.06s<LF>
105 A <LF>
106 A <LF>
106 : <LF>
107 A <LF>
107 : <LF>
103 : message="All sub-commands completed successfully\<LF>
-7 ta_tsc.tsc_fd_pos_abs fd_des_pos_abs_quat=
    [-0.005480571642639572 -0.001715481526798551
    0.999983510092842]\<LF>
-7 W message=\"SEQ_COUNT_CHG\"\<LF>
\<LF>
103 : DONE\<LF>
"<LF>
105 : message="All sub-commands completed successfully\<LF>
-7 ta_tsc.tsc_fd_pos_abs fd_des_pos_abs_quat=
    [0.00548061714854261 -0.0017154971348523278
    0.999982931577133]\<LF>
-7 W message=\"SEQ_COUNT_CHG\"\<LF>
\<LF>
105 : DONE\<LF>
"<LF>
```

For example command 103 has two response messages above – the 'A' acknowledgment and the ':' final (success message). Note that the ':' response to

command 103 includes an optional message=<string> in addition to the "103 :" at the start of the final response. Response strings can extend over multiple lines (with '\' preceding each <LF> or quote sign (") embedded within the string). In this example, the value of the string following "message=" is the concatenation of the command the MCCS issued to the TA to cause the requested action, and the resulting response from the TA. Such internally-generated commands use negative msgID values to differentiate them from user-issued commands (which only use positive msgID values).

As another example, in the response to command 104 the position may not have been the correct final position (based on the examples in the previous section) because the user's program did not wait until the final response was received before issuing the get command.

Unimplemented commands or arguments result in an Error (E) response. For example, sending the following two commands to MCCS:

```
108 ta_pos.stow<CR>
109 ta_pos.track limb1=trc_aoi6 limb2=trc_aoi7
```

results in the following responses:

```
108 A<LF>
108 E message="stow is not yet implemented"<LF>
109 A<LF>
109 E message="track has unimplemented arguments"<LF>
```

Within an error or warning response (respType = E, S, F, W, or N), the MCCS includes after the respType a quote-delimited text string describing the error. If a single SCL command sends multiple commands internally and fails because one or more of them fails, the final command response contains the roll up of all error responses. Additional information, such as error number, may be included as a keyword-value pair after the error string.

The actual data returned with responses will evolve throughout the life of the observatory. It is recommended that programs and scripts not assume a fixed data content in the response, as this field will generally be used by the MCCS to report various values which might be of interest. If particular data values are of interest, the commanding system should specifically request those values. If the user wants to be aware of the returned information from commands, the developer of that user's system may wish to capture the full text of those responses for display or logging, allowing the user to review the responses manually.

Note that subsequent sections of this ICD mostly omit the terminator (e.g. <CR> or <LF>) on commands and responses, although these are always a required part of SCL syntax.

3.3.1.3 Binary Response Syntax

Binary responses are of the form:

<SOH> <respLength> <bmsgID> <respType> {<status> [<bdataLength>] <binaryData>} <EOT>

where <bdataLength> is only present for indefinite length data types. A given binary response may contain zero or more data elements, where each data element is specified by the <status>, <bdataLength>, and <binaryData> fields. All values in the binary response are in big-endian byte order.

Table 7. Binary SCL Command Responses

Token	Description
<SOH>	Start of Header sequence, the ASCII SOH character [0x01]
<respLength>	8 bytes containing, in binary (long long), the number of bytes for all status, bdataLength, and binaryData elements combined.
<bmsgID>	4 bytes containing, in binary (int), the msgID number submitted by the user with the command (see Section 3.3.1.1, Commands).
<respType>	Same as in Table 6 (1 byte)
<status>	1 byte indicating the status of the data element. A value of 0 indicates the element is set to something (i.e., is valid), a value of 1 means the element is "NotSet" (i.e., has not been populated with any data yet), and a value of 2 means the element is "NotFound" (i.e., the data element was not found in the MCCS data tree). If this byte has a value of 1 or 2, the bdataLength and binaryData fields are omitted for the data element (since no data exists for the data element).
<bdataLength>	4 bytes containing, in binary (int), the length of <binaryData> in bytes. This field is included in front of variable length data types (STRING and BINARY) only.
<binaryData>	Variable number of bytes (depends on the representation of the data item), that contains the value of the data element in binary format (as it appears in memory).
<EOT>	End of Transmission sequence, the ASCII EOT character [0x04]

Note that all binary responses have a header of 14 bytes and a footer of 1 byte. The total response length is therefore 15 + value of <respLength>, since <respLength> contains the sum of all the <status>, <bdataLength>, and <binaryData> elements present in the response.

Typically, binary responses will be used in conjunction with subscribe commands, providing a more efficient way to get data from the MCCS. For example, if the user issues the following subscribe command:

```
5 subscribe list=[latitude longitude heading] resp_format=binary
```

The subsequent binary responses would each contain three data elements, one for each of the data items specified in the list argument. So the fields <status> and

<binaryData> would each be repeated three times within the response (note that the <bdataLength> field is not included since these data elements do not have a representation of STRING or BINARY). The data elements will be listed in the same order they were specified in the command, so latitude will be first, followed by longitude and then heading.

To illustrate what will be contained in a typical binary response, assume that each of the data items specified in the command above is a FLOAT8 type (see Table 8 below). The subsequent binary response would contain the following:

1. A 14 byte header, consisting of a 1 byte character (ASCII SOH character), 8 bytes for the response length (number of bytes in all status, bdataLength, and binaryData fields combined)- in our example, response length will have a value of 27 (each data element has a 1 byte status field and 8 byte binaryData field, and we have 3 data elements), 4 bytes for the message ID (in our case, the value here will be 5), and 1 byte for the response type (in our case, the value will be 'I').
2. 27 bytes of data, consisting of three data elements. Each data element consists of a 1 byte status field (containing a value of 0 since the data exists) and an 8 byte binaryData field (containing the binary form of the data). The binaryData field is 8 bytes long since the representation for each of our data items is FLOAT8.
3. A 1 byte footer, consisting of a 1 byte character (ASCII EOT character).

As a more complex example, let's assume we've subscribed to three data elements with the following data types: FLOAT4, STRING, and INT2. The FLOAT4 data element has not been populated with any data yet, the STRING data element has a value of "hello", and the INT2 data element has a value of 10. The subsequent binary response would contain the following:

1. A 14 byte header, consisting of a 1 byte character (ASCII SOH character), 8 bytes for the response length- in our example, response length will have a value of 14 (see the next step for how this is calculated), 4 bytes for the message ID, and 1 byte for the response type.
2. 14 bytes of data, broken down as follows:
   i. FLOAT4 data element- This data element only has the 1 byte status field, which is set to 1 since the data item is not populated yet. There is no binaryData field since no data exists for the data item.
   ii. STRING data element- This data element has the 1 byte status field, the 4 byte bdataLength field (which contains a value of 5), and a 5 byte binaryData field (the string "hello" contains 5 characters, each of which takes up a byte of memory).
   iii. INT2 data element- This data element has the 1 byte status field, and a 2 byte binaryData field (INT2 is a signed 2 byte integer).
3. A 1 byte footer, consisting of a 1 byte character (ASCII EOT character).

Table 8 describes the binary data types supported by MCCS.

Table 8. Binary Data Types

Type	Properties
STRING	Sequence of standard 8 bit ASCII characters with no null termination
BINARY	Serialized byte sequence
BINARRAY	Array of scalar values; the type of data and number of elements are specified in the XML file defining the data; data type can be any fixed-size scalar type (i.e. BYTE, INT2/4, UINT1/2/4, FLOAT4/8, BOOL4, SEXA8, or TIME8)
BYTE	Signed 1 byte integer
UINT1	Unsigned 1 byte integer
INT2	Signed 2 byte integer
UINT2	Unsigned 2 byte integer
INT4	Signed 4 byte integer
UINT4	Unsigned 4 byte integer
BOOL4	Represents a 4 byte logical value (true (1) or false (0))
FLOAT4	4 byte (single-precision) real number in IEEE format
FLOAT8	8 byte (double-precision) real number in IEEE format. Note that this is the preferred data type for storing MCCS timestamps (number of seconds since 1970).
SEXA8	A FLOAT8 type that supports sexagesimal notation (e.g., 15d14m37.45s).
TIME8	TIME8 is one of two alternative timestamp data types, the other being FLOAT8. TIME8 consists of two 4 byte integer parts, the first part representing the number of seconds since 1970 and the last part representing the number of nanoseconds since the beginning of the second. (FLOAT8 is the preferred timestamp data type.)
COMPLEX8	Contains two FLOAT4 numbers, which are used to represent a single complex value. The real component is represented by the first FLOAT4, while the imaginary component is represented by the second FLOAT4.
UCOMPLEX8	Contains two FLOAT4 numbers, which are used to represent a complex value on the unit circle. The real component is represented by the first FLOAT4, while the imaginary component is represented by the second FLOAT4.
COMPLEX16	Contains two FLOAT8 numbers, which are used to represent a single complex value. The real component is represented by the first FLOAT8, while the imaginary component is represented by the second FLOAT8.
UCOMPLEX16	Contains two FLOAT8 numbers, which are used to represent a complex value on the unit circle. The real component is represented by the first FLOAT8, while the imaginary component is represented by the second FLOAT8.
QUAT16	Contains four FLOAT4 numbers, which are used to represent a single quaternion value. The real component is given first, followed by the three imaginary components.
UQUAT16	Contains four FLOAT4 numbers, which are used to represent a unitary quaternion value. The real component is given first, followed by the three imaginary components.
QUAT32	Contains four FLOAT8 numbers, which are used to represent a single quaternion value. The real component is given first, followed by the three imaginary components.
UQUAT32	Contains four FLOAT8 numbers, which are used to represent a unitary quaternion value. The real component is given first, followed by the three imaginary components.
PGAQUAT32	Four 8-byte floating point components: three bivectors and the real part. So if the four coefficient numbers are (a, b, c, d), then the PGA quaternion could be expressed as: aI + bJ + cK + d, where I is the unit bivector for the y-z plane, J is the unit bivector for the z-x plane, and K is the unit bivector for the x-y plane.

Type	Properties
UPGAQUAT32	Unit PGA quaternion (see PGAQUAT32 for further details).
INTVECTOR8	Contains two INT4 numbers, which are used to represent a 2 dimensional vector.
INTVECTOR12	Contains three INT4 numbers, which are used to represent a 3 dimensional vector.
VECTOR16	Contains two FLOAT8 numbers, which are used to represent a 2 dimensional vector.
VECTOR24	Contains three FLOAT8 numbers, which are used to represent a 3 dimensional vector.
UVECTOR24	Contains three FLOAT8 numbers, which are used to represent a 3 dimensional unit vector.
MATRIX72	Contains nine FLOAT8 numbers, which are used to represent a 3x3 matrix. The first three numbers represent the first row of the matrix, the second three numbers represent the second row of the matrix, and the final three numbers represent the third row of the matrix.
UMATRIX72	Contains nine FLOAT8 numbers, which are used to represent a unitary 3x3 matrix. The first three numbers represent the first row of the matrix, the second three numbers represent the second row of the matrix, and the final three numbers represent the third row of the matrix.

See Section 3.3.1 for the various methods of requesting binary responses.

Unlike ASCII responses, binary responses do not contain any keywords (labels) or attributes (e.g., units=sec). STRINGs and BINARY data types are sent as a sequence of bytes, without padding or null termination.

The data type of each housekeeping data item is defined in Appendix B.4. Please note that keeping track of the sequence of data elements in a binary response is left to the commanding system, but, as is described above, this sequence will always match the order of the data elements given in a subscribe or get command's list argument. Since the user must know this sequence up front, however, this means that the 'trigger=all' feature of the subscribe command should not be used in conjunction with binary responses (since there is no way for the user to tell which item(s) were updated).

3.3.1.4 ASCII Value Format

Table 9 describes the format for argument and housekeeping data values in SCL commands and ASCII responses. Note that data items of BINARY or BINARRAY types are not available in SCL commands or ASCII responses, but are only available in binary format (described in Section 3.3.1.3). The data type of each command argument and housekeeping data item is provided in Appendices A and B.

Table 9. ASCII Value Formats

Type	Examples	Description
Integer	1234 -500	Specific integer formats supported within MCCS are: BYTE, UINT1, INT2, UINT2, INT4, UINT4 (Table 8, Binary Data Types, has information about each)
Real	12.34 1.234e-10	Floating Point values: FLOAT4, FLOAT8

Type	Examples	Description
List	[posA posB] [56 89] [0 0 0 1] [[0 0] [300.5 400.6]]	Values of any multi-element type listed in Table 8 (complex, quaternion, vector, and matrix) or a list of values of a scalar, or multi-element type.
"true/false" option response	yes/no true/false on/off open/close 1/0	Boolean value: BOOL4
Text string	"One Two Three Four" A9520 ThisIsAWord so_is_this	STRING Strings may be quoted (i.e. enclosed in double-quote marks) or unquoted. However, unquoted strings must start with a letter and must not contain blanks, tabs, unprintable, or reserved characters (such as brackets, parentheses, or commas). Unquoted strings are folded to lower case by MCCS, whereas quoted strings are retained exactly as entered. Most escape sequences from C++ and Java conventions are also supported within a quoted string.
Sexagesimal	Time: 12:13:14.5 RA & Dec: 12h13m14.5s –12d13m14s	MCCS converts SEXA8 values to FLOAT8 before storing them internally, and converts them back to sexagesimal format before output. Sexagesimal format is normally used for time, RA, or Dec (respectively: use colons for time, hms for RA sidereal time, and dms for angles). Note: all SEXA8 values may be entered as FLOAT8 values as well.
Timestamp	Default format ("%.3ymdz"): 2011-07-28T02:58:53.280Z Timestamp in the name of file created by MCCS: 20110728T025853Z (e.g.	Timestamps may be stored in MCCS as FLOAT8 or TIME8 data types (as described in Table 9). Appendix B.4 (column "Format") specifies the default format for each housekeeping data item. Format "%.3ymdz" gives year, then month, day, hours, minutes, seconds (to the millisecond) of UT in ISO 8601 format. This is the default format for timestamps of FLOAT8 data type provided by MCCS in SCL and in log files. It is recommended SI teams start each line in their own log files with time in the same format; this allows MCCS and SI logs to be merged and sorted to show event sequences – potentially valuable for debugging purposes. In the names of some files created by MCCS, the '-' and ':' characters, as well as the milliseconds part of the above timestamp

Type	Examples	Description
	'cmdlog_20110728T025853Z.txt').	format above are omitted. This format also complies with the ISO 8601 standard. Note that FLOAT8 is the preferred data type for MCCS timestamps since it is a single scalar value (seconds since 1970) and provides more options for how time is displayed. For example, with FLOAT8 timestamps, the user may override the default "%.3ymdz" format with a format such as "%lf". The user may also control time units as listed in Table 13. For example '1 get list=time (format="%lf" units=hours)' will return the current time as number of hours since 1970.
	FLOAT8 timestamps with a double precision format (e.g. "%.3lf"): 1312796160.533	

3.3.2 Integrated Command Log

Commands and their final responses from all SCL sessions are written by MCCS to a single integrated log file available for viewing by users. A new file is created every time MCCS is started. The file name is 'cmdlog _<timestamp>.txt' and is located in the '/modstor/mission_data' directory. (The <timestamp> part of the file name is defined in Table 9.)

The only exceptions to writing the exact command or response to the file are:

- *For 'login' commands, MCCS replaces the text of the password value with '*' before writing.*
- Non-printing characters in a command or response are replaced before output with the escape sequences described in Table 18 to keep each message readable and to facilitate parsing. All characters listed in Table 18 except backslash and double quote, which are left unchanged, will be replaced with the listed escape sequences.

As an example, when MCCS starts it creates file 'cmdlog _20160728T025709Z.txt' in directory /modstor/mission_data. The file starts with a header line and is followed by the chronological sequence of SCL commands, final responses, user connection and disconnection events, and SessionManager start events. Each line starts with a timestamp, and ends with a single linefeed. Note that all lines are written to the file as they occur so that they are immediately accessible from the MCCS Archiver (e.g. via NFS).

```
2017-04-28T21:00:41.699Z naming_server h ***** Integrated command log
    file created *****
2017-04-28T21:00:48.622Z session_to S SessionManager started on
    session_to
2017-04-28T21:00:57.005Z session_md S SessionManager started on
    session_md
```

```
2017-04-28T23:01:35.920Z session_md.session_1 + 10.4.1.4:32873
    connected to session_md:6555

2017-04-28T23:01:36.409Z session_md.session_1:1 c 1 login
    user=tester_to role=to

2017-04-28T23:01:36.511Z session_md.session_tester_to_to_1:1 r 1 :

2017-04-28T23:01:39.227Z session_md.session_tester_to_to_1:2 c 2 get
    list=[ta_scs.scs_status.scs_align_tcm_angle_offset] showlabels=False

2017-04-28T23:01:39.227Z session_md.session_tester_to_to_1:2 r 2 :
    0.000000

2017-04-28T23:01:39.330Z session_md.session_tester_to_to_1:3 c 3 get
    list=[ta_scs.scs_status.scs_align_tcm_scale] showlabels=False

2017-04-28T23:01:39.330Z session_md.session_tester_to_to_1:3 r 3 :
    1.000000

2017-04-28T23:02:03.998Z session_md.session_tester_to_to_1:4 c 7108
    ta_pos.rewind los_margin=0.0 los=1.5

2017-04-28T23:02:08.343Z session_md.session_tester_to_to_1:4 r 7108 :
    message="All sub-commands completed successfully\n -41
    ta_trc.trc_los_limit_set trc_des_tarf_u= 1.06792e-05
    trc_des_tarf_v=1.02167e-05 trc_des_tarf_w= 1 trc_des_delta_rof=
    1.17382\n -41 W message=\"SEQ_COUNT_CHG\"\n 7108 : DONE"
```

The timestamp at the start of each line follows a standard format (ISO 8601) which allows other log files which start with times following the same format to be merged chronologically with this file. Also, each command or response is preceded by a "session name" field which includes the MCCS host name, the user name, the role of the command originator (for details, see 'session_name' in Table 17 in Section 4.2.3, Non-Housekeeping Data Items), then a colon (:) followed by a unique integer for every command and its final response within that session. Finally, each line is preceded by a one-letter identifier of the type of entry: 'h' for the header (first line of file), '+' for a user connecting to a session network socket, '-' for a user disconnecting from a session network socket, 'S' for a Session Manager starting, 'c' for a command, and 'r' for a response.

3.3.3 User Comments and the Integrated Comment Log

To support in-flight communications users need the ability to publish their comments and monitor those of others. The MCCS provides this capability via the coord.set command by setting comment=<string> and file=<filename> keywords. With this functionality user-defined comments are inserted as strings into the HK data stream where they (1) can be subscribed to by other users for monitoring purposes, and (2) are archived by the MCCS as part of the overall HK stream; in addition, each user's comments are saved in the user-defined file in their own working area (mission_data/mcs_users/<username>/comment_files). An example comment command and its responses follows:

```
13 coord.set comment="ready to observe now" file="log1.txt"
13 A
13 I  message="filename=
    /modstor/mission_data/mcs_users/testuser/comment_files/log1.
```

```
    txt"
13 : message="DONE"
```

All comments entered with the 'coord.set comment=<string> …' command or with the 'session.comment' command are written by MCCS to an integrated comment log available for viewing by onboard users. A new file is created every time MCCS is started. The file name is 'comlog_<timestamp>.txt' and is located in the '/modstor/mission_data' directory. (The <timestamp> part of the file name is defined in Table 9.)

As an example, when MCCS starts, it creates file 'comlog_20110728T025711Z.txt' in directory /modstor/mission_data. The file starts with a timestamped header line, followed by all comments. Note that all lines are written to the file as they occur so that they are immediately accessible.

```
2011-07-28T02:57:11.588Z  daemon_node_a.log_daemon  h  ***** Integrated
    comment log file created - PIS software revision = 341a, Mission
    ID=2011-07-28_GR_F073 *****

2011-07-28T02:59:44.653Z  session_node_b.session_ckaminski_md_0  t
    ckaminski md : Power transfer 5 minutes ago.

2011-07-28T03:25:46.804Z  session_node_b.session_ckaminski_md_0  t
    ckaminski md : Taxi now

2011-07-28T03:35:53.274Z  session_node_b.session_ckaminski_md_0  t
    ckaminski md : No problems with power transfer we had the last two
    flights.  Perhaps switching the power source for the FMI cured the
    switching issue.  No reboots of the MCCS needed at this time.
```

As with the Integrated Command Log, the timestamp at the start of each line follows a standard format (ISO 8601) which allows other log files which start with times following the same format to be merged chronologically with this file. This is followed by a "session name" field which includes the MCCS host name, the user name, and role of the comment originator (for details, see 'session_name' in Table 17 in Section 4.2.3, Non-Housekeeping Data Items). The third field is a one-letter identifier of the type of entry: 'h' for the header (first line of file) or 't' for comment lines. This is followed by the user name, role, a colon (:), and finally, the comment itself.

Any line-feeds, carriage-returns, and non-printing characters in the comment text are replaced by '\nnn', where nnn is the octal value of the character, with leading zeroes included (e.g. line-feed would be shown as \012).

3.3.4 Alert and Alarm Log

All alert and alarm messages are written by MCCS to an alert and alarm log file available for viewing by onboard users. A new file is created every time MCCS is started. The file name is 'aalog _<timestamp>.txt' and is located in the '/modstor/mission_data' directory. (The <timestamp> part of the file name is defined in Table 9.)

As an example, when MCCS starts, it create file 'aalog _20110728T025712Z.txt' in directory /modstor/mission_data. The file starts with a timestamped header line, followed by all alert and alarm messages. Note that all lines are written to the file as they occur so that they are immediately accessible.

```
2011-07-28T02:57:11.588Z  daemon_node_a.log_daemon  h  ***** Integrated
    alert and alarm log file created - PIS software revision = 341a,
    Mission ID=2011-07-28_GR_F073 *****


2011-07-28T02:59:44.653Z  daemon_node_a.log_daemon  a  INFO \177 ta
    \177 ID=TA_CONNECT \177  \177  \177 MCS connected to TA \177 [2011-
    07-28T02:59:43.865Z] \177

2011-07-28T03:25:46.804Z  daemon_node_a.log_daemon  a  ERROR \177 ta
    \177 ID=TA_DISCONNECT \177  \177  \177 MCCS disconnected from TA
    \177 [2011-07-28T03:25:46.054Z] \177

2011-07-28T03:35:53.274Z  daemon_node_a.log_daemon  a  INFO \177 coord
    \177 ID=coord00 \177  \177  \177 Coordinate Transforms Command
    Handler is starting up \177 [2011-07-28T03:35:52.986Z] \177

2011-07-28T03:58:53.280Z  daemon_node_a.log_daemon  !  ERROR \177 ta
    \177 ID=TA_DISCONNECT \177  \177  \177 MCCS disconnected from DAS
    \177 [2011-07-28T03:58:52.912Z] \177

2011-07-28T03:59:45.998Z  daemon_node_a.log_daemon  a  INFO \177 ta
    \177 ID=TA_DISCONNECT \177  \177  \177 confirmed by Nancy McKown
    \177 [2011-07-28T03:59:43.865Z] \177
```

The timestamp at the front of each line follows the observatory standard format (ISO 8601) to allow other log files which start with times following the same format to be merged chronologically with this file. Also, each alert or alarm line is preceded by a field which identifies the host and process which originated the alert or alarm. Finally, each line is preceded by a one-letter identifier of the type of entry: 'H' for the header (first line of file), 'a' (lower case) for alert lines, and '!' (exclamation point) for alarm lines.

Any line-feeds, carriage-returns, and non-printing characters in the alert or alarm text are replaced by '\nnn', where nnn is the octal value of the character, with leading zeroes included.

3.3.5 SI Data-Definition XML Files

The MCCS requires certain data from every Science Instrument (SI) in order to determine and publish the SI reference frame (SIRF) information for use in transformation to/from other reference frames, for creating boresight reticles and Field Of View (FOV) indicators on imager overlays, and for monitoring observatory operation. This section describes an interface that allows the SIs to provide that data to the MCCS.

Before an observing flight, the SI team must work with their assigned mission director (MD), telescope operator (TO), and instrument scientist to create an SI Data XML file. The file has two sections – a predefined section which should *not* be modified by the SI team, and a user-modifiable section for instrument-specific

data definitions. In the latter section, the file must define at least one instrument configuration, referred to as a "mode" in the XML file. Note that the mode parameter in this file is independent from similar parameters used in other SOFIA contexts (e.g. the various modes defined and used by DCS).

Each mode defined in the XML file must contain the following minimal set of information:

1) The SI rotation angle (rotation_si) is about the W-axis of TARF (using the right-hand rule) and gives the orientation of the SI X and Y axes. The rotation_si is an angle in degrees. Figure 3 defines a Cartesian coordinate system relative to the center of the SI array to be used by all the SI teams to provide the mounting orientation information to the MCCS. Because Figure 3 is drawn from a perspective on the sky, the U and V axes of TARF are shown where they would be projected on SIRF. With the TA parked at 40 degrees elevation, the X-axis of SIRF ideally would point out the left wing, and Y points directly down. Due to a discrepancy in MCCS (SOF-2182: MCCS uses incorrect sign when rotating 40 degrees from SIRF to TARF), 80 degrees must be subtracted from rotation_si to achieve this ideal orientation.[1] This (X, Y, Z) definition provides a right-handed system that remains fixed relative to the mounted SI as it rotates with rotations in TA elevation (U), cross-elevation (V) or LOS (W) axes. The actual TA elevation rotation axis is about the TA U-axis down the center of the NM tube.
2) TA focus position (ta_focus_position_si) for the SI being mounted. This is the physical location (in millimeters) on the Z-axis of the SIRF where the TA focus is located for the SI. A positive value of 300 would be at the SI mounting flange and greater than 300 would be located within the SI hardware. This physical sense of Z is in contrast to sense of Z on the sky.
3) The pixel array scale factors in both SI axes (x_scale_si and y_scale_si). This is the plate-scale of the detector array, which may not be the same as for the TA at the above focus position (because of re-imaging optics within the SI). x_scale_si and y_scale_si are in arcsec/mm. (See the note in #1 above concerning what would result from setting incorrect signs for these values.)
4) The separation between array pixels in the SI reference frame (x_sep_si, y_sep_si). x_sep_si and y_sep_si are in microns and are measured from the center of an unbinned pixel to the center of the adjacent unbinned pixel (horizontally or vertically, respectively).

[1] Note that rotation_si values of -80 or 100 both address the MCCS error (80 + (-80) = 0, or 80 + 100 = 180) such that the SIRF X axis aligns with TARF U).

5) Pixel ranges in both SI axes (x_pixel_min_si, x_pixel_max_si, y_pixel_min_si and y_pixel_max_si), and pixel ID (i.e., x_pixel_boresight_si, y_pixel_boresight_si) of SI boresight and pixel ID (i.e., x_pixel_origin_ta, y_pixel_origin_ta) of the telescope's optical axis. This is given so the location of the SI boresight with respect to the nominal TA optical axis is known and an outline of the detector array of an SI can be overlaid on the FOV of imagers.

6) Lastly a Boolean indicator (fpi) as to whether or not the Focal Plane Imager (FPI) is in the light path due to the use of a dichroic or a fully reflective tertiary mirror.

Users are allowed to define instrument-specific housekeeping data in the SI Data XML file. Once specified, this data may be set as desired throughout flights and thereby made available to other users and saved in MCCS Archives as housekeeping data.

The data definition shall be provided as an XML file with the indicated names for each of the listed standard data items. The XML data item names are required to be all lowercase. XML tags and attributes are also case sensitive (e.g., 'Value' and 'DataNode' must be mixed case as shown here). The data definition should also provide the units and limits for the data where appropriate.

Note that while units are specified above and in the XML sample file, these may be set by the SI team to whichever MCCS-understandable units are convenient in the SI team's data-definition XML file. (See Table 13, Common MCCS Units and Abbreviations, in Section 4.1.1 for common units understood by MCCS.) The MCCS reads the file and converts as needed to assure proper interpretation of the values.

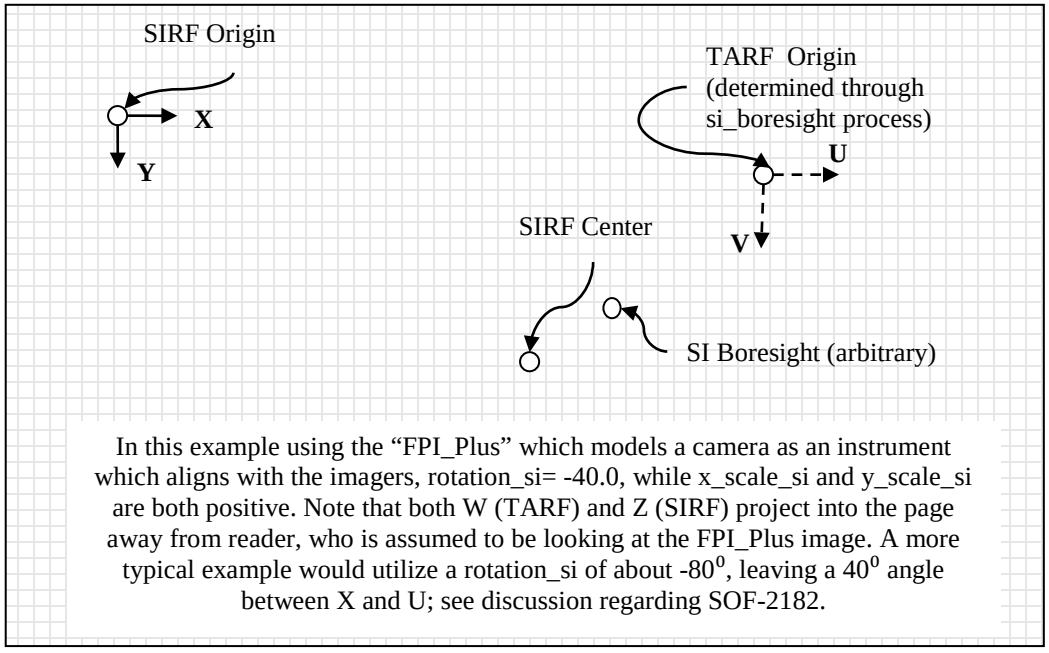

Figure 3. (X, Y, Z) SI Reference Frame (SIRF)

Here is an example of a partial XML file for the High Speed Imaging Photometer for Occultations (HIPO) SI with the first (non-editable) section replaced by an ellipsis other than the header (Please note that distributor and dataGroup settings should **not** be edited):

```
<?xml version="1.0" standalone="no"?>
<!DOCTYPE DataNode SYSTEM "datanodes.dtd">

<DataNode name="hipo" label="HIPO Science Instrument"
        rcs_id="$Id: hipo_data.xml,v 1.7 2002/010/24 23:27:07 jrobbins test $"
    description="Data provided by HIPO"
    distributor="true" dataGroup="false">

    … (non-editable stuff) …

<!--
 ==  EDITABLE SECTION STARTS HERE
 ==
 ==  The DataNode for each science instrument mode must contain the same list of value elements,
 ==  named rotation_si, ta_focus_position_si, x_scale_si, y_scale_si, x_sep_si, y_sep_si,
 ==  x_pixel_min_si, y_pixel_min_si, x_pixel_max_si, y_pixel_max_si, x_pixel_boresight_si,
 ==  y_pixel_boresight_si, x_pixel_origin_ta, y_pixel_origin_ta, and fpi.  The names
 ==  are required but the values of the initial, hilim, lolim, and other settings will vary from mode
 ==  to mode and instrument to instrument.
 ==
 ==  In many cases, the x and y values will be identical, but since there may be instruments (or
 ==  even modes within an instrument) for which they are not, both must be supplied.  Similarly, in
 ==  many cases, a particular value will be the same in all modes of operation, but since there may
 ==  be an instrument in which the value differs by mode, we require the value in each mode.
 -->

<DataNode name="si_config" label="SI configuration"
    description="SI configuration for the current observing mode"
    dataGroup="false">

<!--
 ==  The name within each FieldValue element must match the name of the DataNode (below) and
 ==  will be used as an allowed value for the hipo.mode current=… command.
 -->

    <Value name=”current_mode” rep=”STRING” initial=”hipo_47_red”
        dataGroup=”true” label=”Current mode” description=”Current configuration mode for the SI”>
            <FieldValue
                name=”hipo_47_red ”    value=”1” >
            </FieldValue>
            <FieldValue
                name=”hipo_mode_2”    value=”2” >
            </FieldValue>
            <FieldValue
                name=”hipo_mode_3”    value=”3” >
            </FieldValue>
            *
            *
            *
            <FieldValue
                name=”hipo_mode_n”    value=”n” >
            </FieldValue>

    <Value name="mcstime" rep="FLOAT8" units="seconds" label="MCS Timestamp"
        format="%.3ymdz" attribute="yes"
        description="Time the data arrived at MCS (seconds since 1970)"
```

VERIFY THAT THIS IS THE CORRECT VERSION BEFORE USE

```
        />

    </Value>

<!--
 ==  The analog_chops string is a list of additional chop names appropriate to the instrument
 ==  (“plus”, “minus”, “zero”, and “no_chop” are supplied by default).
 -->

    <Value name=“analog_chops”
            rep=            “STRING”
            label=          “ANALOG CHOPS”
            initial=        “north west south east”
            format=         “%s”
            dataGroup=      “true”
            description=    “Additional chop positions for the HIPO instrument”>

    <Value name=”mcstime” rep=”FLOAT8” units=”seconds” label=”MCS Timestamp”
        format=”%.3ymdz” attribute=”yes”
        description=”Time the data arrived at MCS (seconds since 1970)”
        />
    </Value>

</DataNode>

<!--
 ==  Description of the hipo_47_red mode.
 -->

<DataNode name=" hipo_47_red " label="SI Data"
    description="SI mounting and scaling data"
    dataGroup="false">

<!--
 ==  Angle between SI axes (x,y) and Mounting Flange axes (v,w), typically 50 degrees.
 -->

    <Value name=”rotation_si” rep=”FLOAT8” initial=”100.52” units=”degrees”
        lolim=”-180.0” hilim=”180.0”
        label=”SI rotation” description=”SI rotation relative to the mounting flange”
        dataGroup=”true” format=”%lf”>

    <Value name=”mcstime” rep=”FLOAT8” units=”seconds” label=”MCS Timestamp”
        format=”%.3ymdz” attribute=”yes”
        description=”Time the data arrived at MCS (seconds since 1970)”
        />
    </Value>

<!--
 ==  Backfocus for the instrument in this mode (mm forward of flange).
 -->

    <Value name=”ta_focus_position_si” rep=”FLOAT8” initial=”319.0”
        units=”millimeters” hilim=”900.0”    lolim=”-300.0” hiwarn=”500.0” lowarn=”-100.0”
        label=” TA focus position” description=” TA focus position on the z-axis of SIMFRF
        for a given SI” dataGroup=”true” format=”%lf”>

    <Value name=”mcstime” rep=”FLOAT8” units=”seconds” label=”MCS Timestamp”
        format=”%.3ymdz” attribute=”yes”
        description=”Time the data arrived at MCS (seconds since 1970)”
        />
    </Value>
```

```
<!--
 ==  Horizontal plate scale for the array and canonical backfocus used in this mode.
 -->

    <Value name=" x_scale_si" rep="FLOAT8" initial="25.108" units="arcsec/mm"
        label="SI X pixel scale" description="SI x-axis pixel scale factor" dataGroup="true" format="%lf">

    <Value name="mcstime" rep="FLOAT8" units="seconds" label="MCS Timestamp"
        format="%.3ymdz" attribute="yes"
        description="Time the data arrived at MCS (seconds since 1970)"
        />
    </Value>

<!--
 ==  Vertical plate scale for the array and canonical backfocus used in this mode.
 -->

    <Value name=" y_scale_si" rep="FLOAT8" initial="25.108" units="arcsec/mm"
        label="SI Y pixel scale" description="SI y-axis pixel scale factor" dataGroup="true" format="%lf">

    <Value name="mcstime" rep="FLOAT8" units="seconds" label="MCS Timestamp"
        format="%.3ymdz" attribute="yes"
        description="Time the data arrived at MCS (seconds since 1970)"
        />
    </Value>

<!--
 ==  Distance between centers of pixels adjacent horizontally.
 -->

    <Value name="x_sep_si" rep="FLOAT8" initial="13.0" units="micrometers"
        label="SI X pixel separation" description="SI x-axis pixel separation(center to center)"
        dataGroup="true" format="%lf">
    <Value name="mcstime" rep="FLOAT8" units="seconds" label="MCS Timestamp"
        format="%.3ymdz" attribute="yes"
        description="Time the data arrived at MCS (seconds since 1970)"
        />
    </Value>

<!--
 ==  Distance between centers of pixels adjacent vertically.
 -->

    <Value name="y_sep_si" rep="FLOAT8" initial="13.0" units="micrometers"
        label="SI Y pixel separation" description="SI y-axis pixel separation(center to center)"
        dataGroup="true" format="%lf">
    <Value name="mcstime" rep="FLOAT8" units="seconds" label="MCS Timestamp"
        format="%.3ymdz" attribute="yes"
        description="Time the data arrived at MCS (seconds since 1970)"
        />
    </Value>

<!--
 ==  Position of the first column in the array.
 -->

    <Value name="x_pixel_min_si" rep="INT4" initial="-8"
        label="Min SI X pixel" description="SI x-axis minimum pixel number" dataGroup="true"
    format="%d">

    <Value name="mcstime" rep="FLOAT8" units="seconds" label="MCS Timestamp"
```

```
            format="%.3ymdz" attribute="yes"
            description="Time the data arrived at MCS (seconds since 1970)"
            />
        </Value>

<!--
 ==  Position of the last column in the array (there are x_pixel_max_si - x_pixel_min_si   + 1
 ==  columns).
 -->

        <Value name="x_pixel_max_si" rep="INT4" initial="1056"
            label="Max SI X pixel " description="SI x-axis maximum pixel number " dataGroup="true"
        format="%d">

        <Value name="mcstime" rep="FLOAT8" units="seconds" label="MCS Timestamp"
            format="%.3ymdz" attribute="yes"
            description="Time the data arrived at MCS (seconds since 1970)"
            />
        </Value>

<!--
 ==  Initial x pixel coordinate of the SI boresight.
 -->

        <Value name=" x_pixel_boresight_si" rep="FLOAT8" initial="550.7"
            label=" X location of SI boresight" description=" Column of array in which lies the SI boresight"
            dataGroup="true" format="%lf">

        <Value name="mcstime" rep="FLOAT8" units="seconds" label="MCS Timestamp"
            format="%.3ymdz" attribute="yes"
            description="Time the data arrived at MCS (seconds since 1970)"
            />
        </Value>

<!--
 ==  x pixel coordinate of the TA boresight (optical axis).
 -->

        <Value name=" x_pixel_origin_ta" rep="FLOAT8" initial="552.76"
            label=" X location of TA optical axis" description=" Column of array in which lies the TA optical
            axes" dataGroup="true" format="%lf">

        <Value name="mcstime" rep="FLOAT8" units="seconds" label="MCS Timestamp"
            format="%.3ymdz" attribute="yes"
            description="Time the data arrived at MCS (seconds since 1970)"
            />
        </Value>

<!--
 ==  Position of the first row in the array.
 -->

        <Value name="y_pixel_min_si" rep="INT4" initial="-1"
            label="Min SI y pixel" description="SI y-axis minimum pixel number" dataGroup="true"
        format="%d">

        <Value name="mcstime" rep="FLOAT8" units="seconds" label="MCS Timestamp"
            format="%.3ymdz" attribute="yes"
            description="Time the data arrived at MCS (seconds since 1970)"
            />
        </Value>
```

```
<!--
 ==  Position of the last row in the array (there are y_pixel_max_si - y_pixel_min_si   + 1 rows).
 -->

    <Value name=”y_pixel_max_si” rep=”INT4” initial=”1027”
        label=”Max SI Y pixel ” description=”SI y-axis maximum pixel number ” dataGroup=”true”
    format=”%d”>

    <Value name=”mcstime” rep=”FLOAT8” units=”seconds” label=”MCS Timestamp”
        format=”%.3ymdz” attribute=”yes”
        description=”Time the data arrived at MCS (seconds since 1970)”
        />
    </Value>

<!--
 ==  Initial y pixel coordinate of the SI boresight.
 -->

    <Value name=” y_pixel_boresight_si” rep=”FLOAT8” initial=”515.9”
        label=” Y location of SI boresight” description=”Row of array in which lies the SI boresight”
        dataGroup=”true” format=”%lf”>

    <Value name=”mcstime” rep=”FLOAT8” units=”seconds” label=”MCS Timestamp”
        format=”%.3ymdz” attribute=”yes”
        description=”Time the data arrived at MCS (seconds since 1970)”
        />
    </Value>

<!--
 ==  y pixel coordinate of the TA boresight (optical axis).
 -->

    <Value name=” y_pixel_origin_ta” rep=”FLOAT8” initial=”95”
        label=” Y location of TA optical axis” description=”Row of array in which lies the TA optical axes”
        dataGroup=”true” format=”%lf”>

    <Value name=”mcstime” rep=”FLOAT8” units=”seconds” label=”MCS Timestamp”
        format=”%.3ymdz” attribute=”yes”
        description=”Time the data arrived at MCS (seconds since 1970)”
        />
    </Value>

<!--
 ==  Set fpi = “true”, if a dichroic tertiary mirror is used and the FPI may be used to determine
 ==  chop calibration, etc.
 ==  Set fpi = “false”, if the FPI may not be used in this mode.
 -->

    <Value name=” fpi” rep=”BOOL4” initial=”TRUE”  form=”ENUM”
        label=”Focal Plane Imager” description=”Focal Plane Imager present in light path”
        dataGroup=”true” format=”%s”>

        <FieldValue name=”false”  value=”0” />
        <FieldValue name=”true”   value=”1” />

    <Value name=”mcstime” rep=”FLOAT8” units=”seconds” label=”MCS Timestamp”
        format=”%.3ymdz” attribute=”yes”
        description=”Time the data arrived at MCS (seconds since 1970)”
        />
    </Value>

</DataNode>
```

```
<!--
 ==  Description of the second mode for the instrument (here called “hipo_mode_2” for lack of a
 ==  better name).
 -->

<DataNode name="hipo_mode_2" label="SI Data"
      description="SI mounting and scaling data"
      dataGroup="false">

      <Value name=”rotation_si” rep=”FLOAT8” initial=”3.1” units=”degrees”
            lolim=”-180.0” hilim=”180.0”
            label=”SI rotation” description=”SI rotation relative to the mounting flange” dataGroup=”true”
            format=”%lf”>

      <Value name=”mcstime” rep=”FLOAT8” units=”seconds” label=”MCS Timestamp”
            format=”%.3ymdz” attribute=”yes”
            description=”Time the data arrived at MCS (seconds since 1970)”
            />
      </Value>

      <Value name=”ta_focus_position_si” rep=”FLOAT8” initial=”.3”
            units=”millimeters” hilim=”900.0” lolim=”-300.0” hiwarn=”150.0” lowarn=”-125.0”
            label=” TA focus position” description=” TA focus position on the z-axis of SIMFRF for a given SI”
            dataGroup=”true” format=”%lf”>

      <Value name=”mcstime” rep=”FLOAT8” units=”seconds” label=”MCS Timestamp”
            format=”%.3ymdz” attribute=”yes”
            description=”Time the data arrived at MCS (seconds since 1970)”
            />
      </Value>

      <Value name=” x_scale_si” rep=”FLOAT8” initial=”10.0” units=”arcsec/mm”
            label=”SI X pixel scale” description=”SI x-axis pixel scale factor” dataGroup=”true” format=”%lf”>

      <Value name=”mcstime” rep=”FLOAT8” units=”seconds” label=”MCS Timestamp”
            format=”%.3ymdz” attribute=”yes”
            description=”Time the data arrived at MCS (seconds since 1970)”
            />
      </Value>

      <Value name=” y_scale_si” rep=”FLOAT8” initial=”10.0” units=”arcsec/mm”
            label=”SI Y pixel scale” description=”SI y-axis pixel scale factor” dataGroup=”true” format=”%lf”>

      <Value name=”mcstime” rep=”FLOAT8” units=”seconds” label=”MCS Timestamp”
            format=”%.3ymdz” attribute=”yes”
            description=”Time the data arrived at MCS (seconds since 1970)”
            />
      </Value>

      <Value name=”x_sep_si” rep=”FLOAT8” initial=”1.0” units=”micrometers”
            label=”SI X pixel separation” description=”SI x-axis pixel separation” dataGroup=”true”
      format=”%lf”>

      <Value name=”mcstime” rep=”FLOAT8” units=”seconds” label=”MCS Timestamp”
            format=”%.3ymdz” attribute=”yes”
            description=”Time the data arrived at MCS (seconds since 1970)”
            />
      </Value>

      <Value name=”y_sep_si” rep=”FLOAT8” initial=”1.0” units=”micrometers”
```

```
      label="SI Y pixel separation" description="SI y-axis pixel separation" dataGroup="true"
format="%lf">

<Value name="mcstime" rep="FLOAT8" units="seconds" label="MCS Timestamp"
      format="%.3ymdz" attribute="yes"
      description="Time the data arrived at MCS (seconds since 1970)"
      />
</Value>

<Value name="x_pixel_min_si" rep="INT4" initial="-31"
      label="Min SI X pixel" description="SI x-axis minimum pixel number" dataGroup="true"
format="%d">

<Value name="mcstime" rep="FLOAT8" units="seconds" label="MCS Timestamp"
      format="%.3ymdz" attribute="yes"
      description="Time the data arrived at MCS (seconds since 1970)"
      />
</Value>

<Value name="x_pixel_max_si" rep="INT4" initial="32"
      label="Max SI X pixel " description="SI x-axis maximum pixel number " dataGroup="true"
format="%d">

<Value name="mcstime" rep="FLOAT8" units="seconds" label="MCS Timestamp"
      format="%.3ymdz" attribute="yes"
      description="Time the data arrived at MCS (seconds since 1970)"
      />
</Value>

<Value name=" x_pixel_boresight_si" rep="FLOAT8" initial="0"
      label=" X location of SI boresight" description=" Column of array in which lies the SI boresight"
      dataGroup="true" format="%lf">

<Value name="mcstime" rep="FLOAT8" units="seconds" label="MCS Timestamp"
      format="%.3ymdz" attribute="yes"
      description="Time the data arrived at MCS (seconds since 1970)"
      />
</Value>

<Value name=" x_pixel_origin_ta" rep="FLOAT8" initial="0.5"
      label=" X location of TA optical axis" description=" Column of array in which lies the TA optical
      axes" dataGroup="true" format="%lf">

<Value name="mcstime" rep="FLOAT8" units="seconds" label="MCS Timestamp"
      format="%.3ymdz" attribute="yes"
      description="Time the data arrived at MCS (seconds since 1970)"
      />
</Value>

<Value name="y_pixel_min_si" rep="INT4" initial="-15"
      label="Min SI y pixel" description="SI y-axis minimum pixel number" dataGroup="true"
format="%d">

<Value name="mcstime" rep="FLOAT8" units="seconds" label="MCS Timestamp"
      format="%.3ymdz" attribute="yes"
      description="Time the data arrived at MCS (seconds since 1970)"
      />
</Value>

<Value name="y_pixel_max_si" rep="INT4" initial="16"
      label="Max SI Y pixel" description="SI y-axis maximum pixel number" dataGroup="true"
format="%d">
```

```
        <Value name=”mcstime” rep=”FLOAT8” units=”seconds” label=”MCS Timestamp”
            format=”%.3ymdz” attribute=”yes”
            description=”Time the data arrived at MCS (seconds since 1970)”
            />
        </Value>

        <Value name=” y_pixel_boresight_si” rep=”FLOAT8” initial=”0”
            label=” Y location of SI boresight” description=” Row of array in which lies the SI boresight”
            dataGroup=”true” format=”%lf”>

        <Value name=”mcstime” rep=”FLOAT8” units=”seconds” label=”MCS Timestamp”
            format=”%.3ymdz” attribute=”yes”
            description=”Time the data arrived at MCS (seconds since 1970)”
            />
        </Value>

        <Value name=” y_pixel_origin_ta” rep=”FLOAT8” initial=”0.5”
            label=” Y location of TA optical axis” description=” Row of array in which lies the TA optical axes”
            dataGroup=”true” format=”%lf”>

        <Value name=”mcstime” rep=”FLOAT8” units=”seconds” label=”MCS Timestamp”
            format=”%.3ymdz” attribute=”yes”
            description=”Time the data arrived at MCS (seconds since 1970)”
            />
        </Value>

        <Value name=” fpi” rep=”BOOL4” initial=”TRUE” form=”ENUM”
            label=”Focal Plane Imager” description=”Focal Plane Imager present in light path”
            dataGroup=”true” format=”%s”>

            <FieldValue name=”false”  value=”0” />
            <FieldValue name=”true”   value=”1” />

        <Value name=”mcstime” rep=”FLOAT8” units=”seconds” label=”MCS Timestamp”
            format=”%.3ymdz” attribute=”yes”
            description=”Time the data arrived at MCS (seconds since 1970)”
            />
        </Value>

    </DataNode>

    <DataNode name="hipo_mode_3" label="SI Data"
        description="SI mounting and scaling data"
        dataGroup="false">
            *
            *
            *
    </DataNode>

    <DataNode name="hipo_mode_n" label="SI Data"
        description="SI mounting and scaling data"
        dataGroup="false">
            *
            *
            *
    </DataNode>

    </DataNode>
```

As shown in the above XML example, default or initial values should be provided within the file for all data items. The DataNode element named si_config contains two Values: current_mode, holding the indicator for the current configuration which has been defaulted to hipo_47_red, and analog_chops, containing a list of additional chop names for the MCCS software to accept.

The current_mode Value is an enumerated list defined by the list given of FieldValues. That list needs to match the list of configuration mode DataNodes that are included in the si_config definition. The hipo_47_red FieldValue is the first in that list of SI configuration definitions where each definition contains the full complement of required data from the SI for one observing configuration mode. The DataNode name, hipo_47_red, is simply a label, so it should be given a name that is descriptive of the observation to be made with the configuration it defines. Only one SI mode can be the "current_mode" at a given time, since it is this mode by which the MCCS will point the TA.

Limit values (shown above for focus position) allow MCCS and its GUIs to properly display and alert MCCS operators that a dynamic data value is approaching (warn) or has reached (error) one of its limits. Also, other data beyond what the MCCS requires can also be included in the file to make it available to other MCCS operators/users. But, the “required” data listed above must maintain a fixed definition and structure for each SI configuration block so that the internal MCCS applications can locate that data in a consistent way for any SI.

Once the SI Data XML file is created, mission operations staff will update the accompanying “tags” XML file (e.g. hipo_tags.xml) as needed. This file identifies which parameters defined in the SI Data XML file may be modified using the SCL “set” command.

In order for the SI Data XML file and accompanying tags file to be used, a session must declare the science instrument to be used by issuing the command (for this example –“hipo”):

```
10  coord.instrument   name=hipo
```

Note that only a single instrument reference frame may be active at any time in MCCS. Therefore only one user (SCL session) should issue the coord.instrument command.

A simulated image of the Focal Plane Imager (FPI) with the hipo_47_red definition overlaid upon it could look like the following figure 4:

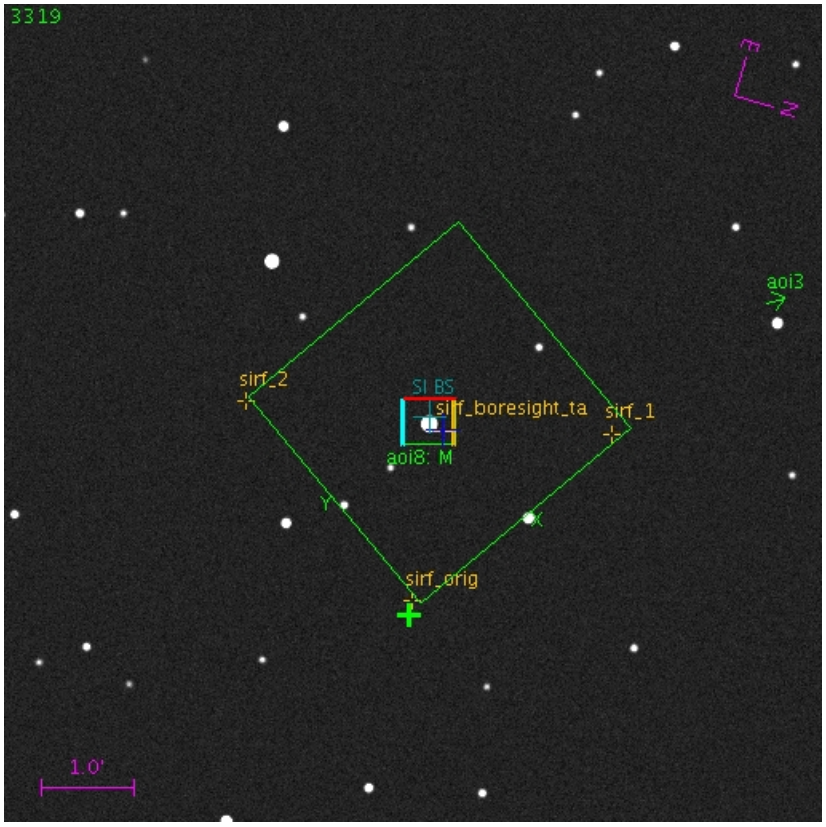

Figure 4 Simulated image with SIRF overlay

The above image models a mode with a rotation_si=100 and a negative x_scale. The value of 100 produces a 180 degree rotation (80 due to the aformentioned software model error plus the 100 specified). The equivalent image can be modeled with a rotation_si = -80 and a negative y_scale. The value of -80 produces a 0 degree rotation (80 + (-80) = 0).

Upon executing the coord.instrument command, the file called “hipo_data.xml” is used to create a new data block within the MCCS data handling (cache) system. There may be more than one SI data block in the data handling system. These data can then be accessed by their name just as any other MCCS Housekeeping (HK) data using SCL commands “subscribe” and “get” (see Section 4.2.1).

This command also gives a name, “hipo” to the “*instrument*” command handler and assigns to this command handler the authority to alter the parameters that have been stored in the SI data block named “hipo”. The complete set of MCCS commands available to any science instrument are listed in Appendix A in the section called ‘rien’. In actual use, each command would be prefaced by the instrument name (such as ‘hipo’) instead of ‘rien’, which is a fictitious SI.

The user can now select a particular hipo mode/configuration by setting the current instrument mode to the name of the desired mode given in the SI Data XML file (say “hipo_47_red”) by entering:

```
11  hipo.mode current=hipo_47_red
```

Other individual data items within a configuration block can be "set" by the SI to allow "tweaking" of the pre-defined configuration values during observation setup. For example, updating the TA focus position to half a millimeter on the z-axis of the SIMFRF (as defined in Appendix A) for the current configuration mode:

```
16 hipo.set hipo_47_blue.ta_focus_position_si=0.5
```

This only sets the data in HK, not in the XML file itself. *To generate a new XML file containing the new setting, use the "save" command, e.g.,*

```
99 hipo.save
```

This new XML file is given the file name "hipo_data_<timestamp>.xml", where <timestamp> follows the convention given in Table 9, ASCII Value Formats (e.g. 'hipo_data_20110728T025853Z.xml').

A "set" command will need to be issued each time users need to change any dynamic data value. MCCS users may automatically get notification of the value change through the use of the "subscribe" command.

If the investigator wishes to "reset" the data configuration to the default (initial) values from the original XML definition file they can issue the "restore" command at their MCCS session:

```
7 hipo.restore file=hipo_data.xml
```

This restores all the configurations from the hipo_data.xml file. The following command will restore just the configurations for modes 2 and 3:

```
7 hipo.restore file=hipo_data.xml list=[hipo_mode_2 hipo_mode_3]
```

The HK data will be restored to the values indicated in the XML for those modes listed.

3.3.6 Observatory Status and Metric Data

In order to monitor observatory status and performance metrics, several housekeeping values must be set by observatory staff and SI team computers. This section defines these parameters and identifies who is responsible for setting each.

The observatory status and metric data housekeeping values shall be set by the SI to record when the SI is acquiring photons from the sky (integrating) and thereby permit post-flight computation of the Data Collection Time (DCT) observatory performance metric. A secondary purpose of setting these values is to allow onboard staff and users to know when the SI is integrating.

SI Integrating Flag (HK name: coord.si.integ):
Set by the SI computer to indicate the start and end of each period when the SI is acquiring photons from the sky (integrating). It is an integer (INT4) that also tells which part of the SI is integrating (if there are multiple integration sources).

= 10 * source# + 1 (if starting an integ.) OR + 0 (if ending an integ.)

Thus a value of 1 means source 0 is starting an integration, and 0 means source 0 is ending an integration. Similarly, a value of 21 means source 2 is starting an integration, and 20 means it is ending an integration.

An SI with a single source of integrations may ignore the source number and set this value to 1/0 at the start/end of each integration. An SI with multiple integration sources must assign each source a unique sequential number (0 for first source, 1 for second, etc.) and use it to generate the value that indicates the start and end of each integration it initiates.

SI Integration Duration (HK name: coord.si.exptime):
Set by the SI computer following completion of each period of acquiring photons from the sky to the duration of the integration in seconds. This value is floating point (FLOAT8). Note that this may be the same value which the SI computer writes to the FITS header keyword EXPTIME in science data files (defined in DCS-SI ICD, SCI-US-ICD-SE03-2023).

In order to keep this quantity comparable to other observatory metrics (e.g. Science Flight Hours (SFH) and Research Hours (RH)) and its measurement as consistent as possible between different instruments coord.si.exptime *should not* include time spent moving the telescope between positions (such as nod beams or dither positions) or the settling time after a move, but *should* include time during which the SMA is moving between chop positions. The computation of DCT for a flight just adds together all the time periods during which the SI reported that one or more integration sources in the instrument were integrating. Essentially, SFH tells the number of hours available to the SI to directly control the telescope and acquire science data, and DCT tells how many of the SFH hours were spent acquiring photons from the sky.

SI Integration End Time (HK name: coord.si.utcend):
Set by the SI computer following completion of each integration to the UTC end time of the integration in seconds since 1970 (i.e. Unix time). This value is floating point (FLOAT8). This may correspond to the same time value the SI computer writes to the FITS header keyword UTCEND in science data files (defined in DCS-SI ICD, SCI-US-ICD-SE03-2023).

SI Integration Start Time (HK name: coord.si.utcstart):
An optional value, set by SI computer following completion of an integration to the UTC start time of the integration in seconds since 1970 (i.e. Unix time). This value is floating point (FLOAT8). This may correspond to the same time value the SI computer writes to the FITS header keyword UTCSTART in science data files (defined in DCS-SI ICD, SCI-US-ICD-SE03-2023).

If coord.si.utcstart is not set by the SI, the integration is assumed to have occurred continuously from coord.si.exptime seconds before coord.si.utcend to the time coord.si.utcend. While if coord.si.utcstart is provided, it is used as the integration start time and a duty cycle is computed for that integration: duty_cycle = si.exptime / (si.utcend – si.utcstart). That duty cycle is used to scale contributions of that integration to accumulated DCT. Thus, providing this value can improve the accuracy of the computed DCT for instruments that do not continuously integrate during an integration period.

For example, immediately before SI integration source 2 (i.e. the third source in the current SI) starts an integration, an SI computer issues the SCL command:

VERIFY THAT THIS IS THE CORRECT VERSION BEFORE USE

```
10 coord.set  si.integ=21
```

And immediately upon completing that integration:

```
12 coord.set si.integ=20 si.exptime=8.155
    si.utcend=1382692072.422 si.utcstart=1382692063.970
```

In order to keep the association clear between exposure duration, start time, end time, and which integration is being ended, these parameters should be set in a single SCL command as shown above.

An aborted integration is indicated like completing a normal integration, but with 0 for si.exptime:

```
12 coord.set si.integ=20 si.exptime=0 si.utcend=1382692068.805
    si.utcstart=1382692063.970
```

In this case, the si.utcend and si.utcstart values do not affect the DCT since there is no exposure time to accumulate; however they are still informative about the actual duration of the aborted integration.

Note that it is mandatory for SI teams to set the above values since these are used to determine the Data Collection Time (DCT) observatory metric.

In addition to the above, three values are provided to identify the state of the observatory throughout the flight. The MCCS and SI software shall set the mission director, telescope operator, and science instrument operator respective status flag per the definitions below whenever the indicated status changes upon operator action.

Mission Director Status Flag (HK name: coord.obsstat_md):

Set by the MD (via workstation GUI) to provide overall observatory status and to indicate hand off of control of the telescope to the TO. This flag is used to generate the RH metric.

It is an enumerated value (INT4) with four possible values:

0 = None (default on MCCS startup)

1 = Observatory fault (when set, stops accumulation of the Research Hours metric)

2 = Turning

3 = On setup leg or science observing leg

Telescope Operator Status Flag (HK name: coord.obsstat_to):

Set by the TO (via workstation GUI) to indicate control or not

It is an enumerated value (INT4) with four possible values:

0 = None (default on MCCS startup)

1 = Telescope Assembly fault (when set, stops accumulation of the Science Flight Hours metric)

2 = Configuring

3 = Ready (grants control of telescope to SI)

Science Instrument Status Flag (HK name: coord.obsstat_si):
Set by the SI computer to provide status of observing.
It is an enumerated value (INT4) with four possible values:
0 = None (default on MCCS startup)
1 = SI fault (when set, stops accumulation of the Data Collection Time metric)
2 = Completed
3 = Observing

The following command sequence illustrates how the above status and metric values might be set during a flight leg.

Who	**Command**	**Meaning**
MD	coord.set obsstat_md=3	On science target observing leg (grant control of telescope to TO)
TO	coord.set obsstat_to=2	Configuring (acquiring target and setting up for SI control)
TO	coord.set obsstat_to=3	Ready (grant control to SI)
SI	coord.set obsstat_si=3	Observing (SI accepting control of telescope)
SI	coord.set si.integ=1	SI 0 source starting an integration
SI	coord.set si.integ=21	SI 2 source starting integration
SI	coord.set si.integ=0 si.exptime=30.0 si.utcstart=1382692015.284 si.utcend=1382692050.422	SI 0 source ending integration
SI	coord.set si.integ=11	SI 1 source starting integration
SI	coord.set si.integ=20 si.exptime=18.565 si.utcstart=1382692030.209 si.utcend=1382692055.691	SI 2 source ending integration
SI	coord.set si.integ=10 si.exptime=35.1 si.utcstart=1382692044.682 si.utcend=1382692088.051	SI 1 source ending integration
SI	…	SI data acquisition continues…
SI	coord.set obsstat_si=2	Completed (SI has completed observations and relinquishes control)
TO	coord.set obsstat_to=2	Configuring (show that TO has accepted telescope control to prepare for turn or for other purposes)
MD	coord.set obsstat_md=2	Turning (ends previous leg and preparing for next)

Note that MCCS automatically timestamps each of the above status and metric values when set to provide a record in HK data of when each change occurs.

3.4 MCCS Performance

MCCS provides the primary functional interface between user computers and the observatory and therefore must rapidly receive, process, and forward commands and data. MCCS contribution to command-to-response times are described below. Note that actual command execution time depends on what the command does and is not part of these times.

For any command of less than 80 characters, MCCS will parse, acknowledge, and forward the command to the designated subsystem for execution within 100 milliseconds of when it was received by MCCS. Similarly, MCCS will forward data (such as a command completion response) from subsystems within 100 milliseconds of receiving that data from the subsystem.

Commands longer than 80 characters are not guaranteed to respond within the times given above.

3.5 Reference Frames

The list of reference frames available to the user in housekeeping data or commands are shown in Table 10.

A number of commands have one or more arguments that allow for defining a position-related item; a summary of these commands and associated arguments is provided in Table 11. See Appendix A for the definition of command arguments.

Note also that most commands that take the name of a position or offset as an argument allow for the so-called inline definition of a new position *or offset* with a user-defined name. The coordinates which may be used in an inline position definition are listed as qualifying attributes for the position argument (e.g., pos, track, aoi, rof1, rof2) being defined. See Section 4.1.3.1 for more on inline definitions.

In addition to those in Table 10, there are a few other reference frames that are only used internally by the MCCS. The complete list is defined in the MCCS ref_frames.xml configuration file, while the details of their use and their definitions are available in *MCCS Coordinate Transform Software Architectural Design: coord.convert* (APP-AR-SPE-SW02-2012). See Section 3.3.5 for the SI data interface definition of the SI Reference Frame (SIRF) using data-definition XML files.

Table 10. User-Accessible Reference Frames

Acronym	Name
EcRF	Ecliptic Reference Frame
ERF	Equatorial Reference Frame
FFIRF	Fine Field Imager Reference Frame
FPIRF	Focal Plane Imager Reference Frame
GalRF	Galactic Reference Frame
HRF	Horizon Reference Frame

IRF	Inertial Reference Frame
SIRF	Science Instrument Reference Frame
SMARF	Secondary Mirror Assembly Reference Frame
TACRF	TA Cradle Reference Frame
TAICRF	TA Inner Cradle Reference Frame
TARF	Telescope Assembly Reference Frame
WFIRF	Wide Field Imager Reference Frame

Table 11. Command Use of Reference Frames

Command	Arguments	ERF	EcRF	GalRF	IRF	TARF	TAICRF	SIRF	FPIRF	FFIRF	WFIRF	Notes & Comments
coord.aoi_create	pos	●	●	●		●		●	●	●	●	ValueType=POSITION
coord.convert	to	●	●	●		●		●	●	●	●	ValueType=NONE (ENUM list)
coord.convert	position	●	●	●		●		●	●	●	●	ValueType=POSITION
coord.correct	position	●	●	●		●		●	●	●	●	ValueType=POSITION
coord.position	name	●	●	●		●		●	●	●	●	ValueType=POSITION
dither.define	pos	●	●	●		●		●	●	●	●	ValueType=POSITION
dither.insert	pos	●	●	●		●		●	●	●	●	ValueType=POSITION
nod.define	coord_sys	●	●	●		●		●	●	●	●	ValueType=NONE (ENUM list)
sma.chop	coord_sys	●	●	●	●	●	●	●	●	●	●	ValueType=NONE (ENUM list)
ta_pos.goto	pos reference	●	●	●		●		●	●	●	●	ValueType=POSITION
ta_pos.pattern	coord_sys	●	●	●		●		●	●	●	●	ValueType=NONE (ENUM list)
ta_pos.rewind	chop_sys nod_sys	●				●						ValueType=NONE (ENUM list)
ta_pos.tweak_define	coord_sys	●	●	●				●	●	●	●	ValueType=NONE (ENUM list)
coord.correct	*offset*	●	●	●		●		●	●	●	●	*ValueType=OFFSET*
ta_pos.goto	*offset*	●	●	●		●		●	●	●	●	*ValueType=OFFSET*
ta_pos.offset	*delta*	●	●	●		●		●	●	●	●	*ValueType=OFFSET*
ta_pos.offset	*pos*	●	●	●		●		●	●	●	●	*ValueType=POSITION*
ta_pos.rewind	*center*	●	●	●		●		●	●	●	●	*ValueType=POSITION*
ta_pos.track	*limb1* *limb2*	●	●	●		●		●	●	●	●	*ValueType=POSITION*

3.6 Observing Effects

SOFIA supports correction for a number of observing effects that affect pointing, such as non-sidereal motion and refraction. Table 12 outlines the effects and their use in position definition, publication, and/or pointing.

See Section 4.1.3.1 for examples of how MCCS correction of some effects may be invoked when the user defines a position.

Table 12. Observing Effects and their Use in Position Definition, Publication, and Pointing

Observing Effect	Position Definition		Position Publication		Pointing Use	
	Allowed	Meaning	Defined	Notes	Used	Notes
Non-sidereal motion	Yes	Associate an earth-centered ephemeris file with a position	Yes	Published sky coordinates will update per time interpolation of provided ephemeris (constant before first and after last point)	Yes	1. Ephemeris will be adjusted for current location (topocentric correction) 2. Ephemeris will be interpolated for current time to orient the TA attitude 3. A velocity will be computed and provided to the TA to adjust its planned attitude over time
Non-sidereal tracking	Yes	Associate an earth-centered ephemeris file with a tracking AOI position	Yes	AOI location reported by TA in IRF will be converted to ERF continuously as per any AOI around a sidereal object	Yes	A velocity will be computed and provided to the TA to adjust its planned AOI center over time
Precession	Yes	Between J2000 and a given mean equinox	No	Positions are published in J2000; but see coord.convert for utility to output in another mean equinox	N/A	ERF-IRF relationship and positions are both defined in J2000 with current apparent precession as observed on an imager, therefore no further precession is needed
Nutation	No		No	Positions published in J2000.	N/A	ERF-IRF relationship and positions are both defined in J2000 with current apparent nutation as observed on an imager, therefore no further nutation is needed.

Refraction	No	Not in a position; but see SI definition section for defining observing wavelength	*Yes*	*Refraction is compensated for when converting between TA coordinates and sky coordinates (i.e. subtracted when converting from TA coordinates and added when converting to TA)*	*Varies*	*1. Refraction is compensated for when pointing with positions defined in a sky reference frame (positions defined in a TA-fixed frame such as an imager would not)* *2. Refraction is compensated for when computing the astrometric offset between a guide star in one wavelength and the observing position in another if both have sky reference frames*
Annual Aberration	No		*Yes*	*Annual aberration is compensated for when converting between TA and sky coordinates (i.e. subtracted when converting from TA and added when converting to TA)*	*Yes*	*The correction due to aberration will be computed after proper motion and applied prior to computing the required IRF absolute location*
Proper Motion	Yes	RA/yr and Dec/yr can be non-zero in a position. If a non-standard equinox is provided at position definition, proper motion is applied prior to precession in order to store in J2000.	Yes	The ra_per_year and Dec_per_year values are published for inspection and applied at the current epoch at definition, to publish astrometric sky coordinates of positions in J2000	Yes	The proper motion is applied at position definition and thus applied prior to computing the required IRF absolute location
Parallax	*Yes*	*Default distance is infinite distance (no parallax)*	*Yes*	*Parallax is compensated for when converting between TA coordinates and sky coordinates (i.e. subtracted when converting from TA coordinates and added when converting to TA)*	*Varies*	*Parallax is compensated for when pointing with positions defined in a sky reference frame (positions defined in a TA-fixed frame such as an imager would not)*

4. SOFIA COMMAND LANGUAGE DESCRIPTION

This section describes how to command MCCS, subscribe to housekeeping data, monitor alerts and alarms, and watch for housekeeping limit violations.

4.1 Command and Housekeeping Data Names

MCCS commands and housekeeping data are identified by their path. Commands to 'session' (such as login, logout, subscribe, get) may omit the 'session.' prefix. For all other commands, if the command is unique across all of MCCS, it may be used alone, however if multiple MCCS entities have the same command, the full path to the command (as shown in the 'Command' column of Appendix A) is required.

For example, the following command identifiers are valid:

```
login, or session.login
get, or session.get
coord.position
nod.goto
sma.chop
```

Housekeeping data items are specified by their full path or by the shortest unique path which includes the item name. MCCS searches available parameters and returns an error if there is more than one which ends with the provided name. The full path of a housekeeping data item is constructed by concatentating the labels in the first two columns of Appendix B.4.

For example, the following are valid housekeeping data item identifers:

```
sibs.ra
sibs.dec
ta_tsc.tsc_fd_pos_abs, or tsc_fd_pos_abs
```

Note that the 'sibs.ra' and 'sibs.dec' items may not be shortened to 'ra' and 'dec' since there is more than one data item ending with the name 'ra', and more than one ending in 'dec'. However, 'tsc_fd_pos_abs' may omit the 'ta_tsc.' prefix since there is only one item with this name in MCCS.

4.1.1 Argument Lists and Modifiers

In commands and responses, some arguments may have multiple values. For example, the session 'get' and 'subscribe' commands include the LIST argument to specify which data items are requested. An example for setting 3 elements and requesting multiple data items to be returned is:

```
001 bigObject.set height=1234 width=5678 length=3
002 get List=[height width]
```

The response returns the housekeeping items 'height' and 'width' with their default units. Default units for all housekeeping data is specified in Appendix B.4. In the case of the above example, the responses would be:

```
001 :
002 : height=1234 width=5678
```

or, with showUnits turned on:

```
003 get List=[height width] units=miles
003 : height=0.2337(units=miles) width=1.0754(units=miles)
```

Note that in the above command, the units are global to the command because the modifier is not enclosed in parentheses.

Individual (i.e., not global) qualifiers may be added to each data item by appending them as attributes, within parentheses:

Generic syntax:
<keyword>=<value> (<keyword>=<value> {<keyword>=<value>})

Then, the following command sets the height in feet but the width in miles:

```
003 smallObject.set height=1234(units=feet)
    width=5678(units=miles)
004 get List=[height width (units=miles)] showUnits=yes
```

will give

```
004 : height=1234(units=feet) width=5678(units=miles)
```

Attributes may also have attributes, which could result in several levels of parenthesized nesting:

```
005 coord.position name=squat (ra=705.62 (units=″sidereal
    minutes″) dec=85d1m12.3s)
```

The example above defines a new position called “squat” using SCL inlining which is described in Section 4.1.3.3. See Appendix A to determine which arguments allow the user to adjust units – only arguments with SET in the ValueType column may have units set.

Table 13 lists common MCCS units and their abbreviations. When specifying a unit in a command, either the full or abbreviated form may be specified. File /modstor/mccs/config/units.xml defines the full list of available MCCS units.

Table 13. Common MCCS Units and Abbreviations

Full Unit Name	Abbreviation
arcminutes	arcmin
arcminutes per second	arcmin/sec
arcseconds	arcsec
arcseconds per second	arcsec/sec
arcseconds per year	arcsec/yr
astronomical units	au
degrees	deg
degrees per second	deg/sec
hertz	hz
hours	hr
julian years	yr
micrometers	microns
milliarcseconds	mas
milliarcseconds per year	mas/yr
millimeters per second	mm/sec
millimeters	mm
milliseconds	ms
minutes	min
pixels	pix
pixels per second	pix/sec
seconds	sec
sidereal hours	sid_hr
sidereal minutes	sid_min
sidereal seconds	sid_sec

4.1.1.1 Per-Command Arguments

In addition to the arguments associated with each specific command as listed in Appendix A (SCL Commands) there are also optional arguments which affect the behavior of many commands. These may be used on any SCL Command, but will only affect the command's behavior where appropriate (e.g. "verbose" setting would affect all commands, but "format" would just affect a 'get' or 'subscribe' command). The available per-command arguments are:

- attr – specifies the name of the attribute to get
  (valid values are: *all* (to retrieve all attributes), or the name of a specific attribute to be retrieved for the identified data)
  e.g. '1 get list=latitude attr=mcstime'
  returns both the value of latitude and the time when latitude was received by MCCS; the time value will be in parentheses after the latitude value
- format – specifies the format for values; format adheres to the C/C++ specification (e.g. format="%7.5f"); also includes custom MCCS formats:

- "%.3ymdz" for timestamp in standard observatory format (ISO 8601) (e.g. "2011-07-28T02:58:53.280Z")
- "%sg" for sexagesimal format (e.g. "12:13:14.5")
- "%sgd" for sexagesimal degrees, minutes, seconds (e.g. "–12d13m14s")
- "%sgh" for sexagesimal hours, minutes, seconds (e.g. "12h13m14.5s")

Note that MCCS attempts to apply the supplied format to each data item requested by the user (e.g. each value selected in the 'list=' keyword for the 'get' and 'subscribe' commands). The user should therefore be very careful when using this keyword to assure that all values being requested are of a type consistent with the format (e.g. '1 get list=coord.sky_los_rate format=%.4lf' is fine, but '1 get list=sma.focus.mode format=%.4lf' would give an erroneous value, since sma.focus.mode is a string and not a double precision (FLOAT8) value).

The Representation and default Format for housekeeping values are provided in Appendix B.4 and should be used to determine whether the user-selected format will work for requested values.

- fullname – display full path names on output
  (valid values are: yes, no, or alternative boolean values – see Table 9)
- resp_format – specifies response format
  (valid values are: LEGACY, BINARY, FORMATTED)
- showlabels – display labels on output
  (valid values are: yes, no, or alternative boolean values – see Table 9)
- showunits – display units on output
  (valid values are: yes, no, or alternative boolean values – see Table 9)
- timeout – specifies timeout for a given command in seconds
  (e.g. timeout=1.5)
- units – specifies the units for values
  (any unit defined in units.xml file; see Table 13 for commonly used units)
- verbose – turn on/off command output verbosity
  (valid values are: none, normal, all; see description in Section 3.3.1.2)

4.1.1.2 Pre-Defined Entries

For easy reference in commands, SCL provides a number of pre-defined entries for positions and reference frames that can be used as keyword values; these are listed in Table 14 below. Note that these pre-defined entries may be used for all command arguments where the 'ValueType' shown in Appendix A is 'POSITION'. Associated housekeeping values for these are listed under the Prefix 'coord.pos.ANY' in Appendix B.4.

Table 14. Pre-Defined Entries Supported in SCL

Entity Name	Entity Description
General Positions	
actual	The current position seen by the designated boresight.
desired	The position of the designated boresight that will result from the execution of the current command.
cmded	Instantaneous position along the trajectory of the telescope generated to move from 'actual' to 'desired'.
chop_ref	Chop position used for transforms internal accounting of beams and orientation of chop in various algorithms (e.g. nod, dither)
reference	A generic user "reference" position saved for future use.
temp	Used for temporary position that will not be saved for future use.
Boresight Positions	
tabs	TA boresight (this is the same as the origin of TARF – see Section 3.5, Reference Frames)
sibs	SI boresight, as specified by the SI team or user in the <*instrument*>*_data.xml* file or as modified through <*instrument*>.sibs_change command.
wfi_fov	Position of the center of the WFI FOV.
ffi_fov	Position of the center of the FFI FOV.
fpi_fov	Position of the center of the FPI FOV.
AOI/Track-Point Positions	
trc_aoi1	Position defined as AOI #1.
trc_aoi2	Position defined as AOI #2.
trc_aoi3	Position defined as AOI #3.
trc_aoi4	Position defined as AOI #4.
trc_aoi5	Position defined as AOI #5.
trc_aoi6	Position defined as AOI #6.
trc_aoi7	Position defined as AOI #7.
trc_aoi8	Position defined as AOI #8.
trk_pos	Tracking position (valid while tracking). The tracking position is not the Area Of Interest where the tracking star is located; rather it is the position (by default the location of sky at which the SI boresight, sibs, is pointed) that will be kept fixed as a result of the tracking action at the AOI.
Nod Positions	
a	Position for nod-beam A.
b	Position for nod-beam B.
x	Position for nod-beam X.
Pattern Positions	
user_pat_pt_1	Pattern position #1.
user_pat_pt_2	Pattern position #2.
Reference Frames	
All reference frame acronyms	The reference frames acronyms shown in column #1 of Table 10

4.1.1.3 Keyword Mutual-Exclusivity

Sometimes within the command keyword set, there are mutually exclusive keywords – keywords that may not be sent with each other because they represent conflicting ways to input the same information. Error responses will be returned if mutually exclusive keywords are specified for the same command. For example, the user may not specify a position's "x-coordinate" in the equatorial reference frame and its "y-coordinate" in the galactic reference frame. The following command will return an error:

```
1 coord.position name=star1(ra=12h13m14.5s bii=–12.876)
```

since ra and bii are mutually exclusive keywords. Currently, only commands within the following command sets contain mutually exclusive command keywords:

- coord
- "instrument"; i.e., flitecam, forcast, great, hawc, hipo, etc
- nod
- session
- sma
- ta_pos
- ta_state

4.1.1.4 Default Keyword Values

SCL supports a "Global Default" structure for keyword values. Default values of keywords are the same for all SCL sessions: If a default is settable and has been changed in one SCL session, it is changed for all sessions.

There are three basic kinds of keyword defaults, "fixed", "previous" (or "same-as-last-time"), and "calculated". Note that not all keywords have defaults. Some are optional and so don't have any default value and some must be specified each time they are used.

A "fixed" default does not change throughout the flight. *The tables in Appendix A indicate which keywords have fixed defaults and what the fixed values are for those defaults.*

A "previous" default will retain its value until reset. There are two types of previous defaults: those that have an initial value set automatically within the MCCS and those that require the initial value to be set at least once by a user. In the latter case, a default value is only assigned to a keyword after it has been used in an SCL command. When that command is next issued, the keyword will maintain the "same-as-last-time" default value, unless explicitly changed. Note that this default is the same for all SCL sessions.

Previous default values are stored by MCCS as Housekeeping data in *cmndgrp.command_name.keyword_name*. Therefore, such values can be retrieved by any user with SCL "subscribe" or "get" commands.

Keywords with defaults already set to the desired value (either because they are "fixed" or because of their "previous" value), do not have to be included in a command line. For example, if the user sets up a chop and starts chopping via:

```
1 sma.chop amplitude=20 pos_ang=0 phase=10 sync_src=internal
    frequency=5
```

and then decides he really wanted a pos_ang of 2 degrees, he need only send:

```
2 sma.chop pos_ang=2
```

since most of the sma.chop command's keywords have either fixed or previous defaults.

A "calculated" default automatically gives a certain default value to a keyword or attribute depending on a calculation made by the MCCS. An example of this is the keyword "track_mode" in the command "ta_pos.goto". If tracking is defined in the position identified by the "pos" keyword, then "track_mode" is set to the appropriate value ("centroid" say) - otherwise it is set to "none" by the MCCS.

Some keywords are optional. They do not have a default value but they need not be specified to make the command valid. Whatever the optional keyword "activates" in the command is not required for the general activation of the command.

The tables in Appendix A include in the Arg_Description column "Default Type". In this column the letters "F", "P", "C", "R", and "O" will indicate the following for each keyword:

F – Fixed default
P – Previous default
C – Calculated default
R – Must be specified (no default)
O – Optional input (no default)

Following an "F" in the "Default Type" column will be the value of that fixed default in parenthesis, for example "F(off)". A "previous" default will either be represented by "P(R)" or "P(initial_value)": P(R) means that the default must be supplied initially by the user; P(initial_value) means the initial default is supplied by the MCCS, and this value is given.

SCL commands may contain mutually-exclusive keywords. In the Default Type column of the SCL tables (Appendix A), the presence of a "/O" after a "R" or "P" value; i.e., "R/O" or "P(R)/O" indicates that the keyword is "optional' in the sense that it is mutually exclusive with other keywords. These combinations mean that some subset of the keyword set are required inputs, but not all of them.

4.1.2 Aliases

An alias capability is provided in the SCL. Each token in a command (any string surrounded by white space, or other delimiting character such as a parenthesis) is checked against defined aliases in an alias table if its leftmost character is a $ sign.

If an alias definition exists, the token is replaced by the string provided in the alias table. The replacement is performed as a string substitution (no variable substitution is performed), and only one substitution pass is performed. (Aliases contained in the substituted string are not expanded.)

The alias command is used to define an alias, while the unalias command is used to eliminate an alias definition. The session environment will include some predefined aliases. *Some of these aliases are locked, to prevent the user from inadvertently reassigning them and losing ready access to the original command.* To see a list of aliases for your session, enter “get list=alias".

An alias must begin with a letter, and may include any number of alphanumeric characters and underscores after the initial letter.

Aliases may be used for values and keyword settings as well as commands. For example:

```
001 Alias temps=”get list=[temp1 temp2 temp3]”
002 alias showAll=”showUnits=yes showLabels=yes”
003 $temps
004 $temps $showAll
005 get list=TAposition($showAll)
```

Note that the keywords "temps" and “showAll” in the above example are not predefined ones. The user may choose any non-reserved word. An error will be returned if the chosen token is reserved.

4.1.2.1 Pre-defined Aliases

There are several sources for most avionics items, and the preferred source may change from flight to flight. By referencing these parameters by alias name instead of the housekeeping data item name, the user is provided with the preferred source for each data item.

The following avionics items are provided as pre-defined aliases for all sessions.

- $altitude
- $gross_weight
- $ground_speed
- $heading
- $latitude
- $longitude
- $pitch
- $pressure_altitude
- $roll
- $static_air_temp
- $track_angle
- $true_airspeed
- $wind_angle

- $wind_speed
- $static_outside_air_pressure

When an alias name is used to access housekeeping data (e.g. in a ‘get’ or ‘subscribe’ command) it is important to understand that the actual housekeeping item name will appear on responses (if ‘showLabels=True’ is used) and not the alias name. The user may use the position of the values in the response string to identify which value is associated with which requested item. Alternatively, the output of "get list=alias" may be used to create a mapping between alias and housekeeping value.

The definition of each is configurable by mission operations staff by changing the MCCS configuration file global_alias.xml prior to starting MCCS.

These aliases are locked (i.e. may not be changed by the user).

4.1.3 User-Defined Positions and Offsets

Positions are locations defined in a selected reference frame using the position-related attributes in Table 16 and Table 17. A user-provided position name must begin with a letter and is comprised of letters, numbers, and underscores (case is ignored).

User-defined positions are created using the SCL construct called “inlining” as defined in the next section. The SCL commands and arguments which permit definition of user-defined positions are listed in Table 15. Other command arguments may reference positions by name, but may not create new positions.

Table 15. Arguments which Permit Inline Position Definition

Command	Argument
coord.position	name
coord.convert	position
coord.correct	position
dither.define	pos
dither.insert	pos
<si>.si_boresight	pos
<si>.si_align	pos
ta_pos.goto	pos

The MCCS can support a total of up to 100 simultaneous user-defined position definitions distributed among all active users. *This will be increased to 200 user-defined positions in a future release.*

An offset is a relative location – a deviation from a position – in a given reference frame. The offset usage implementation details are still TBD, but it is expected that they will be implemented via offset keywords as shown in Table 16.

4.1.3.1 Inline Definitions

Creating an inline definition, or "inlining" is the method for creating named positions *or offsets* within a command that makes use of the definition. The format for an inline position definition is shown by the bold text in the following:

```
<msg id> <cmndgrp.command> <keyword1>=<entity1>
   (<att1>=<value1> <att2>=<value2> …)
```

where <att1> and <att2> are two attributes to <entity1>, which is the value required for the keyword, <keyword1>, that is in turn required by the command, <cmndgrp.command>.

Once defined, a position can be used in any future command as a value for a position-type keyword, but with parenthetical attribute definitions omitted since they have already been defined.

The following is a fairly common yet somewhat complicated example of an inline position definition:

```
25 ta_pos.goto pos=great_pos(ra=193.87125(units=degrees) dec=-
    2.970417(units=degrees) centroid=trc_aoi3  inertial=yes
    chop=no_chop) track_mode=centroid
```

In this example the command ta_pos.goto with <msg id> 25 directs the TA to move to a position called great_pos and to begin tracking there. The command has the two keywords pos and track_mode, with values great_pos and centroid, respectively. The position great_pos is an inline position definition that includes coordinates, tracking information (use inertial via AOI #3), and chopper position; its attributes are the inline keywords ra, dec, centroid, inertial, and chop, each with own inline values. Note also that ra and dec have nested unit attributes of their own.

4.1.3.2 Position and Offset Attributes

Table 16 lists position and offset attributes relevant to each reference frame.

Table 16. Position and Offset Reference Frame Attributes

Reference Frame	Position Attributes	*Offset Attributes*
ecrf	`lambda="<val1>"` `beta="<val2>"` `psi="<val3>"`	*`delta_lambda="<val1>"`* *`delta_beta="<val2>"`* *`[equinox="<val3>"]`*
erf	`ra="<val1>"` `dec="<val2>"` `[equinox="<val3>"]`	*`delta_ra="<val1>"`* *`delta_dec="<val2>"`* *`[equinox="<val3>"]`*
ffirf	`xffi="<val1>"` `yffi="<val2>"`	*`delta_xffi="<val1>"`* *`delta_yffi="<val2>"`*
fpirf	`xfpi="<val1>"` `yfpi="<val2>"`	*`delta_xfpi="<val1>"`* *`delta_yfpi="<val2>"`*

galrf	lii="<val1>" bii="<val2>" rofii="<val3>"	*delta_lii="<val1>"* *delta_bii="<val2>"* *delta_rofii="<val3>"*
sirf	xsi="<val1>" ysi="<val2>"	*delta_xsi="<val1>"* *delta_ysi="<val2>"*
tarf	el="<val1>" xel="<val2>" [los="<val3>"]	*delta_el="<val1>"* *delta_xel="<val2>"* *[delta_los="<val3>"]*
wfirf	xwfi="<val1>" ywfi="<val2>"	*delta_xwfi="<val1>"* *delta_ywfi="<val2>"*

A position definition may include additional attributes as shown in Table 17.

Table 17. Additional Position Attributes

Attribute	Description
coord_sys	Coordinate system to use for the position: IRF, TARF, HRF, ERF, WFIRF, FFIRF, FPIRF, SIRF, EcRF, GalRF
chop	Chop beam (plus, minus, no_chop)
default	This keyword may be used when creating a new position to specify a previously-defined position from which to set non-defined parameters (i.e. those not set using other keywords). It should not be used when the Position already exists and is simply being modified *(eventually this error will be detected and abort the command).*
centroid, rof1, rof2, *limb1, limb2*	AOI to use for limb tracking, centroid, tracking, or ROF tracking
equinox	Equinox of the coordinate system. Default=J2000
inertial	Boolean; track inertially or non-inertially; see TA_MCCS_F ICD *(Note that this keyword is obsolete and may be deleted; the MCCS always enables this TA flag on any TA command that requires it and the TA has not implemented separate algorithms for "inertial" tracking versus "non-inertial")*
track_mode	Mode of tracking: none (i.e. no tracking), centroid (i.e. two axis tracking with one star), or rof (i.e. two axis tracking but with maintenance of rotation of field with three stars)*, or limb tracking*
ephemeris	Filename of the ephemeris; see note below for location.
ra_year, dec_year, epoch	Annual proper motion keywords for ERF (ra_year and dec_year units are milliarcseconds per year; epoch is the reference year for ra_year and dec_year with a default of 2000.0)

ascending, epoch_perihelion, inclination, lamda_at_epoch, lambda_perihelion, mean_anomaly, ecentricitiy, perihelion, mean_daily_motion, period, semi_major	*Orbital elements*
catalog	*Name of the catalog to use*
cat_name	*Name of the entry in the catalog*
guide_star	*Adds object to list of guide stars to identify the position was not for an actual observation but was a guide star (or maybe a calibrator). The coord.list type=guide_star command would then return a list of all positions that were defined with guide_star=yes.*

For example, the following is a simple definition of a position named squat3 with the noted reference frame attributes:

```
6 coord.position name=squat3(ra=5h6m14.3s dec=85d1m12.3s)
```

Note that '`equinox`' is an optional attribute that may accompany 'erf' or 'ecrf' position attributees. If the equinox is omitted in an equatorial (ra and dec) or ecliptic (lambda and beta) position definition, it defaults to 'J2000'.

Once defined, the telescope can be moved to this position by commanding the following:

```
7 ta_pos.goto pos=squat3
```

As a further example, the following illustrates the use of proper motion attributes in defining a position.

```
coord.position name=barnard (ra=17h57m48.97s dec=4d40m5.8s
    ra_year=-798.58 dec_year=10328.12 epoch=1991.25)
```

This command shows the use of an ephemeris to define the position of the target to be centroid tracked.

```
coord.position name=squat4 (ephemeris=squat4_eph.txt
    centroid=trc_aoi3)
```

Notes regarding ephemeris file definition and use:

- Ephemeris files must be placed in directory /modstor/mission_data/ephemeris on the MCCS Archiver. See Appendix F.3 for the format of the ephemeris files supported by MCCS. The observer is encouraged to choose an ephemeris stepsize for the desired accuracy required (e.g. 5 minutes). Choosing an excessively long stepsize (e.g. hours) for a near-earth object could introduce error in the projection of apparent non-sidereal velocity that could conceptually introduce pointing errors.

- MCCS interpolates the ephemeris files using UTC which is assumed to have leap seconds. An ephemeris file generated significantly in the past could be missing a leap second, resulting in pointing and/or tracking error. The observer is encouraged to verify that the ephemeris file was generated recently.

4.1.4 Values and Escape Sequences

Values may be numeric or text. Text values may be words (a string of characters which are not delimiters) or strings. A string is entered by preceding and concluding it with a double quote mark (“ ”). If the user wishes to embed double quote marks within a string, they must be escaped (as a side note, if a STRING housekeeping data type contains embedded double quotes, the MCCS will escape the embedded double quotes (\”) so the user can distinguish the embedded double quotes from the preceding and concluding double quotes). Generally, all printing ASCII characters may be entered into the string; while nonprinting characters and special delimiting characters that the user wants to treat as regular character (such as double quote marks) may be entered using the escape sequences below.

Only three characters have special meaning within a string: a double quote mark to end the string; a backslash to begin an escape sequence within the string; and a terminator (either <LF> or <CR>). When a terminator is found within a string, MCCS will return a syntax error since the string was not terminated before the end of the command.

Escape sequences are typical of those in C++ and Java, and are shown in Table 18.

Table 18. Escape Sequences

Name	Escape Sequence	Octal Equivalent
Horizontal tab	**\t**	**\011**
Vertical tab	**\v**	**\013**
Backspace	**\b**	**\010**
New line	**\n**	**\012**
Carriage return	**\r**	**\015**
Double quote	**\”**	**\042**
Backslash	\\	**\134**
Octal byte code (3-digits)	**\ddd**	**\ddd**

4.2 Monitoring Housekeeping Data

4.2.1 Subscribing to Housekeeping Data

To subscribe to housekeeping data, a subscribe command has been provided:

<msg id> **subscribe list**=<list of value names> <optional keyword=value pairs>

This command allows the user to subscribe to any housekeeping data provided by MCCS (see Appendix B.4 for a list of available MCCS housekeeping data). The

items to subscribe to are provided after the **list** keyword. Note that if multiple items are listed, they must be enclosed in brackets ([]):

```
1 subscribe list=[das.ins_1_12hz.hybrid_lat
    das.ins_1_12hz.hybrid_lon
    das.ins_1_25hz.hybrid_true_heading]<LF>
```

The subscribe command responds with the requested data when a specified condition occurs. The user may specify the following conditions:

1) Trigger on data arrival – The **trigger** keyword can be used to specify one or more data items to trigger retrieval of the current value of data items specified by the **list** keyword. Use of '**trigger**=all' causes retrieval when any of the items specified on the **list** keyword are updated, but the notification only contains the latest value for the updated item(s) and omits the other items in the list.

   Notification when every nth sample of the trigger item(s) arrive (using the **sample** keyword). Whenever the nth sample of a trigger item arrives, the user will be notified with the latest values for all the items specified in the **list** keyword (for example, if **sample**=5 is specified, the user will be notified every fifth time an update is received for the specified trigger item(s)).

2) Trigger on time – Notification at a specified time interval using the **interval** keyword. Time interval is in seconds by default (e.g. 'interval=1.8') but units may be modified with units keyword (e.g. 'interval=0.1 (units=minutes)'. When the specified time interval expires, the user is notified with the latest values for all the items specified in the **list** keyword.

If no condition keyword is specified (trigger, interval or sample), then the first item in the list is the "trigger" (so in the example above, the user will be notified with the latest values for latitude, longitude, and heading whenever the value for latitude is updated; the user is not notified when longitude or heading is updated since they are not designated as "triggers").

When the trigger condition occurs, the user is notified with an intermediate response. For example, assuming showlabels is set to yes and showunits is set to no, the following could be the response to the subscribe command above:

```
1 I das.ins_1_12hz.hybrid_lat=34.6135523790000
    das.ins_1_12hz.hybrid_lon=-118.0754739010000
    das.ins_1_25hz.hybrid_true_heading=89.999374<LF>
```

A subscribe command will continue to run until:

1) The user cancels the command.
2) A terminating condition is met. The user may specify one of two terminating conditions:
   a. Terminate when a specified number of samples have been provided to the user (using the **n_times** keyword).

b. Terminate when a specified time duration has expired (using the **duration** keyword).

The subscribe command is also used to log housekeeping data and contains some special keywords used for subscribing to alerts and alarms (see the description of the subscribe command in Appendix A for more details about these features as well as for additional details about the features described above).

Housekeeping data may also be obtained using the get command. The get command is similar to the subscribe command except that it only gives a one-time update of the items specified in the **list** keyword. Note that unlike subscribe, get has no option to wait for data to arrive (i.e. to "trigger" on data arrival), but always provides the most-recently-acquired data at the time that the command is issued.

Some values requested in either a subscribe or get command may not be available. These will be indicated by special strings NaN (for "not a number"), NotSet (the value has not been populated yet), and NotFound (the value is not in MCCS housekeeping data), or will be presented as some default (but unlikely) value.

4.2.2 Data Timestamps

An important characteristic of observatory housekeeping data is when it was produced, also known as its timestamp. All MCCS times are Universal Time (UT) as provided by the MCCS IRIG/NTP Subsystem. The MCCS IRIG/NTP Subsystem maintains time synchronization with GPS satellites and distributes time to all onboard systems via the NTP protocol over MCCS networks and directly via IRIG-B protocol to selected locations (e.g. the TA and SI patch panel). NTP and IRIG protocols typically provide accuracy of a millisecond or better.

When MCCS acquires data it also records the time when that data was acquired. For some data, there may be multiple timestamps representing different points during the data acquisition cycle. For example, all data is stamped with the time it is received by the Platform Interface Subsystem (PIS). But data from some sources, such as avionics, may also be stamped with the time when each data item (such as pitch, roll, or heading) was acquired. Such timestamps are normally added by the subsystem which directly acquired the data before forwarding it to the PIS.

Timestamp data (as with all other housekeeping data) is accessed via SCL "get" and "subscribe" commands as described below.

The timestamp(s) associated with a data item may be obtained by setting the argument "attr" (meaning attribute). If the user wishes to see the time(s) associated with every data item retrieved with SCL "get" and "subscribe" commands, they may set the attribute globally. For example:

```
1 set attr=all
```

If the user just wishes to see just the time when the Platform Interface Subsystem acquired each data item (its "mcstime"), they could use the following command.

```
2 set attr=mcstime
```

Note that “mcstime” in the above example, could be any attribute name if only that attribute is desired by the user.

```
3 set attr=<some_other_attribute_name>
```

To reset attr so that attribute values are not provided (the default behavior if the user never sets attr), the user can set attr to ‘none’.

```
4 set attr=none
```

In each of the above 4 examples, the value of “attr” is set globally for the user’s session, and will affect all subsystem “get” and “subscribe” commands until the user changes the value of attr or exits the session.

For the first example above (“`1 set attr=all`”), if the user then issued the command:

```
 2  get list=das.ic1080_10hz.pitch
```

MCCS may respond with:

```
2 : das.ic1080_10hz.pitch=10.323(mcstime=2013-07-
    28T03:38:31.351Z pkt_timestamp=2013-07-28T03:38:31.323Z)
```

Note that all that the get commands requested was pitch, but the response included its value and the values of all its attributes.

If the user were to issue the commands:

```
1 set attr=pkt_timestamp
2 get list=das.ic1080_10hz.pitch
```

MCCS may respond with the following:

```
2 : das.ic1080_10hz.pitch=10.323(pkt_timestamp=2013-07-
    28T03:38:35.854Z)
```

Attribute values may also be obtained directly by name. For example, to directly request the above pkt_timestamp value, the user could issue the command:

```
2 get list=das.ic1080_10hz.pitch.pkt_timestamp
```

Note that MCCS is responsible for determining which attributes are associated with each housekeeping data item and providing the user with the associated attributes when requested. In some cases, the ‘mcstime’ attribute is associated with a single data item. In other cases it is associated with a larger group of data items stored together. In the latter case, many data items may be submitted to MCCS as a single data block (identified as a ‘dataGroup’ in MCCS XML files). Upon receipt into MCCS, the PIS places a single timestamp on the entire data block; this is the ‘mcstime’ for all data items in that block of data. When the user then asks for any particular data item in the larger block and one or more attribute(s) of that data item, MCCS returns the associated attribute value(s) from larger block of data.

For example, to the command:

```
3 get list=ta_wfi.ccd_image_data.image_type (attr=mcstime)
```

MCCS could respond with:

```
3 : ta_wfi.ccd_image_data.focus_position=326.423(mcstime=2013-
    07-28T03:41:32.781Z)
```

In the above case, ‘ccd_image_data’ is the name of a block of data with many individual data items, but with a single ‘mcstime’ attribute for the whole data block.

Note that in all the above examples, “subscribe” could have been used instead of “get” to obtain values.

4.2.2.1 Simplified Access to Data Timestamps (planned)

MCCS provides a shorthand method to access data timestamps. The user simply adds “.time” to the name of the data item (i.e. <data item name>.time) in order to get the time closest to when that data item was acquired.

For example to retrieve and current GPS latitude and the time when the latitude was acquired, the user could use the command:

10 get list=[das.gps_1_10hz.gps_lat das.gps_1_10hz.gps_lat.time]

In cases where there is no time available giving when the data was acquired, the MCCS will provide the time when the data was received in the PIS. Thus the SCL response to a request for <data item>.time will use the time that the PIS received the data (currently called “mcstime” in housekeeping data definition XML files).

4.2.3 Generated Data Items

In addition to the availability of all the housekeeping data listed in Appendix B.4, the data items listed in Table 19 are generated dynamically by MCCS when requested with a get or subscribe command.

Note that while a user may use the SCL ‘get’ command to retrieve any of these at any time, the user should not use any of these to ‘trigger’ a subscribe command response (since there is no trigger condition for these besides the user’s request for the data). If a user subscribes to any of these data items, the user should trigger on arrival of a housekeeping data item identified in Appendix B.4, or do periodic data request using the ‘interval’ keyword of the subscribe command. Refer to Section 4.2.1 for more information on the SCL subscribe command.

Table 19. Generated Data Items

Name	Description
alias or aliases	list of all known aliases (name and what they are equivalent to) in the current session
all_alarms	list of the current alarm values in the system
all_alerts	list of the current values of all alert and alarm values in the system
all_audio	*list of the current values of all auditory alerts and alarms in the system*
all_commands	*list of all currently active commands in MCCS*
cfm_alarms	*list of all active alarms, which may be enabled, disabled, or (in the case of alarms) confirmed by the user*
cfm_alerts	*list of all active alerts and alarms, which may be enabled, disabled, or (in the case of alarms) confirmed by the user*
cfm_audio	*list of all active auditory alerts and alarms, which may be enabled, disabled, or (in the case of alarms) confirmed by the user*
cmd_list	list of available commands grouped by host then by commandable. Note that commands preceded by an "_" may not be issued by the user (they are for internal use only)
commands or active_commands	list of active commands for the session that the "get" command was sent to
data_list	content of the current data tree, presented in depth-first format
data_sources	list of all data sources (suppliers of housekeeping data) that are currently active in the system. Each data source is enclosed by its own set of braces ({ })
mission_id	current mission id defined in the file /mission_data/mission_id.txt. If the file is not found, a default mission id is defined as "MID_Undefined"
role	role of the the currently logged-in user
session_name	this session's name in the system. The name is in the format of [host].session_[username]_[role]_[integer]
sessions or active_sessions	*list of all sessions currently active in the system (i.e., everyone currently logged into the MCS). Each user session is listed on its own line, enclosed by braces ({ })*
time	current time
username	name of the currently logged-in user

4.2.4 Monitoring User-Specified Limits

A "watch" command, part of the "session" command group, can be set up by any user to monitor housekeeping data if they wish to be notified when particular housekeeping data items go outside a user-selected range, and when they transition back into the selected range. The usage for this command is as follows:

```
<msg id> watch trigger=<list> min=<minval> max=<maxval>
```

where

- `<msg id>` is a user-defined command identifier
- `<list>` is a list of one or more HK items to watch (enclose multiple in brackets ("`[...]`"); each item in the list is compared to the limits provided in the command
- `<minval>` is the minimum value below which to warn the user
- `<maxval>` is the maximum value above which to warn the user

If either or both minval and maxval are omitted, the “Low Limit” and “High Limit” values from Appendix B.4 for each listed housekeeping value will be used. If no limit(s) are present in Appendix B.4 for the selected items, then the limits are infinite (i.e. the minimum/maximum will never be exceeded).

For example, to monitor when focus position exceeds 0.35 millimeters, the user issues:

```
8 watch trigger=hipo.hipo_47_blue.ta_focus_position_si max=0.35
```

This causes the MCCS to monitor the indicated value and provide a response to the issuing session whenever the value goes outside the specified range (above 0.35 in this example), and a response whenever the value goes back in range (below 0.35).

The command will continue until stopped with a ‘cancel’ command:

```
9 cancel cmdid=8
```

The user can generate a "watch" warning for any (numeric) MCCS HK data item; these should generally be used to monitor the data used for triggering events, such as “watching” TA LOS in order to trigger an ROF update.

Note that the "watch" command should not be confused with Alerts and Alarms, which are built-in warnings to which the user can subscribe; see Section 4.3 for more on Alerts and Alarms.

4.3 Alerts and Alarms

Alerts and alarms are messages generated by the MCCS that provide to users information about its status or that of external systems. The only difference between an alert and an alarm message is that the latter requires confirmation from a user with appropriate privileges as described below. The information provided in alerts/alarms typically reflect either (1) messages hard-coded in the underlying software, or (2) housekeeping data item limits as defined in MCCS configuration XML files.

Each MCCS commandable has a pre-defined set of alerts/alarms, which are enabled by default at MCCS system start-up; each set can be disabled as a block by a user with the appropriately assigned privileges (e.g., ‘md’ per Table 3); similarly, once disabled, the permitted user can then again enable the disabled alerts/alarms.

To view and monitor alerts/alarms a user must subscribe to them. Each alert/alarm message has an associated severity level, as described in Table 20, which indicates

the criticality of the message; note that for housekeeping limits the WARNING and ERROR messages reflect, respectively, yellow and red high/low limit violations. Details on available alert/alarm messages are provided in Appendix G.

Audible alerts are sounds which are generated by MCCS when selected alert or alarm messages are generated. Two channels of audible alerts will be added in future MCCS development, one for mission director and one for telescope operator use. While all users may listen to audible alerts by selecting either or both of the audible-alert channels on MADS, only authorized mission operations staff are permitted to control them.

Note that the specific examples in this section are notional and thus may not be completely accurate. Operational examples will be included when these become available in MCCS.

Table 20. Alert/Alarm Severity Levels

Severity Level Number	Severity Level String	Description
0	INFO	Informational message (event)
1	WARNING	Warning message – notification of a potential problem, such as a housekeeping value going outside a warning (yellow) limit
2	ERROR	A error which is not fatal has occurred, such as a housekeeping value going outside a critical (red) limit
3	FATAL	A failure of some function of MCCS or an external subsystem has occurred resulting in a loss of MCCS capability; this may require a restart of a process or other action on the part of the user

4.3.1 Subscribing to Alerts/Alarms

An alert/alarm subscription command consists of the following:

```
<msg id> subscribe list=<value> [source=<value>] [level=<value>]
   [trigger=<value>]
```

Tokens in the above (<msg id>, <value>, =) are described in Table 5, while keywords are described in Table 21; immediately following that are a few examples.

Table 21. Alert/Alarm Subscribe Command Keywords

Keyword	Description
list	Items to retrieve. The value of the supplied argument can be an alert or alarm name or list option such as “all_alerts” or “all_alarms”. This argument can also be a bracketed list, permitting multiple alerts or alarms to be subscribed to.
trigger	The value of the supplied arguments is assumed to be the names of housekeeping data items. The one exception to this is the “all” value, which instructs the MCCS to trigger on all data items specified in the “list” argument.
source	Particular source from which to retrieve items.
level	Minimum severity level (0-3) for alert/alarm messages: INFO(0), WARNING(1), ERROR(2), and FATAL(3). For example, selecting INFO will return INFO, WARNING, ERROR and FATAL alerts/alarms, but selecting ERROR will return just ERROR and FATAL alerts/alarms.

For example:

```
49 subscribe level=0 list=all_alerts source=ta trigger=all
    showlabels=false
91 subscribe list=[ta_sma.ta_sma_alarm.ta_sma_alarm_fatal
    ta_sma.ta_sma_alarm.ta_sma_alarm_error
    ta_sma.ta_sma_alarm.ta_sma_alarm_warning
    ta_sma.ta_sma_alarm.ta_sma_alarm_info
    ta_sma.ta_sma_alert.ta_sma_alert_info
    ta_sma.ta_sma_alert.ta_sma_alert_warning
    ta_sma.ta_sma_alert.ta_sma_alert_error
    ta_sma.ta_sma_alert.ta_sma_alert_fatal]
    trigger=all showlabels=false
```

Each response to an alert/alarm subscription takes the following form:

<msg id> <respType> [<alert name>=] “<severity> <sep> <source> <sep> <index> <sep> *[<sound>]* <sep> [<status>] <sep> <alert text> <sep> <timestamp> <sep> [<allowed roles>]”

Where <msg id> and <respType> are described in Table 5, and other sub-fields are defined in Table 22. Note that the entire alert string is enclosed in double quotation marks.

Table 22. Alert/Alarm Response Tokens

Token	Description
<alert name>	The alert or alarm name. This name and the following ‘=’ is only included if the ‘showLabels’ flag was set to true in the subscribe command or was set to true globally for the session.
<severity>	Severity level of alert: FATAL, ERROR, WARNING, or INFO.
<sep>	Separator within the quoted alert/alarm string is the following three character sequence: single space, ASCII delete character (177 octal), single space.
<source>	Name of the source of the alert.
<index>	The index begins with “ID=” and the value supplied is the mnemonic ID of the alert.
<sound>	*Name of a sound that is played for an audible alert and is displayed in*

	parentheses.
<status>	The status field holds a "CLEARED", "CONFIRMED BY user", or "REISSUED *count* times" status within angle brackets.
<alert text>	Text message describing what is wrong, provides information, or asks an allowed user to take an action.
<timestamp>	Time (UT) that alert/alarm occurred in standard MCCS time format inside of square brackets: [YYYY-MM-DDThh:mm:ss.mmmZ] (e.g. '[2011-07-28T02:58:53.280Z]')
<allowed roles>	The list of roles begins with "role=" and continues with either "all", "none", or a square-bracketed, blank-separated list of operator types. The supplied list details which operator types are allowed to confirm/enable/disable alert – types that can change the datum.

For example:

```
49 I "INFO \177 ta \177 ID=TA_CONNECT \177  \177  \177 MCS connected to TA \177
    [2011-09-30T04:25:29.814Z] \177 "
49 I "ERROR \177 ta \177 ID=TA_DISCONNECT \177  \177  \177 MCS disconnected from
    TA \177 [2011-09-30T13:19:05.117Z] \177 "
91 I "" "" "" "" "INFO \177 ta_sma \177 ID=ta_sma \177  \177  \177 ^G \177
    [2011-09-30T07:39:53.917Z] \177 " "" "" ""
```

4.3.2 Confirming Alarms

Confirming an alarm is performed with the command:

```
<msg id> <source>.confirm list=<value> [id=<value>]
   [bool=<value>] [response=<value>]
```

with keywords as defined in Table 23, and a few examples following it. Note that the confirm command includes as a prefix the name of the source that issued the alarm (see <source> in Table 22). Upon receipt of this command the MCCS checks the user's role to verify privilege to confirm the alarm; if so, the alarm is confirmed, else an error is returned. Upon successful alarm confirmation the MCCS sends out an alert to notify users accordingly.

Table 23. Alarm Confirm Command Keywords

Keyword	Description
List	Items to confirm. This argument can also be a bracketed list, permitting multiple alarms to be confirmed.
Id	Mnemonic ID of alarm to confirm. If the id keyword is omitted from the command, then all alarms issued by the alert nodes specified in the list keyword will be confirmed.
Bool	Boolean response to an alarm.
Response	Textual response to an alarm.

```
24 hdg.confirm list=hdg_alarm_error id="HDG10"<CR>
26 wvm.confirm list=wvm_alarm_error id="WVM02"<CR>
```

The format for confirmed alarm responses is as follows:

```
<msg id> <respType> [message=]<confirm string>
```

Where "message=" is only included if the session or subscribe command has the "showLabels" flag set to true. The confirmation string (<confirm string>) takes the following format, if successful or unsuccessful, respectively:

```
<alert name> [ID=<ID>] confirmed by <user>
```

or

```
"Insufficient permission to confirm"
```

Table 24 defines the relevant keywords, and a few examples follow it.

Table 24. Confirm Command Responses

Token	Description
<confirm string>	ASCII confirm status string
<alert name>	Alert name
<ID>	Mnemonic ID of the alarm
<user>	Name of the user who confirmed the alarm

```
24 : message="hdg_alarm_error ID=HDG10 confirmed by Jackie
    Davidson"<LF>
26 F message="Insufficient permission to confirm"<LF>
```

4.3.3 Disabling/Enabling Alerts/Alarms

Disabling or re-enabling an alert/alarm is performed with the following:

```
<msg id> <source>.disable list=<value> [id=<value>]
<msg id> <source>.enable list=<value> [id=<value>]
```

with the associated keywords as defined in Table 25. Note that the actual disable or enable command is prefixed with the name of the issuing source reported in the alert/alarm. If the user possesses the proper role-based permissions, the alert/alarm is disabled or re-enabled as requested; else an error message is returned.

Table 25. Disabling Alert/Alarm Keywords

Keyword	Description
List	Items to disable. This argument can also be a bracketed list, permitting multiple alarms to be disabled.
Id	Mnemonic ID of alert to disable. *If the id keyword is omitted from the command, then all alerts/alarms issued by the alert nodes specified in the list keyword will be disabled.*

```
26 wvm_if.disable list=wvm_if_alert_info id=WVM02<CR>
56 wvm_if.enable list=wvm_if_alert_info<CR>
```

Disabled/enabled responses are of the following form:

```
<msg id> <respType> <keyword>=<disable string>
<msg id> <respType> <keyword>=<enable string>
```

where the disable/enable string takes on one of the following formats, based on whether or not the request is successful (based on the user's disable/enable permissions described above):

```
<alert name> ID=<ID> disabled
<alert name> ID=<ID> enabled
```

or

```
“Insufficient permission to disable”
“Insufficient permission to enable”
```

MCCS configuration files control which alerts/alarms may be disabled since some are considered too important to allow disabling. If the alert/alarm cannot be disabled, the quoted disable string will be:

```
“Cannot be disabled”
```

Table 26 provides the associated keyword definitions, and is followed by some usage examples.

Table 26. Disable Command Responses

Token	Description
<disable string>	ASCII disable status string returned with “message” keyword.
<alert name>	The alert name.
<ID>	The mnemonic ID of the alert or alarm.

```
23 : message="hdg_alert_info disabled"<LF>
26 F message="Insufficient permission to disable"<LF>
33 F message="Cannot be disabled"<LF>
43 : message="hdg_alert_info ID=HDG10 enabled"<LF>
56 F message="Insufficient permission to enable"<LF>
```

5. DEFINITIONS, ABBREVIATIONS, ACRONYMS

Abbreviation/Acronym	Description
<CR>	ASCII Carriage Return (a single byte containing the decimal value 13)
<CRLF>	ASCII Carriage Return, Line Feed (sequence of two bytes containing decimal values 13, then 10)
<EOT>	ASCII End Of Transmission (a single byte containing the value 4)
<LF>	ASCII Line Feed (a single byte containing decimal 10)
<SOH>	ASCII Start Of Header (a single byte containing the value 1)
ASCII	De-facto standard for representing 8-bit character data (American Standard Code for Information Interchange)
CDDS	Cavity Door Drive System
CECS	Cavity Environment Control System
CWR	Counter Weight Rack
DAS	Data Acquisition Subsystem
DHCP	Dynamic Host Configuration Protocol; allows computers to receive an IP address automatically from a DHCP server
DVDS	Digital Video Distribution Subsystem; composed of TAIPS & VPARS
EPO	Education and Public Outreach
FFI	Fine Field Imager; SOFIA 1 degree field of view imager
FITS	Flexible Image Transport System; a digital file format commonly used to store astronomical images and data
FM	Flight Manager; MCCS subsystem which controls heading throughout observation legs and provides in-flight flight planning functions to mission crew. (Planned for release in 2014.)
FMI	Flight Management Infrastructure; a subsystem used to track flight plan execution and provide in-flight flight plan modifications if needed. (This subsystem is expected to be replaced by the MCCS In-flight Planner portion of FM in 2014.)
FMS	Flight Management System; avionics system which provides aircraft route and other aircraft data
FPI	Focal Plane Imager; SOFIA 9 arcminute field of view imager
GI	General Investigator
HK	Housekeeping Data
ICD	Interface Control Document
IRIG-B	Inter-Range Instrumentation Group - (format) B

LAN	Local Area Network
MADS	Mission Audio Distribution Subsystem; SOFIA subsystem which provides audio communication between flight participants
MCCS	Mission Controls and Communications System
MD	Mission Director (formerly known as In Flight Director)
MCP	TA Master Control Processor
NFS	Network File System
NTP	Network Time Protocol
OSS	Observatory Support Subsystem
PI	Principal Investigator
PIS	Platform Interface Subsystem
SATA	Type of bus for connecting computers to mass storage devices (Serial Advanced Technology Attachment)
SCL	SOFIA Command Language
SFP	Science Flight Planner
SI	Science Instrument
SMA	TA Secondary Mirror Assembly
SMCU	Secondary Mirror Control Unit
SOFIA	Stratospheric Observatory for Infrared Astronomy
SSMO	SOFIA Science and Mission Operations
TA	Telescope Assembly
TAIPS	TA Imager Processing Subsystem; part of DVDS
TASCU	TA Servo Control Unit
TBD	To Be Determined
TCP/IP	Transmission Control Protocol/Internet Protocol
TO	Telescope Operator
VPARS	Video Processing and Recording Subsystem; part of DVDS
VPS	Vacuum Pump System
VME	Versa Module Eurocard; a common computer backplane bus
WFI	Wide Field Imager; SOFIA 6 degree field of view imager
WVM	Water Vapor Monitor
XML	Extensible Markup Language; a flexible text format for creating structured computer files

APPENDIX A. SCL COMMANDS

This appendix lists available SCL commands and associated arguments. Table A-1 provides an overview of the command categories and associated version information, while Table A-2 lists all SCL commands and their arguments.

Note that references to '<commandable>' in Tables A-1 and A-2 describe commands that are common to all commandable entities in MCCS, and there is no actual "<commandable>" entity to receive these commands. For example, 'coord.help' and 'nod.help' are valid commands that are only listed in the <commandable> section. Also, '[session]' is in brackets to denote that 'session.' is an optional part of session commands (e.g. 'session.get' and 'get' are equivalent).

Table A-1. SCL Command Categories Information

Commandable	Description
<commandable>	Generic commands that all commandables understand
[session]	Session commands
ars	Archiver
cdds	Cavity Door Drive System commands
coord	Coordinate commands
das	Data Acquisition Subsystem (DAS) commands
dither	Dither commands
fltexec	Flight Executor commands
nod	Nod commands
ntp	NTP server commands
rien	Template for Science Instrument (SI) commands ("rien" is a fictitious SI name)
sma	Secondary Mirror Assembly (SMA) commands
ta	Generic Telescope Assembly (TA) commands (not subsystem specific)
ta_ffi	Fine Field Imager (FFI) commands
ta_fpi	Focal Plane Imager (FPI) commands
ta_pos	TA Position commands
ta_state	TA State commands
ta_wfi	Wide Field Imager commands
wvm_if	Water Vapor Monitor (WVM) commands

In Table A-2 the header at the top of each page identifies the associated command and argument information according to the following:

- Command – full name of command (commandable.command)
- Command Description – functional description of command
- Argument – name of command argument
  Note that references to 'ANY' refer to the user-assigned or pre-defined name of a position or offset (refer to Section 4.1.3 for further information), or a settable item within the named commandable entity.
- Argument Description – functional description of command argument
- Representation – how command argument is stored in MCCS, as defined in Table 8, Binary Data Types

- Units – default units for command argument, if applicable
- Low Limit – low limit for command argument, if applicable, in above-noted units
- High Limit – high limit for command argument, if applicable, in above-noted units
- Value Type – type of command argument (e.g., a position, user-settable)

Changes from the previous revision of the ICD (D) are identified in the table below as follows. Unchanged items are in black text. New items are in blue text. Items which have been deleted are in ~~red strike-through text~~.

Table A-2: SCL Command Table
(starts on next page)

Command	Command Description					
Argument	**Argument Description**	**Representation**	**Units**	**Low Limit**	**High Limit**	**Value Type**
<commandable>.alive	Queries the application to see if it is alive					
<commandable>.cancel	Cancel a particular command					
cmdid	cmd ID to identify the command to cancel	INT4	none			NONE
sysid	sys ID to identify the command to cancel	INT4	none			NONE
<commandable>.confirm	acknowledge and dismiss an alarm					
bool	boolean response to an alarm	STRING	none			NONE
id	Mnemonic ID of alarm to confirm	STRING	none			NONE
list	items to confirm	STRING	none			NONE
response	textual response to an alarm	STRING	none			NONE
<commandable>.disable	keep an alert or alarm from being issued					
id	Mnemonic ID of alarm to disable	STRING	none			NONE
list	items to disable	STRING	none			NONE
<commandable>.enable	re-enable an alert or alarm, allow it to resume issuing messages					
id	Mnemonic ID of alarm to enable	STRING	none			NONE
list	items to enable	STRING	none			NONE
<commandable>.help	print brief information for available commands and/or data items					
argument	get help on a specific command argument	STRING	none			NONE
command	get help on a specific commandable command	STRING	none			NONE
data_item	get help on a specific housekeeping data item	STRING	none			NONE
<commandable>.pause_archive	Pause archiving HK data					
<commandable>.resume_archive	Resume archiving HK data					
<commandable>.set_bytes_archived_archive_nsample	Set the archive nsample for bytes archived HK data					
datagroup	Data group name, (Precision: N/A) , (Accuracy: N/A) , [ENUM: (1: current_bytes_archived) (2: total_bytes_archived)]	STRING	none	0	1	NONE
nsample	archive n sample value, (Precision: N/A) , (Accuracy: N/A) , (Default Type: FIXED, Value: 0)	UINT4	none	0	4294967295	NONE
<commandable>.start_archive	Turn archive on and start archiving HK data					
<commandable>.stop_archive	Turn archive off					
[session.]alias	Create an alias					
ANY	Item to alias	STRING	N/A			NONE

Command	Command Description					
Argument	**Argument Description**	**Representation**	**Units**	**Low Limit**	**High Limit**	**Value Type**
[session.]comment	~~Create an alias~~ Enter a log comment					
file	Name of the file to write the comment to. Comments are written to file by writing the time the comment was entered followed by the comment text. If the file specified does not exist, it will be created, and if the file specified does exist, it will be appended to.	STRING	N/A			NONE
text	The text of the comment to be entered	STRING	N/A			NONE
[session.]get	retrieve one or more housekeeping values					
list	items to retrieve	STRING	N/A			GET
[session.]login	Login to the MCCS					
autosend	Flag indicating whether to send the SCL in the [user].xml upon login, (Default Type: FIXED, Value: yes) , [ENUM: (0: no) (1: yes)]	BOOL4	N/A			NONE
password	Associated authentication info (password)	STRING	N/A			NONE
port	Port number to connect to client at (optional)	INT4	N/A			NONE
role	Role being played by this user	STRING	N/A			NONE
user	User of the MCCS	STRING	N/A			NONE
[session.]logout	Logout from the MCCS, no arguments required					
[session.]revision	Report MCCS revision information					
[session.]set	set the value for a housekeeping parameter					
ANY	item to set	STRING	N/A			SET
[session.]subscribe	subscribe to housekeeping values					
duration	amount of time to keep displaying	FLOAT4	seconds	0.00001		SET
file	File used for logging, output goes to specified file	STRING	N/A			NONE
format	Format to output the log file in, (Default Type: CALCULATED, Value:) , [ENUM: (5: fits) (2: ascii) (3: table) (6: mcs)]	STRING	N/A			NONE
interval	Amount of time between each display	FLOAT4	seconds	0.00001		SET
level	A particular severity level at which to retrieve items, [NUMBER: (3: fatal) (2: error) (1: warning) (0: info)]	INT4	N/A			NONE
list	items to retrieve	STRING	N/A			GET
n_times	Number of times to display, (Default Type: CALCULATED, Value:)	INT4	N/A	1		NONE
sample	Take every 'sample_intvl'th sample, (Default Type: FIXED, Value: 1)	INT4	N/A	1		NONE
source	Particular source from which to retrieve items	STRING	N/A			NONE
trigger	Item used to determine when to take the next sample, (Default Type: CALCULATED, Value:)	STRING	N/A			NONE
[session.]unalias	undo a previously created alias					
token	Item to unalias	STRING	N/A			NONE
[session.]watch	Watch for an item to go out-of-range					

Command	Command Description					
Argument	**Argument Description**	**Representation**	**Units**	**Low Limit**	**High Limit**	**Value Type**
max	Maximum acceptable value for item(s), (Default Type: CALCULATED, Value:)	FLOAT8	N/A			NONE
min	Minimum acceptable value for item(s), (Default Type: CALCULATED, Value:)	FLOAT8	N/A			NONE
trigger	Item(s) on which to trigger	STRING	N/A			NONE
ars.debug	Outputs software debug information					
ars.set_archive_nsample	Set the archive nsample for a data group					
datagroup	Data group name, (Precision: N/A) , (Accuracy: N/A)	STRING	none	1	87	NONE
nsample	archive n sample value, (Precision: N/A) , (Accuracy: N/A) , (Default Type: FIXED, Value: 1)	UINT4	none	0	4294967295	NONE
cdds.set_aperture	Sets the door at a fixed position.					
position	Specifies the new desired aperture position , in degrees. Turns off tracking if it is on.	FLOAT8	degrees	24.15	57.35	SET
cdds.set_archive_nsample	Set the archive nsample for a data group					
datagroup	Data group name, (Precision: N/A) , (Accuracy: N/A) , [ENUM: (1: urd_position) (2: urd_load) (3: urd_velocity) (4: lfd_position) (5: lfd_load) (6: lfd_velocity) (7: urd_seal_status) (8: urd_seal_vacuum_status) (9: urd_seal_valve_status) (10: urd_seal_supply_pressure) (11: urd_seal_stepped_down_pressure) (12: urd_seal_pressure) (13: urd_seal_vent_pressure) (14: aperture_limit_switches) (15: urd_seal_system_fault) (16: ifd_console_switch_setting) (17: ifd_enable_switch_setting)]	STRING	none	1	18	NONE
nsample	archive n sample value, (Precision: N/A) , (Accuracy: N/A) , (Default Type: FIXED, Value: 1)	UINT4	none	0	4294967295	NONE
cdds.track_aperture	Makes the door track the telescope, with an offset.					
offset	Specifies the new desired aperture tracking offset, in degrees. Turns on tracking if it is off., (Default Type: FIXED, Value: 0.0)	FLOAT8	degrees			SET
coord.aoi_create	Define a tracker AOI from an MCCS position, which need not be visible in the field of view of any imager					
aoi_id	The AOI to be defined in the tracker	UINT4	N/A	1	8	NONE
beam_id	Defines the beam in which the tracker shall stabilize on the AOI. This parameter is only evaluated and effective when the AOI is defined on the FPI. [ENUM: (0:OFF) (1:MIDPOINT) (2:PLUS_BEAM) (3:MINUS_BEAM)]	STRING				NONE
calculation	Specifies the kind of calculation performed on the AOI [ENUM: (0:OFF) (1:CENTROIDING) (2:LIMB)]	STRING	N/A			NONE
camera	Designates imager to create the AOI on [ENUM: (0:WFI) (1:FFI) (2:FPI)]	STRING	N/A			NONE
column_size	Defines the number of pixel columns in the AOI (adding columns to the right from the starting column)	INT4	N/A	3	150	NONE
pos	Existing position name from which to create an AOI	STRING	N/A			POSITION

Command	Command Description					
Argument	**Argument Description**	**Representation**	**Units**	**Low Limit**	**High Limit**	**Value Type**
row_size	Defines the number of pixel rows in the AOI (counting down from the starting row)	INT4	N/A	3	150	NONE
coord.aoi_delete	Clear a tracker AOI and its referencing within the MCCS, and reset any tracker velocities for this AOI to zero.					
aoi_id	The AOI to be deleted from the system	UINT4	N/A	0	8	NONE
coord.aoi_ref	Modifies and publishes reference position for AOI					
aoi	AOI name used for reference position. Specify trc_aoi1-8	STRING	N/A			NONE
pos	Name of the Position to obtain reference data	STRING	N/A			POSITION
coord.aoi_ref_clear	Clears the AOI reference position					
aoi	AOI name to clear reference position. Specify trc_aoi1-8	STRING				NONE
coord.convert	Get the equivalent Position in another reference frame					
equinox	Equinox to use for precession of output coordinates. Use is optional. Only anticipated use is for ERF and EcRF output coordinates.	STRING	julian years			NONE
position	Position to be converted	STRING	N/A			POSITION
save_as	*Name to save the result under*	*STRING*	*N/A*			*NONE*
to	Reference Frame or coordinate system to rotate to, [ENUM: (0: ERF) (1: TARF) (2: WFIRF) (3: FFIRF) (4: FPIRF) (5: SIRF) (6: EcRF) (7: GalRF) (8: HRF) (9: IRF)]	STRING	N/A			NONE
coord.correct	Update ERF to IRF calibrations					
dec	Actual declination for the specified Position	SEXA8	degrees			SET
images	Number of centroids to consider when correcting the ERF-IRF definition with a centroid	UINT4	N/A	1		SET
old_dec	Uncorrected celestial latitude	SEXA8	degrees			SET
old_ra	Uncorrected celestial longitude	SEXA8	sidereal hours			SET
old_vpa	Uncorrected position angle	FLOAT8	degrees			SET
position	Known location, to be fixed	STRING	N/A			POSITION
ra	Actual RA for the specified Position	SEXA8	sidereal hours			SET
vpa	Actual position angle for the specified Position, (Default Type: CALCULATED, Value:)	SEXA8	degrees			SET
coord.delete	Delete a named object (Position, RefFrame, etc.)					
action	Confirm the deletion?, (Default Type: SETTABLE, Value: coord.delete.action) , [ENUM: (0: ignore) (1: query)]	STRING	N/A			NONE
name	Object or objects to delete	STRING	N/A			NONE

Command	Command Description					
Argument	**Argument Description**	**Representation**	**Units**	**Low Limit**	**High Limit**	**Value Type**
coord.init	Initialize ERF to IRF calibrations					
coord.instrument	Instrument initialization and set-up					
name	Login name for the SI user (without _si)	STRING	N/A			NONE
coord.list	Output a list of named objects					
full	Full or brief output, [ENUM: (0: no) (1: yes)]	BOOL4	N/A			NONE
name	Name of objects which the user wishes listed	STRING	N/A			NONE
sorted_by	Sort by index or name, (Default Type: CALCULATED, Value:) , [ENUM: (-1: index) (-1: array) (1: alpha)]	STRING	N/A			NONE
type	Kind of objects which the user wishes listed, [ENUM: (0x000001: position) (0x000003: aoi) (0x000007: ta_aoi) (0x000007: taaoi) (0x000010: rf) (0x000300: offsets) (0x001000: chops) (0x002001: beams) (0x082000: nodsets) (0x000009: guide) (0x000009: guide_star)]	STRING	N/A			NONE
coord.position	Add a Position to the list of named objects					
action	What to do if new argument is wrong?, (Default Type: SETTABLE, Value: coord.position.action) , [ENUM: (-1: error) (0: ignore) (1: query)]	STRING	N/A			NONE
name	Name for the new/modified Position	STRING	N/A			POSITION
new	Creating a new object?, (Default Type: FIXED, Value: yes) , [ENUM: (0: no) (1: yes)]	BOOL4	N/A			NONE
coord.quiet	Suspends debugging output					
coord.set	Set the value for a mode or parameter					
ANY	The keyword of this argument is the name of the item or array to set; the value supplied in this argument is the new value for the item or specified array element.	STRING	N/A			INDEXED
coord.set_archive_nsample	Set the archive nsample for a data group					
datagroup	Data group name, (Precision: N/A) , (Accuracy: N/A) , [ENUM: (1: aoa) (2: drift_axis) (3: lst) (4: drift_rate) (5: aoi1.name) (6: aoi1.erf) (7: aoi2.name) (8: aoi2.erf) (9: aoi3.name) (10: aoi3.erf) (11: aoi4.name) (12: aoi4.erf) (13: aoi5.name) (14: aoi5.erf) (15: aoi6.name) (16: aoi6.erf) (17: aoi7.name) (18: aoi7.erf) (19: aoi8.name) (20: aoi8.erf)]	STRING	none	1	13	NONE
nsample	archive n sample value, (Precision: N/A) , (Accuracy: N/A) , (Default Type: FIXED, Value: 1)	UINT4	none	0	4294967295	NONE
coord.set_pos_rate	Selects position publication rates					
def_nsample	Number of TASCU HK samples between computation of default positions (default positions are all positions which are not selected as fast positions)	UINT2	N/A	1	50	NONE
fast_nsample	Number of TASCU HK samples between computation of fast positions (fast positions are those selected with ‘fast_pos_list’ argument)	UINT2	N/A	1	50	NONE
fast_pos_list	Position names to publish at rate chosen by ‘fast_nsample’	STRING	N/A			NONE
coord.verbose	Enables debugging output					

Command	Command Description					
Argument	**Argument Description**	**Representation**	**Units**	**Low Limit**	**High Limit**	**Value Type**
das.clear_faults	Clears all latched faults on DAS					
das.disable_group	Disables the specified group to be sent to the PIS for processing					
group	Name of group containing housekeeping data	STRING	none			NONE
das.enable_group	Enables the specified group to be sent to the PIS for processing					
group	Name of group containing housekeeping data	STRING	none			NONE
das.reboot	Reboots the DAS					
das.shutdown	Performs an orderly shutdown of all DAS software					
dither.define	Define a new dither set					
pos	One or more beam positions starting a dither set	STRING	N/A			POSITION
dither.goto	Go to a beam position other than the next					
pos	One target beam position to move to	STRING	N/A			NONE
dither.insert	Add beam positions to a defined dither set					
before	Name of an existing position in the dither set before which the new positions will be inserted. If omitted, positions identified by the 'pos' argument will be appended to the end of the existing dither set.	STRING	N/A			POSITION
pos	One or more beam positions to add to a dither set	STRING	N/A			POSITION
dither.next	Go to the next beam position (dither)					
dither.quiet	Suspends debugging output					
dither.remove	Remove beam positions from a defined dither set					
pos	Removes specified dither positions from list, if they all exist	STRING	N/A			NONE
dither.rotate	Rotates the dither set about the designated boresight					
angle	desired rotation angle	FLOAT8	degrees			NONE
dither.set	Set the value for a mode or parameter					
ANY	The keyword of this argument is the name of the item or array to set; the value supplied in this argument is the new value for the item or specified array element.	STRING	N/A			INDEXED
dither.verbose	Enables debugging output					
fltexec.quiet	Suspends debugging output					
fltexec.resume	Enables the application					

Command	Command Description					
Argument	**Argument Description**	**Representation**	**Units**	**Low Limit**	**High Limit**	**Value Type**
fltexec.set	Set the value for a mode or parameter					
ANY	The keyword of this argument is the name of the item or array to set; the value supplied in this argument is the new value for the item or specified array element.	STRING	N/A			INDEXED
fltexec.suspend	Suspends the application					
fltexec.verbose	Enables debugging output					
nod.correct	Adjusts a nod beam position					
beam	Beam position to be corrected	STRING	N/A			NONE
images	*Number of centroids to use when correcting the nod definition with a centroid. (Default Type: FIXED, Value: 1)*	*UINT4*	*N/A*	*1*		*SET*
pos	Name of an AOI to center the beam on	STRING	N/A			NONE
nod.define	Define a set of nod Beams					
amp2	Secondary nodding amplitude, (Default Type: SETTABLE, Value: nod.define.amp2)	FLOAT8	arcseconds			SET
amplitude	Nodding amplitude, (Default Type: SETTABLE, Value: nod.define.amplitude)	FLOAT8	arcseconds			SET
coord_sys	coordinate system used to define the nod set, (Default Type: FIXED, Value: tarf) , [ENUM: (1: tarf) (2: erf) (3: wfirf) (4: ffirf) (5: fpirf) (6: sirf)]	STRING	N/A			NONE
pos_angle	Nod angle measured clockwise from +Y axis in SIRF, +EL in TARF, and N in ERF. Note that the "clockwise" direction definition is for an image that is not a mirror image of the sky; if it is mirror image the direction will reverse. (Default Type: SETTABLE, Value: nod.define.pos_angle)	FLOAT8	degrees			SET
profile	Number of positions within the nod set, (Default Type: FIXED, Value: 2)	UINT4	N/A	2	3	NONE
track_pos_a	Name of position to use for tracking when at beam A (Default Type: FIXED, Value:)	STRING	N/A			NONE
track_pos_b	Name of position to use for tracking when at beam B (Default Type: FIXED, Value:)	STRING	N/A			NONE
track_pos_x	Name of position to use for tracking when at beam X (Default Type: FIXED, Value:)	STRING	N/A			NONE
nod.goto	Go to a nod position directly					
pos	One target nod position to move to	STRING	N/A			NONE
nod.next	Go to the next beam position (nod)					
nod.quiet	Suspends debugging output					
nod.set	Set the value for a mode or parameter					

Command	Command Description					
Argument	**Argument Description**	**Representation**	**Units**	**Low Limit**	**High Limit**	**Value Type**
ANY	The keyword of this argument is the name of the item or array to set; the value supplied in this argument is the new value for the item or specified array element.	STRING	N/A			INDEXED
nod.verbose	Enables debugging output					
ntp.debug	Outputs software debug information					
cmd	Test command to be sent to archiver over telnet port.	STRING	none			NONE
ntp.set_archive_nsample	Set the archive nsample for a data group					
datagroup	Data group name, (Precision: N/A) , (Accuracy: N/A)	STRING	none	1	87	NONE
nsample	archive n sample value, (Precision: N/A) , (Accuracy: N/A) , (Default Type: FIXED, Value: 1)	UINT4	none	0	4294967295	NONE
rien.mode	Changes the current instrument mode					
change_focus	Refocussing actions, (Default Type: FIXED, Value: neither) , [ENUM: (0: neither) (1: sma) (2: fpi) (3: both)]	STRING				NONE
current	New operating mode, (Default Type: SETTABLE, Value: rien.si_config.current_mode)	STRING				NONE
rien.set	Set the value for a mode or parameter					
ANY	The keyword of this argument is the name of the item or array to set; the value supplied in this argument is the new value for the item or specified array element.	STRING	N/A			INDEXED
rien.si_align	Update the Science Instrument alignment					
images	Number of images to use in the alignment process if position is an AOI	UINT4	N/A	1		SET
pos	Name of the Position to boresight to	STRING	N/A			POSITION
rien.si_boresight	Update the position of the SI Boresight					
images	Number of images to use in the boresighting process if position is an AOI	UINT4	N/A	0		SET
pos	Name of the Position to boresight to	STRING	N/A			POSITION
rien.sibs_change	Change the Science Instrument boresight					
move	Look at the same sky location?, (Default Type: FIXED, Value: no) , [ENUM: (0: no) (1: yes)]	BOOL4	N/A			NONE
xsi	SI pixel location of boresight (abscissa)	FLOAT8	pixels			NONE
ysi	SI pixel location of boresight (ordinate)	FLOAT8	pixels			NONE
sma.align_fcm	Measures FCM alignment and informs TA of FCM scale and angle offset so that it can properly account for the misalignment.					
angle	Position angle of FCM in SMARF	FLOAT8	degrees			SET
aoi	An AOI on the Focal Plane Imager (FPI) that is actively centroiding. Specify trc_aoi1-8.	STRING	N/A			POSITION
dist	Distance of the FCM deflection on the sky that will be used to measure the misalignment.	FLOAT8	arcseconds			SET

Command	Command Description					
Argument	**Argument Description**	**Representation**	**Units**	**Low Limit**	**High Limit**	**Value Type**
images	Number of centroids to consider when determining the alignment.	UINT4	N/A			SET
sma.align_tcm	Measures TCM alignment and informs TA of TCM scale and angle offset so that it can properly account for the misalignment.					
angle	Position angle of TCM in SMARF	FLOAT8	degrees			SET
aoi	An AOI on the Focal Plane Imager (FPI) that is actively centroiding. Specify trc_aoi1-8.	STRING	N/A			POSITION
dist	Distance of the TCM deflection on the sky that will be used to measure the misalignment.	FLOAT8	arcseconds			SET
images	Number of centroids to consider when determining the alignment.	UINT4	N/A			SET
sma.chop	Set up and start chopping					
amp2	Secondary chop amplitude, (Default Type: SETTABLE, Value: sma.chop.user_amp2)	FLOAT8	arcseconds	0	1125	SET
amplitude	Chop amplitude, (Default Type: SETTABLE, Value: sma.chop.user_amplitude)	FLOAT8	arcseconds	0	1125	SET
coord_sys	Coordinate system for pos_angle, (Default Type: FIXED, Value: TARF) , [ENUM: (0: IRF) (1: TARF) (2: TAICRF) (3: ERF) (4: WFIRF) (5: FFIRF) (6: FPIRF) (7: SIRF)]	STRING	N/A			NONE
frequency	TCM Chop frequency, (Precision: 0.5) , (Default Type: SETTABLE, Value: sma.chop.frequency)	FLOAT8	hertz	0	20	SET
phase	Time delay between synch signal and chop, (Default Type: SETTABLE_REQUIRED_ONCE, Value: sma.chop.phase)	FLOAT8	milliseconds	0	1000	SET
pos_ang	Chop angle measured clockwise from +Y axis in SIRF, +EL in TARF, and N in ERF. Note that the "clockwise" direction definition is for an image that is not a mirror image of the sky; if it is mirror image the direction will reverse. (Default Type: SETTABLE_REQUIRED_ONCE, Value: sma.chop.pos_ang)	FLOAT8	degrees	-180	360	SET
profile	Number of chop points, (Default Type: SETTABLE_REQUIRED_ONCE, Value: sma.chop.profile)	UINT4	N/A	2		NONE
settling_time	Minimum time interval for mirror to stabilize, (Precision: 0.1) , (Default Type: SETTABLE_REQUIRED_ONCE, Value: sma.chop.settling_time)	FLOAT8	milliseconds	5	100	SET
start	Start chopping after download?, (Default Type: FIXED, Value: yes) , [ENUM: (0: no) (1: yes)]	BOOL4	N/A			NONE
sync_src	Internal or external chop synch signal?, (Default Type: SETTABLE_REQUIRED_ONCE, Value: sma.chop.sync_src) , [ENUM: (0: internal) (1: external) (3: analog)]	STRING	N/A			NONE
tilt	Deflection of SMA orthogonal to the chop throw, (Default Type: SETTABLE_REQUIRED_ONCE, Value: sma.chop.user_tilt)	FLOAT8	arcseconds	-1125	1125	SET
tip	Deflection of SMA parallel to the chop throw, (Default Type: SETTABLE_REQUIRED_ONCE, Value: sma.chop.user_tip)	FLOAT8	arcseconds	-1125	1125	SET
sma.focus	Move the secondary mirror along the focus axis					
offset	Focus offset in the T axis, (Precision: 1.0) , (Accuracy: 1.0)	FLOAT8	micrometers	-10000	10000	SET
position	Focus position in the T axis, (Precision: 1.0) , (Accuracy: 10.0)	FLOAT8	micrometers	-5000	5000	SET
temp_focus	Move to predefined back focus, [ENUM: (0: reset_offset) (4: here) (8: use)]	STRING	N/A			SET

Command	Command Description					
Argument	**Argument Description**	**Representation**	**Units**	**Low Limit**	**High Limit**	**Value Type**
sma.init	Reinitializes or restarts the SMA					
sma.park	Park the SMA					
shutdown	Power off the SMA, (Default Type: FIXED, Value: no) , [ENUM: (0: no) (1: yes)]	BOOL4	N/A			NONE
sma.position	*Reposition the secondary mirror*					
r	*New position in the R axis, (Precision: 1.0) , (Accuracy: 4.0) , (Default Type: SETTABLE, Value: sma.position.r)*	*FLOAT8*	*micrometers*	*-5000*	*5000*	*SET*
s	*New position in the S axis, (Precision: 1.0) , (Accuracy: 4.0) , (Default Type: SETTABLE, Value: sma.position.s)*	*FLOAT8*	*micrometers*	*-5000*	*5000*	*SET*
tilt	*Rotation around the S axis, (Precision: 0.7) , (Accuracy: 2.0) , (Default Type: SETTABLE, Value: sma.position.tilt)*	*FLOAT8*	*arcseconds*	*-1800*	*1800*	*SET*
tip	*Rotation around the R axis, (Precision: 0.7) , (Accuracy: 2.0) , (Default Type: SETTABLE, Value: sma.position.tip)*	*FLOAT8*	*arcseconds*	*-1800*	*1800*	*SET*
sma.quiet	Suspends debugging output					
sma.reset	Resets TCM and FCM parameters to zero and stops the chopper					
sma.set	Set the value for a mode or parameter					
ANY	The keyword of this argument is the name of the item or array to set; the value supplied in this argument is the new value for the item or specified array element.	STRING	N/A			INDEXED
sma.temp_focus	Turn on/off the auto-focus					
auto_enable	Auto enable status, (Default Type: SETTABLE_REQUIRED_ONCE, Value: sma.temp_focus.auto_enable) , [ENUM: (0:false) (1:true)]	STRING	N/A			SET
auto_max_change	Maximum delta change for autofocus, (Default Type: SETTABLE_REQUIRED_ONCE, Value: sma.temp_focus.auto_max_change)	FLOAT8	micrometers	1	5000	SET
auto_min_change	Minimum delta change for autofocus, (Default Type: SETTABLE_REQUIRED_ONCE, Value: sma.temp_focus.auto_min_change)	FLOAT8	micrometers	1	5000	SET
sma.verbose	Enables debugging output					
ta.cmd_timeout	Turns on and off the automatic setting of a timer for TA commands (system default is to set a timer)					
enable	Specifies whether or not to set a command timer, [ENUM: (0: false) (1: true)]	BOOL4	none			NONE
ta.debug	Outputs software debug information					
ta.poll	Turns on and off polling to the TA interfaces (system default is to have polling on)					
state	Specifies whether to turn polling on or off, [ENUM: (0: false) (1: true)]	BOOL4	none			NONE
system	The TA subsystem (MCP or TRC) to affect the polling for	STRING	none			NONE

Command	Command Description					
Argument	**Argument Description**	**Representation**	**Units**	**Low Limit**	**High Limit**	**Value Type**
ta_ffi.debug	Outputs software debug information					
ta_ffi.image_to_hk	Causes the next received image to be sent to HK data					
comment	If the format type is fits, then the contents of this argument will be added as a comment to the FITS header	STRING	N/A			NONE
type	The format of the HK data (binary or fits), [ENUM: (0: binary) (1: fits)]	STRING	N/A			NONE
ta_ffi.log	Causes the next received image to be written to file					
comment	If the type of the file is fits, then the contents of this argument will be added as a comment to the FITS header	STRING	N/A			NONE
file	The file to write the image data to, do not add the file extension.	STRING	N/A			NONE
n_times	Number of image files to write.	INT4	N/A	1	100	NONE
type	The format of the file data (binary or fits), [ENUM: (0: binary) (1: fits)]	STRING	N/A			NONE
ta_ffi.retransmit_packets	Indicates whether or not to re-transmit packets received from the imagers to the PI Patch Panel (system default is not to re-transmit the packets). The user may also specify a host and a port to transmit to- if a host and port are not specified, then the default host and port are used.					
host	The domain name or IP address of the host to re-transmit received imager packets to	STRING	N/A			NONE
port	The port number to re-transmit received imager packets to	UINT2	N/A			NONE
state	A value of true causes packets to be re-transmitted to the PI Patch Panel while a value of false causes packets not to be re-transmitted, [ENUM: (0: no) (1: yes)]	BOOL4	N/A			NONE
ta_ffi.show_test_images	Indicates whether or not to process test images sent from the TA					
state	A value of true causes test images to be processed while a value of false causes test images to be ignored, [ENUM: (0: no) (1: yes)]	BOOL4	N/A			NONE
ta_fpi.debug	Outputs software debug information					
ta_fpi.image_to_hk	Causes the next received image to be sent to HK data					
comment	If the format type is fits, then the contents of this argument will be added as a comment to the FITS header	STRING	N/A			NONE
type	The format of the HK data (binary or fits), [ENUM: (0: binary) (1: fits)]	STRING	N/A			NONE
ta_fpi.log	Causes the next received image(s) to be written to file					
comment	If the type of the file is fits, then the contents of this argument will be added as a comment to the FITS header	STRING	N/A			NONE
file	The file to write the image data to, do not add the file extension.	STRING	N/A			NONE
n_times	Number of image files to write.	INT4	N/A	1	100	NONE
type	The format of the file data (binary or fits), [ENUM: (0: binary) (1: fits)]	STRING	N/A			NONE
ta_fpi.retransmit_packets	Indicates whether or not to re-transmit packets received from the imagers to the PI Patch Panel (system default is not to re-transmit the packets). The user may also specify a host and a port to transmit to- if a host and port are not specified, then the default host and port are used.					

Command	Command Description					
Argument	**Argument Description**	**Representation**	**Units**	**Low Limit**	**High Limit**	**Value Type**
host	The domain name or IP address of the host to re-transmit received imager packets to	STRING	N/A			NONE
port	The port number to re-transmit received imager packets to	UINT2	N/A			NONE
state	A value of true causes packets to be re-transmitted to the PI Patch Panel while a value of false causes packets not to be re-transmitted, [ENUM: (0: no) (1: yes)]	BOOL4	N/A			NONE
ta_fpi.show_test_images	Indicates whether or not to process test images sent from the TA					
state	A value of true causes test images to be processed while a value of false causes test images to be ignored, [ENUM: (0: no) (1: yes)]	BOOL4	N/A			NONE
ta_pos.autofocus	Autofocus the FPI trombone					
aoi	Star to measure seeing on, (Default Type: SETTABLE, Value: ta_pos.autofocus.aoi)	STRING	N/A			AOI_SET
timeout	Maximum time allowed for command, (Default Type: SETTABLE_REQUIRED_ONCE, Value: ta_pos.autofocus.timeout)	UINT4	seconds	1	90	NONE
ta_pos.fpi_focus	Move the FPI trombone					
back_focus	Use SI-supplied back focus position?, (Default Type: FIXED, Value: no) , [ENUM: (0: no) (1: yes)]	BOOL4	N/A			NONE
offset	Relative focus position	FLOAT8	millimeters	-1200	1200	SET
position	Absolute focus position	FLOAT8	millimeters	-300	900	SET
ta_pos.goto	Absolute move telescope to supplied position					
los	Desired LOS position	FLOAT8	degrees	-3	3	SET
pos	Name of the Position to go to	STRING	N/A			POSITION
reference	Record the specified Position as a reference	STRING	N/A			POSITION
rof	Desired Rotation of Field	FLOAT8	degrees	-180	360	SET
track_mode	Enable tracking?, (Default Type: CALCULATED, Value:) , [ENUM: (0: none) (0: off) (1: centroid) (3: rof) (4: limb) (7: on)]	STRING	N/A			NONE
vaa	Desired Vertical Aperture Angle	FLOAT8	degrees	-180	360	SET
ta_pos.imager_align	Compensates MCCS for imager misalignment					
corr	AOI defined on the imager to be corrected. Specify trc_aoi1-8	STRING	N/A			POSITION
images	The number of images to use in the alignment procedure.	UINT4	N/A			SET
std	AOI defined on the imager assumed to be aligned; this is usually an AOI on the FPI. Specify trc_aoi1-8	STRING	N/A			POSITION
ta_pos.pattern	Complicated movement through points					
chop	Select the chop position in which to scan	STRING	N/A			NONE
coord_sys	Coordinate system for coordinate axes, (Default Type: SETTABLE, Value: ta_pos.pattern.coord_sys) , [ENUM: (1: TARF) (3: ERF) (4: WFIRF) (5: FFIRF) (6: FPIRF) (7: SIRF) (8:GALRF) (9: ECRF)]	STRING	N/A			NONE
extra_points	Number of user requested extra ramp points, (Default Type: FIXED, Value: 2)	INT4	N/A	0	17	NONE

Command	Command Description					
Argument	**Argument Description**	**Representation**	**Units**	**Low Limit**	**High Limit**	**Value Type**
points	Array of 2+ positions (coordinate pairs) to scan	VECTOR16	N/A			NONE
rate	Speed at which the points will be scanned, (Default Type: SETTABLE, Value: ta_pos.pattern.rate)	FLOAT8	arcseconds per second	0	3609.63411	SET
sample_spacing	Distance between sample points along the curve – set to 0.0 to allow the pattern command to choose a distance equal to 0.5sec * rate.	FLOAT8	arcseconds	0	1000	SET
start_delay	Time until passing through the first point, (Default Type: CALCULATED, Value:)	FLOAT8	seconds	0	3600	SET
start_time	Time at which to pass through the first point, (Default Type: CALCULATED, Value:)	SEXA8	hours	0	25	NONE
type	What kind of scan to do, (Default Type: FIXED, Value: linear) , [ENUM: (1: step) (2: smooth) (3: linear)]	STRING	N/A			SET
ta_pos.quiet	Suspends debugging output					
ta_pos.rewind	Rewind the telescope around LOS axis					
beam	Which chop image the rewind center lives in [ENUM: (0: off) (1: midpoint) (2: plus_beam) (3: minus_beam)]	INT4	N/A			SET
los	TARF->IRF coordinate to rewind to, (Default Type: CALCULATED, Value:)	FLOAT8	degrees	-3	3	SET
los_margin	Buffer between soft and hard LOS limits, (Default Type: SETTABLE, Value: ta_pos.rewind.los_margin)	FLOAT8	degrees	0	2.8	SET
ta_pos.set	Set the value for a mode or parameter					
ANY	The keyword of this argument is the name of the item or array to set; the value supplied in this argument is the new value for the item or specified array element.	STRING	N/A			INDEXED
ta_pos.set_archive_nsample	Set the archive nsample for a data group					
datagroup	Data group name, (Precision: N/A) , (Accuracy: N/A) , [ENUM: (1: time_until_rewind)]	STRING	none	1	13	NONE
nsample	archive n sample value, (Precision: N/A) , (Accuracy: N/A) , (Default Type: FIXED, Value: 1)	UINT4	none	0	4294967295	NONE
ta_pos.stop	Command the telescope to halt any movement					
ta_pos.track	Tell the TA to track (and how)					
centroid	AOI to guide by, (Default Type: CALCULATED, Value:)	STRING	N/A			POSITION
inertial	Deprecated - Track inertially (or not)?, (Default Type: FIXED, Value:yes) , [ENUM: (1: yes) (0: no)]	BOOL4	N/A			NONE
rof1	AOI to guide by (for ROF calculations), (Default Type: CALCULATED, Value:)	STRING	N/A			POSITION
rof2	AOI to guide by (for ROF calculations), (Default Type: CALCULATED, Value:)	STRING	N/A			POSITION
track_mode	Enable tracking?, (Default Type: CALCULATED, Value:) , [ENUM: (0: none) (0: off)]	STRING	N/A			NONE
ta_pos.trc_boresight	Set the trc boresight to the sibs position					
beam_id	Defines the beam for the TRC to use in its boresight, (Precision: N/A) , (Accuracy: N/A) , [ENUM: (0: OFF) (1: MIDPOINT) (2: PLUS_BEAM) (3: MINUS_BEAM)]	INT4	N/A	0	3	NONE

Command	Command Description					
Argument	**Argument Description**	**Representation**	**Units**	**Low Limit**	**High Limit**	**Value Type**
ta_pos.tweak	Tweak the telescope position					
x	+1, 0, or -1 depending on button push, (Default Type: FIXED, Value: 0)	INT4	unity			NONE
y	+1, 0, or -1 depending on button push, (Default Type: FIXED, Value: 0)	INT4	unity			NONE
ta_pos.tweak_define	Set the current tweaker parameters					
coord_sys	Reference frame or coordinate system, (Default Type: SETTABLE_REQUIRED_ONCE, Value: ta_pos.tweak.sys) , [ENUM: (2: ERF) (3: SIRF) (4: FPIRF) (5: WFIRF) (6: FFIRF) (7: ECRF) (8: GALRF)]	STRING	N/A			NONE
mode	Type of motion, (Default Type: SETTABLE_REQUIRED_ONCE, Value: ta_pos.tweak.mode) , [ENUM: *(1: scan)* (2: offset) (4: step)] NOTE: "offset" is deprecated and is same as "step"	STRING	N/A			NONE
x	Horizontal motion parameter, (Default Type: CALCULATED, Value:)	FLOAT8	appropriate coordinate units			SET_NO_CONVERT
y	Vertical motion parameter, (Default Type: CALCULATED, Value:)	FLOAT8	appropriate coordinate units			SET_NO_CONVERT
ta_pos.verbose	Enables debugging output					
ta_state.camera	Privileged command to set up WFI, FFI, or FPI					
adcu	Number of bits in the output data, (Default Type: CALCULATED, Value: ta_state.camera.*.adcu) , [ENUM: (8: 8) (14: 14) (16:16)]	STRING	N/A	8	16	NONE
binning	Binning factor for camera CCD array, (Default Type: CALCULATED, Value: ta_state.camera.*.binning) , [ENUM: (1: 1x1) (1: 1) (2: 2x2) (2: 2) (4: 4x4) (4: 4)]	STRING	unity	1	4	NONE
camera	Name of the camera to be initialized, [ENUM: (1: wfi) (2: ffi) (4: fpi)]	STRING	N/A	1	4	NONE
integration	Integration time for each camera image, (Precision: 0.001) , (Accuracy: 0.001) , (Default Type: CALCULATED_REQUIRED_ONCE, Value: ta_state.camera.*.integration)	FLOAT8	seconds	0.01	10	SET
ta_wfi.debug	Outputs software debug information					
ta_wfi.image_to_hk	Causes the next received image to be sent to HK data					
comment	If the format type is fits, then the contents of this argument will be added as a comment to the FITS header	STRING	N/A			NONE
type	The format of the HK data (binary or fits), [ENUM: (0: binary) (1: fits)]	STRING	N/A			NONE
ta_wfi.log	Causes the next received image to be written to file					
comment	If the type of the file is fits, then the contents of this argument will be added as a comment to the FITS header	STRING	N/A			NONE
file	The file to write the image data to, do not add the file extension.	STRING	N/A			NONE
n_times	Number of image files to write.	INT4	N/A	1	100	NONE
type	The format of the file data (binary or fits), [ENUM: (0: binary) (1: fits)]	STRING	N/A			NONE

Command	Command Description					
Argument	**Argument Description**	**Representation**	**Units**	**Low Limit**	**High Limit**	**Value Type**
ta_wfi.retransmit_packets	Indicates whether or not to re-transmit packets received from the imagers to the PI Patch Panel (system default is not to re-transmit the packets). The user may also specify a host and a port to transmit to- if a host and port are not specified, then the default host and port are used.					
host	The domain name or IP address of the host to re-transmit received imager packets to	STRING	N/A			NONE
port	The port number to re-transmit received imager packets to	UINT2	N/A			NONE
state	A value of true causes packets to be re-transmitted to the PI Patch Panel while a value of false causes packets not to be re-transmitted, [ENUM: (0: no) (1: yes)]	BOOL4	N/A			NONE
ta_wfi.show_test_images	Indicates whether or not to process test images sent from the TA					
state	A value of true causes test images to be processed while a value of false causes test images to be ignored, [ENUM: (0: no) (1: yes)]	BOOL4	N/A			NONE
wvm_if.calibrate_los	Commands the WVM to calibrate its line-of-sight with respect to the aircraft's roll and pitch angle					
wvm_if.exit	Commands the WVM application to terminate					
wvm_if.go_to_operate	Commands the WVM to go to the Operate state					
wvm_if.go_to_standby	Commands the WVM to go to the Standby state					
wvm_if.quit	Commands the WVM application to terminate					
wvm_if.run_bit_tests	Commands the WVM to run its internal built-in-tests					
wvm_if.set	Sets a WVM data item. Works similarly to the Session set command					
ANY	Specifies a WVM data item and the value to set it to. The WVM data items are listed in the wvm_if_data.xml file	STRING	none			SET
wvm_if.status	Commands the WVM to report its present state					

APPENDIX B. HOUSEKEEPING DATA

This appendix describes MCCS housekeeping data. It contains four sub-appendices:

- Appendix B.1 provides an overview of the naming convention by MCCS subsystem.
- Appendix B.2 describes data latency classification.
- Appendix B.3 describes the data rates required of periodic data.
- Appendix B.4 lists all housekeeping data (with the exception of TA pass-through data which is listed in TA_MCCS_F).

APPENDIX B.1. DATA ORGANIZATION

This section describes SCL housekeeping data. Table B-1 below provides an overview of the housekeeping data by subsystem, while Table B-2 (in Appendix B.4) lists available housekeeping data.

In addition to the housekeeping data items, some items such as ‘mission_id’ and ‘time’ are generated when requested by a user (e.g. with an SCL ‘get’ command) – these are described in Section 4.2.3.

Note that references to ‘<session>’ in Tables B-1 and B-2 refer to the ‘session_name’ value described in Table 19 in Section 4.2.3.

References to ‘<connection>’ in Tables B-1 and B-2 refer to the names of MCCS connections from the following list:

- cdds.serial
- das.cmd_socket
- das.hk_socket
- ta.cmd_socket
- ta.hk_socket
- ta_ffi.socket
- ta_fpi.socket
- ta_wfi.socket
- wvm_if.wvm_manager

For example, ‘cdds.serial.total_bytes_read’ is the name of the housekeeping value telling how many bytes have been read by the PIS from the CDDS.

All housekeeping data is saved to the MCCS Archiver for in-flight or post-flight access with the exception of:

- <connection> data listed above
- <session> data
- *.archive_info.current_bytes_archived
- *.archive_info.total_bytes_archived.

Table B-1: Housekeeping Data Subsystems

Subsystem	Description
ars	Archiver
cdds	Cavity Door Drive System (CDDS)
<connection>	Generic connection data produced by proxies
coord	Coordinate-related data
das	Data Acquisition Subsystem (DAS)
dither	Dither-related data
fltexec	Flight Executor (fltexec_data provided by FMI flight planning application)
nod	Nod-related data
ntp	NTP server Housekeeping Data
rien	Template for Science Instrument (SI) data ("rien" is a fictitious SI name)
<session>	User session data logged by default
sma	Secondary Mirror Assembly (SMA)
ta	Generic Telescope Assembly (TA) data (not subsystem specific)
ta_ffi	Fine Field Imager (FFI)
ta_fpi	Focal Plane Imager (FPI)
ta_mcp	TA Master Control Processor (MCP)
ta_pos	TA position-related data
ta_scs	TA Secondary Mirror Assembly Controller System (SCS)
ta_state	TA state-related data
ta_trc	TA Tracker Controller
ta_tsc	TA Servo Control Unit (TASCU)
ta_wfi	Wide Field Imager (WFI)
wvm_if	Water Vapor Monitor (WVM)

APPENDIX B.2. DATA LATENCY

Housekeeping data is classified according to its latency requirements. Currently there are two classes: Class 1 and Class U; further numbered classes (beyond class 1) are reserved for future classification of observatory Housekeeping Data.

Class 1:
Housekeeping parameters that require timely delivery to subscribers because the data is used in real time calculations or for real time decisions.

Class U:
Housekeeping parameters that are not used in real time calculations but that are useful for in-flight analysis, post-flight analysis, or situational awareness.

Class 1 Housekeeping Data Consists of:

- All TA Housekeeping Parameters
  - ta_mcp.*, ta_scs.*, ta_trc.*, ta_tsc.* (described in TA_MCCS_F)
  - ta.*, ta_fpi.*, ta_ffi.*, ta_wfi.*
- All Housekeeping Parameters from xform
  - coord.*
  - dither.*
  - nod.*
  - sma.*
  - ta_pos.*
  - ta_state.*
  - <rien>.*
- All Housekeeping Parameters from CDDS
  - cdds.*
- The Following Housekeeping Parameters from DAS (TBR)
  - das.ic1080_2hz.lat_fms_1
  - das.ic1080_2hz.lat_fms_2
  - das.ic1080_2hz.lon_fms_1
  - das.ic1080_2hz.lon_fms_2
  - das.ic1080_2hz.true_heading
  - das.ic1080_5hz.roll
  - das.ic1080_2hz.ground_speed
  - das.ic1080_10hz.pitch
  - das.ic1080_2hz.true_track_angle
  - das.gps_1_10hz.gps_msl_alt
  - das.gps_2_10hz.gps_msl_alt

Class U Housekeeping Data Consists of:

- All Housekeeping Parameters from Archiver
  - ars.*

- All Housekeeping Parameters from NTP Server
  - ntp.*
- All Housekeeping Parameters from WVM
  - wvm_if.*
- All Housekeeping Parameters from DAS which are not included in the Class 1 list.
  - das.*

APPENDIX B.3. PERIODIC DATA RATE

The table below defines the rates of periodic distribution and archiving of housekeeping data that MCCS is required to support in the final delivered system. Current rates for periodic data, where known, are listed in the last column of the table in the next section, B.4.

Rates stated in the table are minimums – in other words MCCS may support rates faster than those stated in the table. In many cases, actual data rates from external subsystems (such as the TA) are configurable, and therefore, at the choice of observatory users, data may be distributed and archived at rates slower than those listed in the table. The stated rates therefore should be interpreted as rates that MCCS will be capable of supporting, not of the rates that will be used at all times.

Housekeeping data is conveyed within MCCS in data blocks called data groups. Therefore the data rate requirements for distribution and archiving are levied by data group. The table organizes all periodic housekeeping data into commandables which are further broken down into data groups.

The first two columns in the table identify the commandable and data group. The third column defines the required rate for real time distribution of data contained in that data group. The forth column defines the required rate for archiving the data group. The entries in the fifth column describe the rationale for the rate requirements for data distribution and archiving. Many of the entries in the "Rationale" column are references to a rationale statement contained in the table at the end of this section.

Commandable	Data Group(s)	Real Time Rate (Hz)	Archiving Rate (Hz)	Rationale for Data Rates
ta_mcp	mcp_hk_anu	10	10	***Rationale for Real Time Data Rate:*** See Rationale Note 1 ***Rationale for Archiving Rate:*** See Rationale Note 2
ta_mcp	mcp_hk_states	5	10	***Rationale for Real Time Data Rate:*** See Rationale Note 1 ***Rationale for Archiving Rate:*** See Rationale Note 2
ta_mcp	mcp_hk_pdu	2	10	***Rationale for Real Time Data Rate:*** See Rationale Note 1 ***Rationale for Archiving Rate:*** See Rationale Note 2
ta_mcp	mcp_hk_pms	2	10	***Rationale for Real Time Data Rate:*** Manage focus scenario uses temperature; presumably these do not change rapidly ***Rationale for Archiving Rate:*** See Rationale Note 2
ta_mcp	mcp_hk_pwc	2	10	***Rationale for Real Time Data Rate:*** See Rationale Note 1 ***Rationale for Archiving Rate:*** See Rationale Note 2

Commandable	Data Group(s)	Real Time Rate (Hz)	Archiving Rate (Hz)	Rationale for Data Rates
ta_tsc	tsc_mcs_high_rate_hk	50	50	***Rationale for Real Time Data Rate*:** This data group is used to deliver 20ms-segments of a waveform. That is, each data record contains several data points that make up a 20ms-segment of a waveform. So in order to display an unbroken waveform on the GUI, all of the packets must be delivered. (0.02 seconds -> 50Hz) ***Rationale for Archiving Rate:*** The full archiving rate allows the waveform to be completely reconstructed from stored data.
ta_tsc	tsc_mcs_hk	10	50	***Rationale for Real Time Data Rate*:** Scenarios 0, 1, 3 - indicates desired attitude (and cmded and actual) used for accurate moves ***Rationale for Archiving Rate:*** See Rationale Note 2
ta_scs	tcm_process_status	50	50	***Rationale for Real Time Data Rate*:** This data group is used to deliver 20ms-segments of a waveform. That is, each data record contains several data points that make up a 20ms-segment of a waveform. So in order to display an unbroken waveform on the GUI, all of the packets must be delivered. (0.02 seconds -> 50Hz) ***Rationale for Archiving Rate:*** The full archiving rate allows the waveform to be completely reconstructed from stored data.
ta_scs	scs_status	4	10	***Rationale for Real Time Data Rate*:** Scenario 0 and 5; XFORMS may not need faster than this ***Rationale for Archiving Rate:*** See Rationale Note 2
ta_scs	fcm_status	4	10	***Rationale for Real Time Data Rate*:** Scenario 0 and 5; XFORMS may not need faster than this ***Rationale for Archiving Rate:*** See Rationale Note 2
ta_scs	tcm_status	10	50	***Rationale for Real Time Data Rate*:** Scenario 0 and 5; computation of chop offsets for commanding ***Rationale for Archiving Rate:*** See Rationale Note 2
ta_trc	trc_watchdog	2	2	See Rationale Note 3
ta_trc	trc_status_table	10	10	***Rationale for Real Time Data Rate*:** Scenario 3, indicates completion of TRC_INERTIAL_SET command (during moves with same AOI), indicates track state ***Rationale for Archiving Rate:*** See Rationale Note 2
ta_trc	trc_main_aoi_table	10	10	***Rationale for Real Time Data Rate*:** Scenario 3, AOI IRF axis data ***Rationale for Archiving Rate:*** See Rationale Note 2
ta_trc	trc_centroid_table	10	10	***Rationale for Real Time Data Rate*:** Scen3; indicates when TRC_INERTIAL_SET exception ends (used when moving while tracking; i.e. new position AOIs are same as existing) ***Rationale for Archiving Rate:*** See Rationale Note 2

Commandable	Data Group(s)	Real Time Rate (Hz)	Archiving Rate (Hz)	Rationale for Data Rates
ta_trc	trc_limb_table	10	10	***Rationale for Real Time Data Rate*:** If limb tracking is ever enabled, XFORMS could have similar dependencies on status for Scenario 3 ***Rationale for Archiving Rate:*** See Rationale Note 2
ta_trc	trc_distribution_table	2	10	***Rationale for Real Time Data Rate:*** See Rationale Note 1 ***Rationale for Archiving Rate:*** See Rationale Note 2
ta_trc	trc_boresight_table	2	10	***Rationale for Real Time Data Rate*:** Scenario 1, distortion parameters - used in real-time step but should not change thru flight unless DSI is debugging; Scenario 3 for trc_inertial_set use upon command ***Rationale for Archiving Rate:*** See Rationale Note 2
ta_trc	trc_tracking_table	10	10	***Rationale for Real Time Data Rate*:** Scenario1; computation of SIRF coordinates of tracking position Scenario 3; check for completion of tracking loop to finish command ***Rationale for Archiving Rate:*** See Rationale Note 2
ta_trc	trc_rof_table	10	10	***Rationale for Real Time Data Rate*:** Scenario 3; check for completion of tracking loop to finish command ***Rationale for Archiving Rate:*** See Rationale Note 2
ta_trc	trc_tascu_comm_table	2	10	***Rationale for Real Time Data Rate:*** See Rationale Note 1 ***Rationale for Archiving Rate:*** See Rationale Note 2
ta_trc	wfi_img_hist_table	2	10	***Rationale for Real Time Data Rate:*** See Rationale Note 1 ***Rationale for Archiving Rate:*** See Rationale Note 2
ta_trc	ffi_img_hist_table	2	10	***Rationale for Real Time Data Rate:*** See Rationale Note 1 ***Rationale for Archiving Rate:*** See Rationale Note 2
ta_trc	fpi_img_hist_table	2	10	***Rationale for Real Time Data Rate:*** See Rationale Note 1 ***Rationale for Archiving Rate:*** See Rationale Note 2
ta_trc	trc_temperature_table	2	10	***Rationale for Real Time Data Rate:*** See Rationale Note 1 ***Rationale for Archiving Rate:*** See Rationale Note 2
ta_trc	trc_alignment_table	2	10	***Rationale for Real Time Data Rate*:** Scenario 0, imager orientation - used in real-time step but should not change thru flight unless DSI is debugging ***Rationale for Archiving Rate:*** See Rationale Note 2
ta_trc	wfi_control_table	2	10	***Rationale for Real Time Data Rate:*** See Rationale Note 1 ***Rationale for Archiving Rate:*** See Rationale Note 2
ta_trc	wfi_aoi_bs_table	1	10	***Rationale for Real Time Data Rate:*** See Rationale Note 1 ***Rationale for Archiving Rate:*** See Rationale Note 2

Commandable	Data Group(s)	Real Time Rate (Hz)	Archiving Rate (Hz)	Rationale for Data Rates
ta_trc	ffi_control_table	2	10	***Rationale for Real Time Data Rate:*** See Rationale Note 1 ***Rationale for Archiving Rate:*** See Rationale Note 2
ta_trc	ffi_aoi_bs_table	1	10	***Rationale for Real Time Data Rate:*** See Rationale Note 1 ***Rationale for Archiving Rate:*** See Rationale Note 2
ta_trc	fpi_control_table	2	10	***Rationale for Real Time Data Rate:*** See Rationale Note 1 ***Rationale for Archiving Rate:*** See Rationale Note 2
ta_trc	fpi_aoi_bs_table	1	10	***Rationale for Real Time Data Rate:*** See Rationale Note 1 ***Rationale for Archiving Rate:*** See Rationale Note 2
das	ac_engine	1	1	See Rationale Note 3
das	ac_fuel	1	1	See Rationale Note 3
das	Two data groups: adc_[12]_15hz	2	15	See Rationale Note 6
das	Two data groups: adc_[12]_8hz	2	8	See Rationale Note 6
das	Two data groups: adc_[12]_2hz	2	2	See Rationale Note 3
das	Two data groups: cabin_env cavity_env	1	1	See Rationale Note 3
das	dss	2	50	See Rationale Note 6
das	Three data groups: fms_[123]_10hz	2	10	See Rationale Note 6
das	Two data groups: gps_[12]_10hz	2	10	See Rationale Note 6
das	hall_effect	50	50	See Rationale Note 6
das	hepa_filter	2	50	See Rationale Note 6
das	Three data groups: ins_[123]_1hz	1	1	See Rationale Note 3
das	Three data groups: ins_[123]_12hz	2	12	See Rationale Note 6
das	Three data groups: ins_[123]_25hz	2	25	See Rationale Note 6
das	Three data groups: ins_[123]_50hz	2	50	See Rationale Note 6
das	nasmyth_env	1	1	See Rationale Note 3
das	nasmyth_micro	50	50	See Rationale Note 6
das	pds	1	1	See Rationale Note 3
das	si_flange_acc	50	50	See Rationale Note 6
das	si_patch	50	50	See Rationale Note 6
das	telescope_env	1	1	See Rationale Note 3
das	vacuum_pump	1	1	See Rationale Note 3
das	Two data groups: ic1080_15hz, ic1080_2_15hz	2	15	See Rationale Note 6

Commandable	Data Group(s)	Real Time Rate (Hz)	Archiving Rate (Hz)	Rationale for Data Rates
das	Two data groups: ic1080_10hz, ic1080_2_10hz	2	10	See Rationale Note 6
das	Two data groups: ic1080_5hz, ic1080_2_5hz	2	5	See Rationale Note 6
das	Two data groups: ic1080_2hz, ic1080_2_2hz	2	2	See Rationale Note 3
das	Two data groups: ic1080_1hz, ic1080_2_1hz	1	1	See Rationale Note 3
das	health_status	1	1	See Rationale Note 3
cdds	Three data groups: lfd_position lfd_velocity lfd_load	2	20	***Rationale for Real Time Data Rate***: The data distributed in real-time is intended for a user interface. A requirement of 2Hz maximizes situation awareness. ***Rationale for Archiving Rate:*** See Rationale Note 4
cdds	Three data groups: urd_position urd_velocity urd_load	2	20	***Rationale for Real Time Data Rate***: The data distributed in real time is intended for a human-readable interface. 2Hz maximizes situation awareness. ***Rationale for Archiving Rate:*** See Rationale Note 4
cdds	urd_seal_status	1	1	See Rationale Note 3
cdds	aperture_limit_switches	1	1	See Rationale Note 3
cdds	ifd_console_switch_setting	1	1	See Rationale Note 3
cdds	ifd_enable_switch_setting	1	1	See Rationale Note 3
coord	boresight	10	10	See Rationale Note 5
coord	pos	10	10	See Rationale Note 5
coord	sky_los_rate	10	10	See Rationale Note 5
dither	points	10	10	See Rationale Note 5
nod	beams	10	10	See Rationale Note 5
sma	focus_delta	2	2	See Rationale Note 3
sma	focus_fcm_t_calc	2	2	See Rationale Note 3
sma	focus_param_value_t1	2	2	See Rationale Note 3
sma	points	10	10	See Rationale Note 5
sma	sky_amp2	10	10	See Rationale Note 5
sma	sky_amplitude	10	10	See Rationale Note 5
sma	sky_angle	10	10	See Rationale Note 5
sma	sky_coord_sys	10	10	See Rationale Note 5
sma	sky_tilt	10	10	See Rationale Note 5
sma	sky_tip	10	10	See Rationale Note 5
ta_pos	rewind_trigger	10	10	See Rationale Note 5

Commandable	Data Group(s)	Real Time Rate (Hz)	Archiving Rate (Hz)	Rationale for Data Rates
ta_pos	time_until_rewind	10	10	See Rationale Note 5
ta_state	coarse_deg_el	10	10	See Rationale Note 5
ta_state	fd_deg_el	10	10	See Rationale Note 5
ta_state	fd_deg_los	10	10	See Rationale Note 5
ta_state	fd_deg_xel	10	10	See Rationale Note 5
ta_state	irf_deg_el	10	10	See Rationale Note 5
ta_state	irf_deg_los	10	10	See Rationale Note 5
ta_state	irf_deg_xel	10	10	See Rationale Note 5
ta_state	ta_deg_el	10	10	See Rationale Note 5
ta_state	ta_deg_los	10	10	See Rationale Note 5
ta_state	ta_deg_xel	10	10	See Rationale Note 5
ta_state	tsc_status	10	10	See Rationale Note 5
ta_state	vpa	10	10	See Rationale Note 5

Rationale Number	Rationale
Rationale Note 1	***Rationale for Real Time Data Distribution Rate*** This data distribution rate requirement is driven by the TA developers who require these rates for real-time distribution in order to carry out testing, investigations, and trouble shooting.
Rationale Note 2	***Rationale for Archiving Rate:*** These data archiving rate requirements are driven by the TA developers who require these rates for post-operation data analysis in order to carry out investigations and trouble shooting.
Rationale Note 3	***Note:*** This data is generated or sampled at slow enough rates for each data sample to be observed by a human user. For data items that are generated or sampled this slowly, the requirement for both distribution and for archiving is the rate at which the data is generated or sampled. ***Rationale for Real Time Data Rate*:** For real time distribution this maximizes situational awareness. ***Rationale for Archiving Rate:*** For archiving, this provides a full data record of the generated data.
Rationale Note 4	***Rationale for Archiving Rate:*** This archiving rate provides a record of the full data set. The requirement is driven by developers who request that every data sample be recorded for post-operation data analysis for investigations and trouble shooting.
Rationale Note 5	***Note:*** This data is generated from input data which is required to be available at 10Hz. ***Rationale for Real Time Data Rate*:** Calculating and distributing this data at the required input rate will allow the real-time subscribers, such as the science instruments, to receive all of the derived data points. (A higher data rate would result in redundant data. A lower data rate would result in missed data points and wasted input data.) ***Rationale for Archiving Rate:*** A required archiving rate that matches the rate of the input data from which this data is generated will provide a full data record of the results.
Rationale Note 6	***Rationale for Real Time Data Rate and Archiving Rate:*** These data rates were originally determined through a measurement-solicitation process and later vetted with the Science and Platform projects.

APPENDIX B.4. DATA ITEM LIST

The header at the top of each page identifies the associated housekeeping data information according to the following:

- Prefix – prefix "path" of housekeeping item on MCCS data tree
  Note that references to 'ANY' means:
  - user-assigned name of a position or offset (refer to Section 4.1.3 for further information)
  - pre-defined position as described in Table 14, Pre-Defined Entries Supported in SCL
  - predefined MCCS HK item used as a default value as specified in Argument Description column of Appendix A, identified as SETTABLE.
- Value Name – name of housekeeping item
- Description – functional description of housekeeping item
- Representation – how data item is stored in MCCS, as defined in Table 8, Binary Data Types
- Units – default units for housekeeping item, if applicable
- Format – default format for this data item in SCL responses
- Low Limit – low limit for housekeeping item, if applicable, in above-noted units
- High Limit – high limit for housekeeping item, if applicable, in above-noted units
- Data Rate (Hz) – rate in Hz that this value is provided by MCCS. This is the expected rate; the actual rate may differ depending on the current load on MCCS or whether the source of the data is delivering that data as expected. The rates should eventually reach those defined in Appendix B.3 in the final system.

Changes from the previous revision of the ICD are identified in the table below as follows. Unchanged items are in black text. New items are in blue text. Items which have been deleted are in ~~red strike-through text~~. Housekeeping items which may be added in the future are highlighted with a light green background. If users have an upcoming need for such highlighted items, they should advise observatory staff so that their addition may be worked into the observatory schedule.

Table B-2: Housekeeping Data Table.
(starts on next page)

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
ars								
ars	rcs_id	Version information for the XML file that defines this data source	STRING	N/A	%s			event
ars.ars_alarm	ars_alarm_error	Error alert	STRING	N/A	%s			event
ars.ars_alarm	ars_alarm_fatal	Fatal alert	STRING	N/A	%s			event
ars.ars_alarm	ars_alarm_info	Information alarm	STRING	N/A	%s			event
ars.ars_alarm	ars_alarm_warning	Warning alarm	STRING	N/A	%s			event
ars.ars_alert	ars_alert_error	Error alert	STRING	N/A	%s			event
ars.ars_alert	ars_alert_fatal	Fatal alert	STRING	N/A	%s			event
ars.ars_alert	ars_alert_info	Information alert	STRING	N/A	%s			event
ars.ars_alert	ars_alert_warning	Warning alert	STRING	N/A	%s			event
ars.archive_info	archive_file_name	Archive file name	STRING	N/A	%s			event
ars.archive_info	archive_status	[ENUM: (0: false) (1: true)]	BOOL4	N/A	%s	0	1	
ars.archive_info	current_bytes_archived	Total bytes written to the current ark file	FLOAT8	bytes	%.0lf	0		
ars.archive_info	total_bytes_archived	Total bytes written since this data source starts	FLOAT8	bytes	%.0lf	0		
ars.cpu_idle	cpu_idle_all	Percentage of CPU idle averaged over all CPUs	FLOAT4	percent	%.2f			0.11
ars.cpu_idle	cpu0_idle	Percentage of CPU idle on CPU0	FLOAT4	percent	%.2f			0.11
ars.cpu_idle	cpu1_idle	Percentage of CPU idle on CPU1	FLOAT4	percent	%.2f			0.11
ars.cpu_idle	cpu10_idle	Percentage of CPU idle on CPU10	FLOAT4	percent	%.2f			0.11
ars.cpu_idle	cpu11_idle	Percentage of CPU idle on CPU11	FLOAT4	percent	%.2f			0.11
ars.cpu_idle	cpu12_idle	Percentage of CPU idle on CPU12	FLOAT4	percent	%.2f			0.11
ars.cpu_idle	cpu13_idle	Percentage of CPU idle on CPU13	FLOAT4	percent	%.2f			0.11
ars.cpu_idle	cpu14_idle	Percentage of CPU idle on CPU14	FLOAT4	percent	%.2f			0.11
ars.cpu_idle	cpu15_idle	Percentage of CPU idle on CPU15	FLOAT4	percent	%.2f			0.11
ars.cpu_idle	cpu16_idle	Percentage of CPU idle on CPU16	FLOAT4	percent	%.2f			0.11
ars.cpu_idle	cpu17_idle	Percentage of CPU idle on CPU17	FLOAT4	percent	%.2f			0.11
ars.cpu_idle	cpu18_idle	Percentage of CPU idle on CPU18	FLOAT4	percent	%.2f			0.11
ars.cpu_idle	cpu19_idle	Percentage of CPU idle on CPU19	FLOAT4	percent	%.2f			0.11
ars.cpu_idle	cpu2_idle	Percentage of CPU idle on CPU2	FLOAT4	percent	%.2f			0.11
ars.cpu_idle	cpu20_idle	Percentage of CPU idle on CPU20	FLOAT4	percent	%.2f			0.11
ars.cpu_idle	cpu21_idle	Percentage of CPU idle on CPU21	FLOAT4	percent	%.2f			0.11
ars.cpu_idle	cpu22_idle	Percentage of CPU idle on CPU22	FLOAT4	percent	%.2f			0.11
ars.cpu_idle	cpu23_idle	Percentage of CPU idle on CPU23	FLOAT4	percent	%.2f			0.11
ars.cpu_idle	cpu3_idle	Percentage of CPU idle on CPU3	FLOAT4	percent	%.2f			0.11
ars.cpu_idle	cpu4_idle	Percentage of CPU idle on CPU4	FLOAT4	percent	%.2f			0.11
ars.cpu_idle	cpu5_idle	Percentage of CPU idle on CPU5	FLOAT4	percent	%.2f			0.11
ars.cpu_idle	cpu6_idle	Percentage of CPU idle on CPU6	FLOAT4	percent	%.2f			0.11
ars.cpu_idle	cpu7_idle	Percentage of CPU idle on CPU7	FLOAT4	percent	%.2f			0.11
ars.cpu_idle	cpu8_idle	Percentage of CPU idle on CPU8	FLOAT4	percent	%.2f			0.11
ars.cpu_idle	cpu9_idle	Percentage of CPU idle on CPU9	FLOAT4	percent	%.2f			0.11
ars.fs_usage	removable_usage	Percentage of removable drives used.	UINT4	percent	%d			0.11
ars.fs_usage	root_usage	Percentage of root used.	UINT4	percent	%d			0.11
ars.memory	mem_free	Total amount of memory(RAM) free	FLOAT8	bytes	%lf			0.11
ars.memory	mem_total	Total amount of memory(RAM) in the system	FLOAT8	bytes	%lf			0.11
ars.quota	das_quota_hard	Quota limit for DAS	FLOAT8	bytes	%lf			
ars.quota	das_quota_used	Quota usage by DAS	FLOAT8	bytes	%lf			
ars.quota	hk_quota_hard	Quota limit for HK	FLOAT8	bytes	%lf			0.11
ars.quota	hk_quota_used	Quota usage by HK	FLOAT8	bytes	%lf			0.11
ars.quota	mads_quota_hard	Quota limit for MADS	FLOAT8	bytes	%lf			
ars.quota	mads_quota_used	Quota usage by MADS	FLOAT8	bytes	%lf			
ars.quota	si_quota_hard	Quota limit for SI	FLOAT8	bytes	%lf			
ars.quota	si_quota_used	Quota usage by SI	FLOAT8	bytes	%lf			
ars.quota	taips_quota_hard	Quota limit for TAIPS	FLOAT8	bytes	%lf			

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
ars.quota	taips_quota_used	Quota usage by TAIPS	FLOAT8	bytes	%lf			
ars.quota	vpars_quota_hard	Quota limit for VPARS	FLOAT8	bytes	%lf			
ars.quota	vpars_quota_used	Quota usage by VPARS	FLOAT8	bytes	%lf			
ars.quota	wvm_quota_hard	Quota limit for WVM	FLOAT8	bytes	%lf			
ars.quota	wvm_quota_used	Quota usage by WVM	FLOAT8	bytes	%lf			
cdds								
cdds	ac_power_event	AC Power Event, [ENUM: (1: POWER_UP) (2: UNDER_VOLTAGE) (3: UNDER_VOLTAGE_50MS) (4: UNDER_VOLTAGE_200MS)]	UINT4	N/A	%d	1	4	
cdds	commanded_setpoint	LFD setpoint sent to the CDDS	FLOAT8	degrees	%.2f	23.15	58.20	
cdds	dc_power_event	DC Power Event, [ENUM: (1: POWER_UP) (2: UNDER_VOLTAGE) (3: UNDER_VOLTAGE_50MS) (4: UNDER_VOLTAGE_200MS)]	UINT4	N/A	%d	1	4	
cdds	ifd_console_switch_setting	CDDS opinion of IFD console switch, [ENUM: (0: INDETERMINATE) (1: RESET) (2: CLOSE_PARK) (4: OPEN) (8: TRACK)]	UINT4	N/A	%d	0	8	1
cdds	ifd_enable_switch_setting	CDDS opinion of IFD enable switch, [ENUM: (1: ENABLE) (2: DISABLE)]	UINT4	N/A	%d	1	2	1
cdds	invalid_transition	Invalid transition detected	UINT4	N/A	%d			
cdds	lfd_end_of_travel	LFD end of travel flag, [ENUM: (0: CLOSED_LIMIT) (1: OPEN_LIMIT)]	UINT4	N/A	%d	0	1	
cdds	lfd_load	Dynamic load of lower flexible door	FLOAT8	pounds	%.0f	-10000	10000	20
cdds	lfd_over_speed	Excessive LFD speed measured	FLOAT8	degrees per second	%.2f	-10	10	
cdds	lfd_over_torque	Excessive LFD torque measured	FLOAT8	pounds	%.0f	-10000	10000	
cdds	lfd_position	Position of lower flexible door	FLOAT8	degrees	%.2f	19	61	20
cdds	lfd_velocity	Velocity of lower flexible door	FLOAT8	degrees per second	%.2f	-10	10	20
cdds	ok_to_close_urd_flag	After a system fault, flag that says if the URD can be closed, [ENUM: (0: NOT_OK_TO_CLOSE) (1: OK_TO_CLOSE)]	UINT4	N/A	%d	0	1	
cdds	rcs_id	Version information for the XML file that defines this data source	STRING	N/A	%s			event
cdds	setpoint_error	Erroneous setpoint sent from MCCS	FLOAT8	degrees	%.2f			
cdds	setpoint_latency	Time difference between TASCU ingestion into MCCS and setpoint generation	FLOAT8	seconds	%lf			
cdds	startup_bit_failure	Startup BIT failure, [ENUM: (1: SELF_TEST_PROCESSOR_INSTRUCTIONS) (2: RAM) (3: NON_VOLATILE_MEMORY) (4: EXTERNAL_WATCHDOG) (5: ROM) (6: PREVIOUS) (7: ADJUSTABLE_PARAMETERS) (8: URD_UPPER_END_OF_TRAVEL_1) (9: URD_UPPER_END_OF_TRAVEL_2) (10: URD_LOWER_END_OF_TRAVEL_1) (11: URD_LOWER_END_OF_TRAVEL_2) (12: LFD_UPPER_END_OF_TRAVEL_1) (13: LFD_UPPER_END_OF_TRAVEL_2) (14: LFD_LOWER_END_OF_TRAVEL_1) (15: LFD_LOWER_END_OF_TRAVEL_2) (16: GAP_MONITOR_LATENT_1) (17: GAP_MONITOR_LATENT_2) (18: URD_UPPER_TORQUE_LIMIT_OVERTORQUE_MONITOR_A1) (19: URD_LOWER_TORQUE_LIMIT_OVERTORQUE_MONITOR_A1) (20: URD_UPPER_TORQUE_LIMIT_OVERTORQUE_MONITOR_A2) (21: URD_LOWER_TORQUE_LIMIT_OVERTORQUE_MONITOR_A2) (22: URD_UPPER_TORQUE_LIMIT_OVERTORQUE_MONITOR_B1) (23: URD_LOWER_TORQUE_LIMIT_OVERTORQUE_MONITOR_B1) (24: URD_UPPER_TORQUE_LIMIT_OVERTORQUE_MONITOR_B2) (25: URD_LOWER_TORQUE_LIMIT_OVERTORQUE_MONITOR_B2) (26: LFD_UPPER_TORQUE_LIMIT_OVERTORQUE_MONITOR_A1) (27: LFD_LOWER_TORQUE_LIMIT_OVERTORQUE_MONITOR_A1) (28: LFD_UPPER_TORQUE_LIMIT_OVERTORQUE_MONITOR_A2) (29: LFD_LOWER_TORQUE_LIMIT_OVERTORQUE_MONITOR_A2) (30: LFD_UPPER_TORQUE_LIMIT_OVERTORQUE_MONITOR_B1) (31: LFD_LOWER_TORQUE_LIMIT_OVERTORQUE_MONITOR_B1) (32: LFD_UPPER_TORQUE_LIMIT_OVERTORQUE_MONITOR_B2) (33: LFD_LOWER_TORQUE_LIMIT_OVERTORQUE_MONITOR_B2) (34: URD_POSITIVE_OVERSPEED_MONITOR_1) (35: URD_NEGATIVE_OVERSPEED_MONITOR_1) (36: URD_POSITIVE_OVERSPEED_MONITOR_2) (37: URD_NEGATIVE_OVERSPEED_MONITOR_2) (38: LFD_POSITIVE_OVERSPEED_MONITOR_1) (39: LFD_NEGATIVE_OVERSPEED_MONITOR_1) (40: LFD_POSITIVE_OVERSPEED_MONITOR_2) (41: LFD_NEGATIVE_OVERSPEED_MONITOR_2) (42: SHOP_MODE_ENABLED) (43: RS_422_LOOPBACK) (44: TORQUE_LOOPBACK)]	UINT4	N/A		1	44	
cdds	timeout_rate	Number of CDDS timeout complaints per second	FLOAT8	Hz	%lf			
cdds	urd_end_of_travel	URD end of travel flag, [ENUM: (0: CLOSED_LIMIT) (1: OPEN_LIMIT)]	UINT4	N/A	%d	0	1	
cdds	urd_init_open_failure	URD initial opening sequence failure, [ENUM: (0: OPENED) (1: NOT_OPENED)]	UINT4	N/A	%d	0	1	
cdds	urd_lfd_gap_flag	Flag for gap between URD and LFD, [ENUM: (0: NO_GAP) (1: GAP_CREATED)]	UINT4	N/A	%d	0	1	

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
cdds	urd_load	Dynamic load of upper rigid door	FLOAT8	pounds	%.0f	-10000	10000	20
cdds	urd_over_speed	Excessive URD speed measured	FLOAT8	degrees per second	%.2f	-10	10	
cdds	urd_over_torque	Excessive URD torque measured	FLOAT8	pounds	%.0f	-10000	10000	
cdds	urd_position	Position of upper rigid door	FLOAT8	degrees	%.2f	-10	90	20
cdds	urd_seal_pressure	URD seal pressure within the seal	FLOAT8	pounds per square inch	%.2f	0	50	
cdds	urd_seal_status	Status of URD seal, [ENUM: (0: SEALED) (1: UNSEALED)]	UINT4	N/A	%d	0	1	2
cdds	urd_seal_stepped_down_pressure	URD seal stepped down pressure	FLOAT8	pounds per square inch	%.0f	0	100	
cdds	urd_seal_supply_pressure	URD seal supply pressure	FLOAT8	pounds per square inch	%.0f	0	4000	
cdds	urd_seal_vacuum_status	Status of URD seal vacuum pump, [ENUM: (0: RUNNING) (1: STOPPED)]	UINT4	N/A	%d	0	1	
cdds	urd_seal_vent_pressure	URD seal vent line pressure	FLOAT8	pounds per square inch	%.2f	0	50	
cdds	urd_velocity	Velocity of upper rigid door	FLOAT8	degrees per second	%.2f	-10	10	20
cdds	watchdog_reset	Watchdog reset fault	UINT4	N/A	%d			
cdds.aperture_limit_switches	aft	Status of aft aperture limit switch, [ENUM: (0: NORMAL) (4: OPEN)]	UINT4	N/A	%d	0	4	1
cdds.aperture_limit_switches	forward	Status of forward aperture limit switch, [ENUM: (0: NORMAL) (1: OPEN)]	UINT4	N/A	%d	0	1	1
cdds.aperture_limit_switches	middle	Status of middle aperture limit switch, [ENUM: (0: NORMAL) (2: OPEN)]	UINT4	N/A	%d	0	2	1
cdds.archive_info	archive_file_name	Archive file name	STRING	N/A	%s			event
cdds.archive_info	archive_status	[ENUM: (0: false) (1: true)]	BOOL4	N/A	%s	0	1	
cdds.archive_info	current_bytes_archived	Total bytes written to the current ark file	FLOAT8	bytes	%.0lf	0		
cdds.archive_info	total_bytes_archived	Total bytes written since this data source starts	FLOAT8	bytes	%.0lf	0		
cdds.archive_nsamples	aperture_limit_switches_nsample		INT4	N/A	%d	0		event
cdds.archive_nsamples	current_bytes_archived_nsample		INT4	N/A	%d	0		event
cdds.archive_nsamples	ifd_console_switch_setting_nsample		INT4	N/A	%d	0		event
cdds.archive_nsamples	ifd_enable_switch_setting_nsample		INT4	N/A	%d	0		event
cdds.archive_nsamples	lfd_load_nsample		INT4	N/A	%d	0		event
cdds.archive_nsamples	lfd_position_nsample		INT4	N/A	%d	0		event
cdds.archive_nsamples	lfd_velocity_nsample		INT4	N/A	%d	0		event
cdds.archive_nsamples	total_bytes_archived_nsample		INT4	N/A	%d	0		event
cdds.archive_nsamples	urd_load_nsample		INT4	N/A	%d	0		event
cdds.archive_nsamples	urd_position_nsample		INT4	N/A	%d	0		event
cdds.archive_nsamples	urd_seal_pressure_nsample		INT4	N/A	%d	0		event
cdds.archive_nsamples	urd_seal_status_nsample		INT4	N/A	%d	0		event
cdds.archive_nsamples	urd_seal_stepped_down_pressure_nsample		INT4	N/A	%d	0		event
cdds.archive_nsamples	urd_seal_supply_pressure_nsample		INT4	N/A	%d	0		event
cdds.archive_nsamples	urd_seal_system_fault_nsample		INT4	N/A	%d	0		event
cdds.archive_nsamples	urd_seal_vacuum_status_nsample		INT4	N/A	%d	0		event
cdds.archive_nsamples	urd_seal_valve_status_nsample		INT4	N/A	%d	0		event
cdds.archive_nsamples	urd_seal_vent_pressure_nsample		INT4	N/A	%d	0		event
cdds.archive_nsamples	urd_velocity_nsample		INT4	N/A	%d	0		event

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
cdds.brake_fault	lfd1	LFD drive #1 brake, [ENUM: (0: PASS) (4: FAIL)]	UINT4	N/A	%d	0	4	
cdds.brake_fault	lfd2	LFD drive #2 brake, [ENUM: (0: PASS) (8: FAIL)]	UINT4	N/A	%d	0	8	
cdds.brake_fault	urd1	URD drive #1 brake, [ENUM: (0: PASS) (1: FAIL)]	UINT4	N/A	%d	0	1	
cdds.brake_fault	urd2	URD drive #2 brake, [ENUM: (0: PASS) (2: FAIL)]	UINT4	N/A	%d	0	2	
cdds.cdds_alarm	cdds_alarm_error	Serious problem requiring user confirmation	STRING	N/A	%s			event
cdds.cdds_alarm	cdds_alarm_fatal	Fatal problem requiring user confirmation	STRING	N/A	%s			event
cdds.cdds_alarm	cdds_alarm_info	Normal event requiring user confirmation	STRING	N/A	%s			event
cdds.cdds_alarm	cdds_alarm_warning	Abnormal event requiring user confirmation	STRING	N/A	%s			event
cdds.cdds_alert	cdds_alert_error	Non-recoverable problem which is not fatal	STRING	N/A	%s			event
cdds.cdds_alert	cdds_alert_fatal	Fatal problem	STRING	N/A	%s			event
cdds.cdds_alert	cdds_alert_info	Informational message (event)	STRING	N/A	%s			event
cdds.cdds_alert	cdds_alert_warning	Warning message (may require user action for recovery)	STRING	N/A	%s			event
cdds.continuous_bit_failure	crc_flight	Status of Flight-Mode CRC BIT test, [ENUM: (0: PASS) (1: FAIL)]	UINT4	N/A	%d	0	1	
cdds.continuous_bit_failure	crc_shop	Status of Shop-Mode CRC BIT test, [ENUM: (0: PASS) (2: FAIL)]	UINT4	N/A	%d	0	2	
cdds.drive_system_jammed	urd_close_sequence	URD closing sequence, [ENUM: (0: OK) (2: JAMMED)]	UINT4	N/A	%d	0	2	
cdds.drive_system_jammed	urd_open_sequence	URD opening sequence, [ENUM: (0: OK) (1: JAMMED)]	UINT4	N/A	%d	0	1	
cdds.ifd_console_switch_invalid	close_park	Closed IFD console switch	UINT4	N/A	%d	0	2	
cdds.ifd_console_switch_invalid	indeterminate	Indeterminate IFD console switch	UINT4	N/A	%d	0	15	
cdds.ifd_console_switch_invalid	open	Open IFD console switch	UINT4	N/A	%d	0	4	
cdds.ifd_console_switch_invalid	reset	Reset IFD console switch	UINT4	N/A	%d	0	1	
cdds.ifd_console_switch_invalid	track	Track IFD console switch	UINT4	N/A	%d	0	8	
cdds.loss_of_motion	lfd	LFD loss of motion monitor, [ENUM: (0: PASS) (2: FAIL)]	UINT4	N/A	%d	0	2	
cdds.loss_of_motion	urd	URD loss of motion monitor, [ENUM: (0: PASS) (1: FAIL)]	UINT4	N/A	%d	0	1	
cdds.motor_controller_fault	lfd1	LFD controller #1, [ENUM: (0: PASS) (4: FAIL)]	UINT4	N/A	%d	0	4	
cdds.motor_controller_fault	lfd2	LFD controller #2, [ENUM: (0: PASS) (8: FAIL)]	UINT4	N/A	%d	0	8	
cdds.motor_controller_fault	urd1	URD controller #1, [ENUM: (0: PASS) (1: FAIL)]	UINT4	N/A	%d	0	1	
cdds.motor_controller_fault	urd2	URD controller #2, [ENUM: (0: PASS) (2: FAIL)]	UINT4	N/A	%d	0	2	
cdds.motor_current_monitor	lfd	LFD motor current monitor, [ENUM: (0: PASS) (2: FAIL)]	UINT4	N/A	%d	0	2	
cdds.motor_current_monitor	lfd_secondary	LFD secondary motor current monitor, [ENUM: (0: PASS) (8: FAIL)]	UINT4	N/A	%d	0	8	
cdds.motor_current_monitor	urd	URD motor current monitor, [ENUM: (0: PASS) (1: FAIL)]	UINT4	N/A	%d	0	1	
cdds.motor_current_monitor	urd_secondary	URD secondary motor current monitor, [ENUM: (0: PASS) (4: FAIL)]	UINT4	N/A	%d	0	4	
cdds.motor_enable_logic_fault	lfd1	LFD drive 1 enable fault, [ENUM: (0: PASS) (4: FAULTED)]	UINT4	N/A	%d	0	4	
cdds.motor_enable_logic_fault	lfd2	LFD drive 2 enagble fault, [ENUM: (0: PASS) (8: FAULTED)]	UINT4	N/A	%d	0	8	
cdds.motor_enable_logic_fault	urd1	URD motor 1 enable fault, [ENUM: (0: PASS) (1: FAULTED)]	UINT4	N/A	%d	0	1	
cdds.motor_enable_logic_fault	urd2	URD motor 2 enable faults, [ENUM: (0: PASS) (2: FAULTED)]	UINT4	N/A	%d	0	2	
cdds.motor_speed_monitor	lfd	LFD motor speed monitor, [ENUM: (0: PASS) (2: FAIL)]	UINT4	N/A	%d	0	2	
cdds.motor_speed_monitor	urd	URD motor speed monitor, [ENUM: (0: PASS) (1: FAIL)]	UINT4	N/A	%d	0	1	
cdds.position_monitor	lfd	LFD position monitor, [ENUM: (0: PASS) (2: FAIL)]	UINT4	N/A	%d	0	2	
cdds.position_monitor	urd	URD position monitor, [ENUM: (0: PASS) (1: FAIL)]	UINT4	N/A	%d	0	1	

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
cdds.positioning_mode	mode	Flag for whether we are tracking manually or automatically, [ENUM: (0: MANUAL) (1: AUTOMATIC)]	UINT4	N/A	%d	0	1	
cdds.positioning_mode	offset	Desired distance between the aperture and telescope elevation	FLOAT8	degrees	%.2f			
cdds.power_up	mode	CDDS power-up mode, [ENUM: (17920: FLIGHT_MODE) (21248: SHOP_MODE)]	UINT4	N/A	%d			
cdds.power_up	version	CDDS version	UINT4	N/A	%c			
cdds.setpoint_comm_error	checksum	Checksum error, [ENUM: (0: VALID) (1: INVALID)]	UINT4	N/A	%d	0	1	
cdds.setpoint_comm_error	msgid	Message ID error, [ENUM: (0: VALID) (8: INVALID)]	UINT4	N/A	%d	0	8	
cdds.setpoint_comm_error	parity	Parity error, [ENUM: (0: VALID) (2: INVALID)]	UINT4	N/A	%d	0	2	
cdds.setpoint_comm_error	timeout	Timeout error, [ENUM: (0: NO_TIMEOUT) (4: TIMEOUT)]	UINT4	N/A	%d	0	4	
cdds.uncommanded_motion	lfd	LFD uncommanded motion monitor, [ENUM: (0: PASS) (2: FAIL)]	UINT4	N/A	%d	0	2	
cdds.uncommanded_motion	urd	URD uncommanded motion monitor, [ENUM: (0: PASS) (1: FAIL)]	UINT4	N/A	%d	0	1	
cdds.urd_seal_system_fault	adjustable_param	URD seal adjustable parameter error fault, [ENUM: (0: OK) (1024: WARN)]	UINT4	N/A	%d	0	1024	
cdds.urd_seal_system_fault	inflation_pressure	URD seal inflation pressure fault, [ENUM: (0: OK) (256: FAULT) (512: WARN)]	UINT4	N/A	%d	0	512	
cdds.urd_seal_system_fault	inflation_time	URD seal inflation time fault, [ENUM: (0: OK) (128: WARN)]	UINT4	N/A	%d	0	128	
cdds.urd_seal_system_fault	seal	URD seal fault, [ENUM: (0: OK) (64: FAULT)]	UINT4	N/A	%d	0	64	
cdds.urd_seal_system_fault	source_pressure	URD seal source pressure fault, [ENUM: (0: OK) (1: FAULT) (2: WARN)]	UINT4	N/A	%d	0	2	
cdds.urd_seal_system_fault	stepped_down_pressure	URD seal stepped down pressure fault, [ENUM: (0: OK) (4: FAULT) (8: WARN)]	UINT4	N/A	%d	0	8	
cdds.urd_seal_valve_status	sov1	Status of URD seal solenoid valve #1, [ENUM: (0: OPEN) (1: CLOSED)]	UINT4	N/A	%d	0	1	
cdds.urd_seal_valve_status	sov2	Status of URD seal solenoid valve #2, [ENUM: (0: OPEN) (2: CLOSED)]	UINT4	N/A	%d	0	2	
cdds.urd_seal_valve_status	sov3	Status of URD seal solenoid valve #3, [ENUM: (0: OPEN) (4: CLOSED)]	UINT4	N/A	%d	0	4	
<connection>								
<connection>	connected	Indicates whether or not the connection exists	STRING	N/A	%s			
<connection>	connection_losses	Number of connections that have been lost	FLOAT8	N/A	%.0lf			
<connection>	connection_num	Number of successful connections made	FLOAT8	N/A	%.0lf			
<connection>	current_bytes_read	Number of bytes read on the current connection	FLOAT8	N/A	%.0lf			
<connection>	current_bytes_written	Number of bytes written on the current connection	FLOAT8	N/A	%.0lf			
<connection>	current_reads	Number of reads on the current connection	FLOAT8	N/A	%.0lf			
<connection>	current_writes	Number of writes on the current connection	FLOAT8	N/A	%.0lf			
<connection>	start_connection_time	Time the current connection state began	FLOAT8	seconds	%.6lf			
<connection>	total_bytes_read	Total number of bytes read on all connections	FLOAT8	N/A	%.0lf			
<connection>	total_bytes_written	Total number of bytes written on all connections	FLOAT8	N/A	%.0lf			
<connection>	total_reads	Total number of reads on all connections	FLOAT8	N/A	%.0lf			
<connection>	total_writes	Total number of writes on all connections	FLOAT8	N/A	%.0lf			
coord								
coord	aoa	mean pitch of the aircraft	FLOAT8	degrees	%lf	-90	90	
coord	boresight	Boresight location currently in use	STRING	N/A	%s			
coord	commands	Contains the last SCL command sent to coord	STRING	N/A	%s			event
coord	comment	Coord comment	STRING	N/A	%s			
coord	current_instrument	current SI name	STRING	N/A	%s			
coord	diag_d	Induced offset in previous coord.correct point by new correction - should be small less than 1arcsec. See Scenario 0.	FLOAT8	degrees	%lf	-180	180	
coord	deltaq	Change in VPA by the new correction can be large could be many degrees worst case, probably large fraction of a degree. See Scenario 0.	FLOAT8	degrees	%lf	-180	180	
coord	file	Coord comment file	STRING	N/A	%s			
coord	lst	local sidereal time	FLOAT8	sidereal hours	%.1sgh			
coord	obsstat_md	Observatory status flag used by MD for in-flight communications and to govern Research Hours (RH) metric clock [ENUM: (0: UNKNOWN) (1: OBS_FAULT) (2: TURNING) (3: ON_LEG)]	INT4	N/A	%d	0	3	event

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
coord	obsstat_si	Observatory status flag used by SI for in-flight communications and to govern Data Collection Time (DCT) metric clock [ENUM: (0: UNKNOWN) (1: SI_FAULT) (2: COMPLETED) (3: OBSERVING)]	INT4	N/A	%d	0	3	event
coord	obsstat_to	Observatory status flag used by TO for in-flight communications and to govern Science Flight Hours (SFH) metric clock [ENUM: (0: UNKNOWN) (1: TA_FAULT) (2: CONFIGURING) (3: READY)]	INT4	N/A	%d	0	3	event
coord	oper_state	Operating state	INT4	N/A	0x%2.2X			
coord	q_irf_to_erf	The current IRF to ERF transformation. See Scenario 0.	UQUAT32	N/A	%lf	-1	1	
coord	rcs_id	Version information for the XML file that defines this data source	STRING	N/A	%s			event
coord	session_a_obj_count	object handle count in session_a	INT4	hertz	%d			
coord	session_b_obj_count	object handle count in session_b	INT4	hertz	%d			
coord	sky_los_rate	Sky rotation rate estimated from actual conditions	FLOAT8	degrees per hour	%lf			
coord	vlos	Velocity along the Line Of Sight	FLOAT8	meters per second	%lf	0	99000	
coord	vlsr	Velocity with respect to the Local Standard of Rest	VECTOR24	meters per second	%lf	0	99000	
coord	vlsr_itvl	interval between VLSR/VLOS calculations	FLOAT8	seconds	%lf	0.5	3600	
coord.archive_info	archive_file_name	Archive file name	STRING	N/A	%s			event
coord.archive_info	archive_status	[ENUM: (0: false) (1: true)]	BOOL4	N/A	%s	0	1	
coord.archive_info	current_bytes_archived	Total bytes written to the current ark file	FLOAT8	bytes	%.0lf	0		
coord.archive_info	total_bytes_archived	Total bytes written since this data source starts	FLOAT8	bytes	%.0lf	0		
coord.archive_nsamples	aoa_nsample		INT4	N/A	%d	0		event
coord.archive_nsamples	current_bytes_archived_nsample		INT4	N/A	%d	0		event
~~coord.archive_nsamples~~	~~drift_axis_nsample~~		~~INT4~~	~~N/A~~	~~%d~~	~~0~~		~~event~~
~~coord.archive_nsamples~~	~~drift_rate_nsample~~		~~INT4~~	~~N/A~~	~~%d~~	~~0~~		~~event~~
coord.archive_nsamples	lst_nsample		INT4	N/A	%d	0		event
coord.archive_nsamples	total_bytes_archived_nsample		INT4	N/A	%d	0		event
coord.archive_nsamples.aoi1	erf_nsample		INT4	N/A	%d	0		event
coord.archive_nsamples.aoi1	name_nsample		INT4	N/A	%d	0		event
coord.archive_nsamples.aoi2	erf_nsample		INT4	N/A	%d	0		event
coord.archive_nsamples.aoi2	name_nsample		INT4	N/A	%d	0		event
coord.archive_nsamples.aoi3	erf_nsample		INT4	N/A	%d	0		event
coord.archive_nsamples.aoi3	name_nsample		INT4	N/A	%d	0		event
coord.archive_nsamples.aoi4	erf_nsample		INT4	N/A	%d	0		event
coord.archive_nsamples.aoi4	name_nsample		INT4	N/A	%d	0		event
coord.archive_nsamples.aoi5	erf_nsample		INT4	N/A	%d	0		event
coord.archive_nsamples.aoi5	name_nsample		INT4	N/A	%d	0		event
coord.archive_nsamples.aoi6	erf_nsample		INT4	N/A	%d	0		event
coord.archive_nsamples.aoi6	name_nsample		INT4	N/A	%d	0		event

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
coord.archive_nsamples.aoi7	erf_nsample		INT4	N/A	%d	0		event
coord.archive_nsamples.aoi7	name_nsample		INT4	N/A	%d	0		event
coord.archive_nsamples.aoi8	erf_nsample		INT4	N/A	%d	0		event
coord.archive_nsamples.aoi8	name_nsample		INT4	N/A	%d	0		event
coord.coord_alarm	coord_alarm_error	Error alert	STRING	N/A	%s			event
coord.coord_alarm	coord_alarm_fatal	Fatal alert	STRING	N/A	%s			event
coord.coord_alarm	coord_alarm_info	Information alert	STRING	N/A	%s			event
coord.coord_alarm	coord_alarm_warning	Warning alert	STRING	N/A	%s			event
coord.coord_alert	coord_alert_error	Error alarm	STRING	N/A	%s			event
coord.coord_alert	coord_alert_fatal	Fatal alarm	STRING	N/A	%s			event
coord.coord_alert	coord_alert_info	Information alarm	STRING	N/A	%s			event
coord.coord_alert	coord_alert_warning	Warning alarm	STRING	N/A	%s			event
coord.delete	action	Default action if new argument is wrong, [ENUM: (0: ignore) (1: query)]	STRING	N/A	%s			
coord.non_sidereal.aoi1	ephemeris	Name of ephemeris file, if one has been associated with the current AOI; i.e., this item will contain the name of the ephemeris file if observing via one, else it will be an empty string	STRING	N/A	%s			
coord.non_sidereal.aoi1	prev_time	The previous ephemeris timestamp for the AOI, i.e., the beginning time for the given {ra,dec}_rate	FLOAT8	seconds	%lf			
coord.non_sidereal.aoi1	stepsize	The stepsize to be used for monitoring the non-sidereal velocity of the AOI. This is nominally detected via the AOI's association with the position and its ephemeris file; however, it is limited for performance reasons to a minimum threshold.	FLOAT4	seconds	%f	1.0		
coord.non_sidereal.aoi1_velocity	dec_rate	The commanded rate of change of the Declination for the given AOI	FLOAT4	arcseconds per second	%f			
coord.non_sidereal.aoi1_velocity	ra_rate	The commanded rate of change of the Right Ascension for the given AOI	FLOAT4	arcseconds per second	%f			
coord.non_sidereal.aoi1_velocity	vpa_rate	The commanded rate of change of the Vertical Position Angle for the given AOI	FLOAT4	arcseconds per second	%f			
coord.non_sidereal.aoi2	ephemeris	Name of ephemeris file, if one has been associated with the current AOI; i.e., this item will contain the name of the ephemeris file if observing via one, else it will be an empty string	STRING	N/A	%s			
coord.non_sidereal.aoi2	prev_time	The previous ephemeris timestamp for the AOI, i.e., the beginning time for the given {ra,dec}_rate	FLOAT8	seconds	%lf			
coord.non_sidereal.aoi2	stepsize	The stepsize to be used for monitoring the non-sidereal velocity of the AOI. This is nominally detected via the AOI's association with the position and its ephemeris file; however, it is limited for performance reasons to a minimum threshold.	FLOAT4	seconds	%f	1.0		
coord.non_sidereal.aoi2_velocity	dec_rate	The commanded rate of change of the Declination for the given AOI	FLOAT4	arcseconds per second	%f			
coord.non_sidereal.aoi2_velocity	ra_rate	The commanded rate of change of the Right Ascension for the given AOI	FLOAT4	arcseconds per second	%f			
coord.non_sidereal.aoi2_velocity	vpa_rate	The commanded rate of change of the Vertical Position Angle for the given AOI	FLOAT4	arcseconds per second	%f			
coord.non_sidereal.aoi3	ephemeris	Name of ephemeris file, if one has been associated with the current AOI; i.e., this item will contain the name of the ephemeris file if observing via one, else it will be an empty string	STRING	N/A	%s			
coord.non_sidereal.aoi3	prev_time	The previous ephemeris timestamp for the AOI, i.e., the beginning time for the given {ra,dec}_rate	FLOAT8	seconds	%lf			

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
coord.non_sidereal.aoi3	stepsize	The stepsize to be used for monitoring the non-sidereal velocity of the AOI. This is nominally detected via the AOI's association with the position and its ephemeris file; however, it is limited for performance reasons to a minimum threshold.	FLOAT4	seconds	%f	1.0		
coord.non_sidereal.aoi3_velocity	dec_rate	The commanded rate of change of the Declination for the given AOI	FLOAT4	arcseconds per second	%f			
coord.non_sidereal.aoi3_velocity	ra_rate	The commanded rate of change of the Right Ascension for the given AOI	FLOAT4	arcseconds per second	%f			
coord.non_sidereal.aoi3_velocity	vpa_rate	The commanded rate of change of the Vertical Position Angle for the given AOI	FLOAT4	arcseconds per second	%f			
coord.non_sidereal.aoi4	ephemeris	Name of ephemeris file, if one has been associated with the current AOI; i.e., this item will contain the name of the ephemeris file if observing via one, else it will be an empty string	STRING	N/A	%s			
coord.non_sidereal.aoi4	prev_time	The previous ephemeris timestamp for the AOI, i.e., the beginning time for the given {ra,dec}_rate	FLOAT8	seconds	%lf			
coord.non_sidereal.aoi4	stepsize	The stepsize to be used for monitoring the non-sidereal velocity of the AOI. This is nominally detected via the AOI's association with the position and its ephemeris file; however, it is limited for performance reasons to a minimum threshold.	FLOAT4	seconds	%f	1.0		
coord.non_sidereal.aoi4_velocity	dec_rate	The commanded rate of change of the Declination for the given AOI	FLOAT4	arcseconds per second	%f			
coord.non_sidereal.aoi4_velocity	ra_rate	The commanded rate of change of the Right Ascension for the given AOI	FLOAT4	arcseconds per second	%f			
coord.non_sidereal.aoi4_velocity	vpa_rate	The commanded rate of change of the Vertical Position Angle for the given AOI	FLOAT4	arcseconds per second	%f			
coord.non_sidereal.aoi5	ephemeris	Name of ephemeris file, if one has been associated with the current AOI; i.e., this item will contain the name of the ephemeris file if observing via one, else it will be an empty string	STRING	N/A	%s			
coord.non_sidereal.aoi5	prev_time	The previous ephemeris timestamp for the AOI, i.e., the beginning time for the given {ra,dec}_rate	FLOAT8	seconds	%lf			
coord.non_sidereal.aoi5	stepsize	The stepsize to be used for monitoring the non-sidereal velocity of the AOI. This is nominally detected via the AOI's association with the position and its ephemeris file; however, it is limited for performance reasons to a minimum threshold.	FLOAT4	seconds	%f	1.0		
coord.non_sidereal.aoi5_velocity	dec_rate	The commanded rate of change of the Declination for the given AOI	FLOAT4	arcseconds per second	%f			
coord.non_sidereal.aoi5_velocity	ra_rate	The commanded rate of change of the Right Ascension for the given AOI	FLOAT4	arcseconds per second	%f			
coord.non_sidereal.aoi5_velocity	vpa_rate	The commanded rate of change of the Vertical Position Angle for the given AOI	FLOAT4	arcseconds per second	%f			
coord.non_sidereal.aoi6	ephemeris	Name of ephemeris file, if one has been associated with the current AOI; i.e., this item will contain the name of the ephemeris file if observing via one, else it will be an empty string	STRING	N/A	%s			
coord.non_sidereal.aoi6	prev_time	The previous ephemeris timestamp for the AOI, i.e., the beginning time for the given {ra,dec}_rate	FLOAT8	seconds	%lf			
coord.non_sidereal.aoi6	stepsize	The stepsize to be used for monitoring the non-sidereal velocity of the AOI. This is nominally detected via the AOI's association with the position and its ephemeris file; however, it is limited for performance reasons to a minimum threshold.	FLOAT4	seconds	%f	1.0		
coord.non_sidereal.aoi6_velocity	dec_rate	The commanded rate of change of the Declination for the given AOI	FLOAT4	arcseconds per second	%f			
coord.non_sidereal.aoi6_velocity	ra_rate	The commanded rate of change of the Right Ascension for the given AOI	FLOAT4	arcseconds per second	%f			

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
coord.non_sidereal.aoi6_velocity	vpa_rate	The commanded rate of change of the Vertical Position Angle for the given AOI	FLOAT4	arcseconds per second	%f			
coord.non_sidereal.aoi7	ephemeris	Name of ephemeris file, if one has been associated with the current AOI; i.e., this item will contain the name of the ephemeris file if observing via one, else it will be an empty string	STRING	N/A	%s			
coord.non_sidereal.aoi7	prev_time	The previous ephemeris timestamp for the AOI, i.e., the beginning time for the given {ra,dec}_rate	FLOAT8	seconds	%lf			
coord.non_sidereal.aoi7	stepsize	The stepsize to be used for monitoring the non-sidereal velocity of the AOI. This is nominally detected via the AOI's association with the position and its ephemeris file; however, it is limited for performance reasons to a minimum threshold.	FLOAT4	seconds	%f	1.0		
coord.non_sidereal.aoi7_velocity	dec_rate	The commanded rate of change of the Declination for the given AOI	FLOAT4	arcseconds per second	%f			
coord.non_sidereal.aoi7_velocity	ra_rate	The commanded rate of change of the Right Ascension for the given AOI	FLOAT4	arcseconds per second	%f			
coord.non_sidereal.aoi7_velocity	vpa_rate	The commanded rate of change of the Vertical Position Angle for the given AOI	FLOAT4	arcseconds per second	%f			
coord.non_sidereal.aoi8	ephemeris	Name of ephemeris file, if one has been associated with the current AOI; i.e., this item will contain the name of the ephemeris file if observing via one, else it will be an empty string	STRING	N/A	%s			
coord.non_sidereal.aoi8	prev_time	The previous ephemeris timestamp for the AOI, i.e., the beginning time for the given {ra,dec}_rate	FLOAT8	seconds	%lf			
coord.non_sidereal.aoi8	stepsize	The stepsize to be used for monitoring the non-sidereal velocity of the AOI. This is nominally detected via the AOI's association with the position and its ephemeris file; however, it is limited for performance reasons to a minimum threshold.	FLOAT4	seconds	%f	1.0		
coord.non_sidereal.aoi8_velocity	dec_rate	The commanded rate of change of the Declination for the given AOI	FLOAT4	arcseconds per second	%f			
coord.non_sidereal.aoi8_velocity	ra_rate	The commanded rate of change of the Right Ascension for the given AOI	FLOAT4	arcseconds per second	%f			
coord.non_sidereal.aoi8_velocity	vpa_rate	The commanded rate of change of the Vertical Position Angle for the given AOI	FLOAT4	arcseconds per second	%f			
coord.non_sidereal.target	ephemeris	Name of ephemeris file, if one has been associated with the current target pointing; i.e., this item will contain the name of the ephemeris file if observing via one, else it will be an empty string	STRING	N/A	%s			
coord.non_sidereal.target	prev_time	The previous ephemeris timestamp for the target being observed, i.e., the beginning time for the given {ra,dec}_rate	FLOAT8	seconds	%lf			
coord.non_sidereal.target	stepsize	The stepsize to be used for monitoring the non-sidereal velocity of the target to be observed. This is nominally detected via the ephemeris file; however, it is limited for performance reasons to a minimum threshold.	FLOAT4	seconds	%f	1.0		
coord.non_sidereal.tascu_velocity	dec_rate	The rate of change of Declination, as converted from the Fine Drive, for the non-sidereal target being observed	FLOAT4	arcseconds per second	%f			
coord.non_sidereal.tascu_velocity	ra_rate	The rate of change of Right Ascension, as converted from the Fine Drive, for the non-sidereal target being observed	FLOAT4	arcseconds per second	%f			
coord.non_sidereal.tascu_velocity	vpa_rate	The rate of change of Vertical Position Angle, as converted from the Fine Drive, for the non-sidereal target being observed	FLOAT4	arcseconds per second	%f			
coord.pos	all_names	ASCII list of the Position names	STRING	N/A	%s			
coord.pos	last_changed	Last Position or AOI that was added or deleted	STRING	N/A	%s			
coord.pos	last_distributed	Last Position that was distributed	STRING	N/A	%s			
coord.pos	number	Number of Positions currently defined	INT4	N/A	%d	0		
coord.pos.ANY	alt	object elevation above the horizon	FLOAT8	degrees	%lf			
coord.pos.ANY	azim	object direction in degrees clockwise from north	FLOAT8	degrees	%lf			
coord.pos.ANY	centroid	Default centroid AOI for tracking	STRING	N/A	%s			

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
coord.pos.ANY	chop	Default chop point for TARF and tracking	STRING	N/A	%s			
coord.pos.ANY	coord_sys	Coordinate system for the position	STRING	N/A	%s			
coord.pos.ANY	debug	Tag for debugging	STRING	N/A	%s			
coord.pos.ANY	dec	declination for the Position or AOI chop image	FLOAT8	degrees	%.3sgd			
coord.pos.ANY	dec_user	Declination specified at position definition if coordinate system is ERF; otherwise dec_user is a duplicate of coord.pos.ANY.dec	FLOAT8	degrees	%.3sgd			
coord.pos.ANY	dec_year	This Declination per year is provided to indicate the proper motion as defined at position creation	FLOAT4	milliarcseconds per year	%f			
coord.pos.ANY	el	Telescope fine drive elevation for the Position or AOI chop image assuming the current LOS	FLOAT8	degrees	%lf			
coord.pos.ANY	ephemeris	Name of ephemeris file if one has been associated with this position	STRING	N/A	%s			
coord.pos.ANY	epoch	The reference time (Julian Years) of the proper motion parameters provided by user. The epoch is used to determine the amount of time to apply the proper motion given by ra_year and dec_year. The amount of time is given by ((JD – 2451545.0)/365.25) – (epoch – 2000.0) where JD is the current Julian Day number and 2451545.0 is the Julian Day of January 1, 2000 12:00 UT.	FLOAT4	years	%.3f			
coord.pos.ANY	equinox	Equinox that the position was defined in	STRING	N/A	%s			
coord.pos.ANY	erf_quat	Position expressed as a unitary quaternion	UQUAT32	N/A	%lf			
coord.pos.ANY	ffi_quat	Position expressed as a unitary quaternion	UQUAT32	N/A	%lf			
coord.pos.ANY	fpi_quat	Position expressed as a unitary quaternion	UQUAT32	N/A	%lf			
coord.pos.ANY	inertial	Tracking inertially?, [ENUM: (0: no) (1: yes)]	BOOL4	N/A	%s	0	1	
coord.pos.ANY	irf_quat	Position expressed as a unitary quaternion	UQUAT32	N/A	%lf			
coord.pos.ANY	irf_u	IRF Cartesian coordinate along elevation axis	FLOAT8	N/A	%lf			
coord.pos.ANY	irf_v	IRF Cartesian coordinate along cross-elevation axis	FLOAT8	N/A	%lf			
coord.pos.ANY	irf_w	IRF Cartesian coordinate along LOS axis	FLOAT8	N/A	%lf			
coord.pos.ANY	limb1	Default limb #1 AOI for tracking	STRING	N/A	%s			
coord.pos.ANY	limb2	Default limb #2 AOI for tracking	STRING	N/A	%s			
coord.pos.ANY	los	Current line of sight (LOS) angle for the telescope fine drive	FLOAT8	degrees	%lf			
coord.pos.ANY	lower_right	End pixel of the AOI (supplied as [col row])	STRING	pixels	%s			
coord.pos.ANY	ra	Right Ascension for the Position or AOI in J2000	FLOAT8	sidereal hours	%.3sgh			
coord.pos.ANY	ra_dec_error_estimate	Estimate of error between current and reference positions, excluding ROF/VPA	FLOAT8	arcseconds	%lf			
coord.pos.ANY	ra_user	Right Ascension specified at position definition if coordinate system is ERF; otherwise ra_user is a duplicate of coord.pos.ANY.ra	FLOAT8	sidereal hours	%.3sgh			
coord.pos.ANY	ra_year	This Right Ascension per year is provided to indicate the proper motion as defined at position creation	FLOAT4	milliarcseconds per year	%f			
coord.pos.ANY	ref_dec	A constant reference declination.	FLOAT8	degrees	%.3sgd			
coord.pos.ANY	ref_pos	Name of position that provided the reference values on the sky via the coord.aoi_create or coord.aoi_ref ommands	STRING	N/A	%s			
coord.pos.ANY	ref_ra	A constant reference right ascension.	FLOAT8	sidereal hours	%.3sgh			
coord.pos.ANY	ref_state	State of the reference position: CLEARED or SET, [ENUM: (0: cleared) (1: set)]	STRING	N/A	%s			
coord.pos.ANY	rof1	Default rof #1 AOI for tracking	STRING	N/A	%s			
coord.pos.ANY	rof2	Default rof #2 AOI for tracking	STRING	N/A	%s			
coord.pos.ANY	si_quat	Position expressed as a unitary quaternion	UQUAT32	N/A	%lf			
coord.pos.ANY	state	Is the data valid?, [ENUM: (-1: deleted) (0: uninitialized) (1: created) (2: inactive) (3: active)]	INT4	N/A	%d	-1	3	
coord.pos.ANY	tarf_quat	Position expressed as a unitary quaternion	UQUAT32	N/A	%lf			

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
coord.pos.ANY	tsc_mcs_hk_pkt_timestamp	The packet time associated with the various TA Fine Drive attitudes (specifically, the ta_tsc.tsc_mcs_hk.pkt_timestamp), which more accurately reflects the time of relevance to various coord position data	TIME8	seconds	%d.%09d	0	2147483647	
coord.pos.ANY	type_code	Type of Position object	INT4	N/A	0x%X	0		
coord.pos.ANY	upper_left	Starting pixel of the AOI (supplied as [col row])	STRING	pixels	%s			
coord.pos.ANY	user_count	Number of threads using the AOI	INT4	N/A	%d	-1	100	
coord.pos.ANY	vpa	VPA for the Position or AOI chop image	FLOAT8	degrees	%lf			
coord.pos.ANY	vpa_user	VPA for the Position or AOI chop image	FLOAT8	degrees	%lf			
coord.pos.ANY	wfi_quat	Position expressed as a unitary quaternion	UQUAT32	N/A	%lf			
coord.pos.ANY	xel	Telescope fine drive cross-elevation for the Position or AOI chop image assuming the current LOS	FLOAT8	degrees	%lf			
coord.pos.ANY	xffi	FFI column for the Position or AOI chop image	FLOAT8	pixels	%lf			
coord.pos.ANY	xfpi	FPI column for the Position or AOI chop image	FLOAT8	pixels	%lf			
coord.pos.ANY	xsi	SI column for the Position or AOI chop image	FLOAT8	pixels	%lf			
coord.pos.ANY	xwfi	WFI column for the Position or AOI chop image	FLOAT8	pixels	%lf			
coord.pos.ANY	yffi	FFI row for the Position or AOI chop image	FLOAT8	pixels	%lf			
coord.pos.ANY	yfpi	FPI row for the Position or AOI chop image	FLOAT8	pixels	%lf			
coord.pos.ANY	ysi	SI row for the Position or AOI chop image	FLOAT8	pixels	%lf			
coord.pos.ANY	ywfi	WFI row for the Position or AOI chop image	FLOAT8	pixels	%lf			
coord.position	action	Default action if new argument is wrong, [ENUM: (-1: error) (0: ignore) (1: query)]	STRING	N/A	%s			
coord.responses	aoi_create	Last SCL response sent back to the issuer of the aoi_create command	STRING	N/A	%s			event
coord.responses	aoi_ref	Last SCL response sent back to the issuer of the aoi_ref command	STRING	N/A	%s			event
coord.responses	aoi_ref_clear	Last SCL response sent back to the issuer of the aoi_ref_clear command	STRING	N/A	%s			event
coord.responses	convert	Last SCL response sent back to the issuer of the convert command	STRING	N/A	%s			event
coord.responses	correct	Last SCL response sent back to the issuer of the correct command	STRING	N/A	%s			event
coord.responses	delete	Last SCL response sent back to the issuer of the delete command	STRING	N/A	%s			event
coord.responses	init	Last SCL response sent back to the issuer of the init command	STRING	N/A	%s			event
coord.responses	instrument	Last SCL response sent back to the issuer of the instrument command	STRING	N/A	%s			event
coord.responses	list	Last SCL response sent back to the issuer of the list command	STRING	N/A	%s			event
coord.responses	position	Last SCL response sent back to the issuer of the position command	STRING	N/A	%s			event
coord.responses	quiet	Last SCL response sent back to the issuer of the quiet command	STRING	N/A	%s			event
coord.responses	resume	Last SCL response sent back to the issuer of the resume command	STRING	N/A	%s			event
coord.responses	set	Last SCL response sent back to the issuer of the set command	STRING	N/A	%s			event
coord.responses	suspend	Last SCL response sent back to the issuer of the suspend command	STRING	N/A	%s			event
coord.responses	verbose	Last SCL response sent back to the issuer of the verbose command	STRING	N/A	%s			event
coord.pos_rate	def_nsample	Number of TASCU HK samples between computation of default positions; this defaults to 1 on startup and may be changed by coord.set_pos_rate command	UINT2	N/A	%d			event
coord.pos_rate	fast_nsample	Number of TASCU HK samples between computation of fast positions; this defaults to 1 on startup and may be changed by coord.set_pos_rate command	UINT2	N/A	%d			event
coord.pos_rate	fast_pos_list	List of position names published at rate indicated by coord.pos_rate.fast_nsample value; defaults to an empty list on startup, but may be changed by coord.set_pos_rate command; the list is case-insensitive	STRING	N/A	%s			event
coord.si	exptime	SI integration duration in seconds, set upon completion of integration; this may correspond to the value inserted as keyword EXPTIME in the SI data FITS header	FLOAT8	seconds	%.3lf			
coord.si	integ	SI integration-in-progress flag that indicates which part of the SI is integrating (if there are multiple integration sources) and whether an integration is starting or ending [value = 10 * source# + 1 (start integ) or 0 (stop integ)]	INT4	N/A	%d			
coord.si	utcstart	UTC start time for integration from an SI source; in seconds since 1970 (i.e., Unix time)	FLOAT8	seconds	%.3lf			

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
coord.si	utcend	UTC end time for integration from an SI source; in seconds since 1970 (i.e., Unix time)	FLOAT8	seconds	%.3lf			
das								
das	rcs_id	Version information for the XML file that defines this data source	STRING	N/A	%s			event
das.ac_engine	engine_thrust	engine thrust	FLOAT4	pounds	%f			1
das.ac_engine	epr_1	EPR #1	FLOAT4	N/A	%f			1
das.ac_engine	epr_2	EPR #2	FLOAT4	N/A	%f			1
das.ac_engine	epr_3	EPR #3	FLOAT4	N/A	%f			1
das.ac_engine	epr_4	EPR #4	FLOAT4	N/A	%f			1
das.ac_engine	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.ac_fuel	center_tank_fuel_qty	Center tank fuel quantity	FLOAT4	pounds	%f			1
das.ac_fuel	gross_weight	gross weight	FLOAT4	pounds	%f			1
das.ac_fuel	main_reserve_1_fuel_qty	Main Reserve #1 fuel quantity	FLOAT4	pounds	%f			1
das.ac_fuel	main_reserve_2_fuel_qty	Main Reserve #2 fuel quantity	FLOAT4	pounds	%f			1
das.ac_fuel	main_reserve_3_fuel_qty	Main Reserve #3 fuel quantity	FLOAT4	pounds	%f			1
das.ac_fuel	main_reserve_4_fuel_qty	Main Reserve #4 fuel quantity	FLOAT4	pounds	%f			1
das.ac_fuel	main_tank_1_fuel_qty	Main Tank #1 fuel quantity	FLOAT4	pounds	%f			1
das.ac_fuel	main_tank_2_fuel_qty	Main Tank #2 fuel quantity	FLOAT4	pounds	%f			1
das.ac_fuel	main_tank_3_fuel_qty	Main Tank #3 fuel quantity	FLOAT4	pounds	%f			1
das.ac_fuel	main_tank_4_fuel_qty	Main Tank #4 fuel quantity	FLOAT4	pounds	%f			1
das.ac_fuel	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.adc_1_15hz	adc_vert_speed	adc vertical speed (ADC 1)	FLOAT4	feet per minute	%f			1
das.adc_1_15hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.adc_1_15hz	press_alt	pressure altitude (ADC 1)	FLOAT4	feet	%f			1
das.adc_1_2hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.adc_1_2hz	static_air_temp	static air temperature (OAT) (ADC 1)	FLOAT4	degrees Celsius	%f			1
das.adc_1_2hz	static_outside_air_press	static outside air pressure (ADC 1)	FLOAT4	inches of mercury	%f			1
das.adc_1_8hz	indicated_airspeed	indicated airspeed (ADC 1)	FLOAT4	knots	%f			1
das.adc_1_8hz	mach	Mach (ADC 1)	FLOAT4	N/A	%f			1
das.adc_1_8hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.adc_1_8hz	true_airspeed	true airspeed (ADC 1)	FLOAT4	knots	%f			1
das.adc_2_15hz	adc_vert_speed	adc vertical speed (ADC 2)	FLOAT4	feet per minute	%f			1
das.adc_2_15hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.adc_2_15hz	press_alt	pressure altitude (ADC 2)	FLOAT4	feet	%f			1
das.adc_2_2hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.adc_2_2hz	static_air_temp	static air temperature (OAT) (ADC 2)	FLOAT4	degrees Celsius	%f			1
das.adc_2_2hz	static_outside_air_press	static outside air pressure (ADC 2)	FLOAT4	inches of mercury	%f			1
das.adc_2_8hz	indicated_airspeed	indicated airspeed (ADC 2)	FLOAT4	knots	%f			1
das.adc_2_8hz	mach	Mach (ADC 2)	FLOAT4	N/A	%f			1

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
das.adc_2_8hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.adc_2_8hz	true_airspeed	true airspeed (ADC 2)	FLOAT4	knots	%f			1
das.archive_info	archive_file_name	Archive file name	STRING	N/A	%s			event
das.archive_info	archive_status	[ENUM: (0: false) (1: true)]	BOOL4	N/A	%s	0	1	
das.archive_info	current_bytes_archived	Total bytes written to the current ark file	FLOAT8	bytes	%.0lf	0		
das.archive_info	total_bytes_archived	Total bytes written since this data source starts	FLOAT8	bytes	%.0lf	0		
das.archive_nsamples	current_bytes_archived_nsample		INT4	N/A	%d	0		event
das.archive_nsamples	total_bytes_archived_nsample		INT4	N/A	%d	0		event
das.cabin_env	1st_class_humidity	1st class humidity	FLOAT4	percent	%f			1
das.cabin_env	1st_class_temp	1st class temperature	FLOAT4	degrees Fahrenheit	%f			1
das.cabin_env	cabin_alt	cabin altitude (AKA cabin pressure)	FLOAT4	pounds per square inch absolute	%f			1
das.cabin_env	epo_console_temp_1	EPO console temperature #1	FLOAT4	degrees Celsius	%f			1
das.cabin_env	epo_console_temp_2	EPO console temperature #2	FLOAT4	degrees Celsius	%f			1
das.cabin_env	fwd_rack_3_temp_1	FWD rack 3 temperature #1	FLOAT4	degrees Celsius	%f			1
das.cabin_env	fwd_rack_4_temp_1	FWD rack 4 temperature #1	FLOAT4	degrees Celsius	%f			1
das.cabin_env	fwd_rack_4_temp_2	FWD rack 4 temperature #2	FLOAT4	degrees Celsius	%f			1
das.cabin_env	fwd_rack_5_temp_1	FWD rack 5 temperature #1	FLOAT4	degrees Celsius	%f			1
das.cabin_env	fwd_rack_5_temp_2	FWD rack 5 temperature #2	FLOAT4	degrees Celsius	%f			1
das.cabin_env	fwd_rack_5_temp_3	FWD rack 5 temperature #3	FLOAT4	degrees Celsius	%f			1
das.cabin_env	instrument_flange_dewpoint	Instrument Flange Dewpoint	FLOAT4	degrees Fahrenheit	%f			1
das.cabin_env	main_cabin_upstairs_temp	Main cabin upstairs temperature	FLOAT4	degrees Fahrenheit	%f			1
das.cabin_env	md_console_temp_1	MD console temperature #1	FLOAT4	degrees Celsius	%f			1
das.cabin_env	md_console_temp_2	MD console temperature #2	FLOAT4	degrees Celsius	%f			1
das.cabin_env	mission_area_humidity	Mission area humidity	FLOAT4	percent	%f			1
das.cabin_env	mission_area_temp	Mission area temperature	FLOAT4	degrees Fahrenheit	%f			1
das.cabin_env	optical_window_dewpoint	Optical Window Dewpoint	FLOAT4	degrees Fahrenheit	%f			1
das.cabin_env	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.cabin_env	ta_area_humidity	TA area humidity (main cabin)	FLOAT4	percent	%f			1
das.cabin_env	ta_area_temp	TA area temperature (main cabin)	FLOAT4	degrees Fahrenheit	%f			1
das.cabin_env	to_console_temp_1	TO console temperature #1	FLOAT4	degrees Celsius	%f			1

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
das.cabin_env	to_console_temp_2	TO console temperature #2	FLOAT4	degrees Celsius	%f			1
das.cavity_env	aft_cavity_temp	Aft cavity temperature	FLOAT4	degrees Fahrenheit	%f			1
das.cavity_env	cavity_dewpoint	cavity dewpoint	FLOAT4	degrees Celsius	%f			1
das.cavity_env	cavity_door_air_bottle_press	cavity door air bottle pressure	FLOAT4	pounds per square inch absolute	%f			1
das.cavity_env	cavity_door_seal_press	cavity door seal pressure	FLOAT4	pounds per square inch absolute	%f			1
das.cavity_env	cavity_door_seal_press_diff	Cavity Door seal pressure differential	FLOAT4	pounds per square inch differential	%f			1
das.cavity_env	cavity_humidity	cavity humidity	FLOAT4	percent	%f			1
das.cavity_env	cavity_recovery_temp	Cavity recovery temperature	FLOAT4	degrees Fahrenheit	%f			1
das.cavity_env	cavity_rh_aft_dewpoint	Cavity RH Aft Dewpoint	FLOAT4	degrees Fahrenheit	%f			1
das.cavity_env	forward_cavity_temp	Forward cavity temperature	FLOAT4	degrees Fahrenheit	%f			1
das.cavity_env	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.cavity_env	seal_vent_press	Seal Vent Pressure	FLOAT4	pounds per square inch absolute	%f			1
das.cavity_env	static_cavity_press	static cavity pressure	FLOAT4	pounds per square inch differential	%f			1
das.cbit_results	last_result_test_1	last result - Test 1 (Clock), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_10	last result - Test 10 (MV64360 Enet2), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_11	last result - Test 11 (MV64360 MPSC1), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_12	last result - Test 12 (MV64360 MPSC2), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_13	last result - Test 13 (MV64360 SRAM), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_14	last result - Test 14 (MV64360 DMA), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_15	last result - Test 15 (MV64360 Timer), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_16	last result - Test 16 (MV64360 PIC), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
das.cbit_results	last_result_test_17	last result - Test 17 (SDRAM), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_18	last result - Test 18 (FLASH A), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_19	last result - Test 19 (FLASH B), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_2	last result - Test 2 (CPU), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_20	last result - Test 20 (FPGA), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_21	last result - Test 21 (FPGA DPRAM), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_22	last result - Test 22 (FPGA GPIO), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_23	last result - Test 23 (FPGA WD), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_24	last result - Test 24 (FPGA PIC), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_25	last result - Test 25 (PCI), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_26	last result - Test 26 (RTC), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_27	last result - Test 27 (UART A), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_28	last result - Test 28 (UART B), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_29	last result - Test 29 (PMC 1), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_3	last result - Test 3 (L1 Inst Cache), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_30	last result - Test 30 (PMC 2), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_31	last result - Test 31 (CPU Temp), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_32	last result - Test 32 (Board Temp), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_33	last result - Test 33 (NV resistor), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_34	last result - Test 34 (SPD EEPROM), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_35	last result - Test 35 (User EEPROM), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_36	last result - Test 36 (Chipset EEPROM), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_37	last result - Test 37 (VME64), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_38	last result - Test 38 (SATA), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_39	last result - Test 39 (ADC), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_4	last result - Test 4 (L1 Data Cache), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
das.cbit_results	last_result_test_40	last result - Test 40 (USER 1), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_41	last result - Test 41 (USER 2), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_42	last result - Test 42 (USER 3), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_43	last result - Test 43 (USER 4), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_5	last result - Test 5 (L2 Cache), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_6	last result - Test 6 (L3 Cache), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_7	last result - Test 7 (MV64360), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_8	last result - Test 8 (MV64360 TWSI), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	last_result_test_9	last result - Test 9 (MV64360 Enet1), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			0.14
das.cbit_results	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			0.14
das.cbit_results	total_errors_test_1	total errors - Test 1 (Clock)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_10	total errors - Test 10 (MV64360 Enet2)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_11	total errors - Test 11 (MV64360 MPSC1)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_12	total errors - Test 12 (MV64360 MPSC2)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_13	total errors - Test 13 (MV64360 SRAM)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_14	total errors - Test 14 (MV64360 DMA)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_15	total errors - Test 15 (MV64360 Timer)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_16	total errors - Test 16 (MV64360 PIC)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_17	total errors - Test 17 (SDRAM)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_18	total errors - Test 18 (FLASH A)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_19	total errors - Test 19 (FLASH B)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_2	total errors - Test 2 (CPU)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_20	total errors - Test 20 (FPGA)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_21	total errors - Test 21 (FPGA DPRAM)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_22	total errors - Test 22 (FPGA GPIO)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_23	total errors - Test 23 (FPGA WD)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_24	total errors - Test 24 (FPGA PIC)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_25	total errors - Test 25 (PCI)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_26	total errors - Test 26 (RTC)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_27	total errors - Test 27 (UART A)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_28	total errors - Test 28 (UART B)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_29	total errors - Test 29 (PMC 1)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_3	total errors - Test 3 (L1 Inst Cache)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_30	total errors - Test 30 (PMC 2)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_31	total errors - Test 31 (CPU Temp)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_32	total errors - Test 32 (Board Temp)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_33	total errors - Test 33 (NV resistor)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_34	total errors - Test 34 (SPD EEPROM)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_35	total errors - Test 35 (User EEPROM)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_36	total errors - Test 36 (Chipset EEPROM)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_37	total errors - Test 37 (VME64)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_38	total errors - Test 38 (SATA)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_39	total errors - Test 39 (ADC)	UINT4	N/A	%d			0.14

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
das.cbit_results	total_errors_test_4	total errors - Test 4 (L1 Data Cache)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_40	total errors - Test 40 (USER 1)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_41	total errors - Test 41 (USER 2)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_42	total errors - Test 42 (USER 3)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_43	total errors - Test 43 (USER 4)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_5	total errors - Test 5 (L2 Cache)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_6	total errors - Test 6 (L3 Cache)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_7	total errors - Test 7 (MV64360)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_8	total errors - Test 8 (MV64360 TWSI)	UINT4	N/A	%d			0.14
das.cbit_results	total_errors_test_9	total errors - Test 9 (MV64360 Enet1)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_1	total runs - Test 1 (Clock)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_10	total runs - Test 10 (MV64360 Enet2)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_11	total runs - Test 11 (MV64360 MPSC1)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_12	total runs - Test 12 (MV64360 MPSC2)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_13	total runs - Test 13 (MV64360 SRAM)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_14	total runs - Test 14 (MV64360 DMA)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_15	total runs - Test 15 (MV64360 Timer)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_16	total runs - Test 16 (MV64360 PIC)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_17	total runs - Test 17 (SDRAM)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_18	total runs - Test 18 (FLASH A)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_19	total runs - Test 19 (FLASH B)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_2	total runs - Test 2 (CPU)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_20	total runs - Test 20 (FPGA)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_21	total runs - Test 21 (FPGA DPRAM)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_22	total runs - Test 22 (FPGA GPIO)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_23	total runs - Test 23 (FPGA WD)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_24	total runs - Test 24 (FPGA PIC)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_25	total runs - Test 25 (PCI)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_26	total runs - Test 26 (RTC)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_27	total runs - Test 27 (UART A)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_28	total runs - Test 28 (UART B)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_29	total runs - Test 29 (PMC 1)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_3	total runs - Test 3 (L1 Inst Cache)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_30	total runs - Test 30 (PMC 2)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_31	total runs - Test 31 (CPU Temp)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_32	total runs - Test 32 (Board Temp)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_33	total runs - Test 33 (NV resistor)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_34	total runs - Test 34 (SPD EEPROM)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_35	total runs - Test 35 (User EEPROM)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_36	total runs - Test 36 (Chipset EEPROM)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_37	total runs - Test 37 (VME64)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_38	total runs - Test 38 (SATA)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_39	total runs - Test 39 (ADC)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_4	total runs - Test 4 (L1 Data Cache)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_40	total runs - Test 40 (USER 1)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_41	total runs - Test 41 (USER 2)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_42	total runs - Test 42 (USER 3)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_43	total runs - Test 43 (USER 4)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_5	total runs - Test 5 (L2 Cache)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_6	total runs - Test 6 (L3 Cache)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_7	total runs - Test 7 (MV64360)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_8	total runs - Test 8 (MV64360 TWSI)	UINT4	N/A	%d			0.14
das.cbit_results	total_runs_test_9	total runs - Test 9 (MV64360 Enet1)	UINT4	N/A	%d			0.14

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
das.cecs	cecs_drum_detector	cecs drum detector sink transistor	UINT2	N/A	%d			1
das.cecs	cecs_dryer_regen_temp	CECS dryer regen temperature	FLOAT4	degrees Fahrenheit	%f			1
das.cecs	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.clear_faults_resp	information	information (for clear_faults command response)	STRING	N/A	%s			
das.clear_faults_resp	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			
das.clear_faults_resp	response	response (for clear_faults command), [ENUM: (1: COMMAND_ACKNOWLEDGED) (2: COMMAND_INITIATED) (3: COMMAND_COMPLETED) (4: COMMAND_REJECTED_INVALID_ARGUMENT) (5: COMMAND_REJECTED_INVALID_HEADER_SYNC) (6: COMMAND_REJECTED_INVALID_HEADER_SIZE) (7: COMMAND_REJECTED_INVALID_HEADER_MSGID) (8: COMMAND_REJECTED_INVALID_HEADER_PKTTYPE) (100: FILE_WRITE_FAILURE) (101: FILE_READ_FAILURE) (102: FILE_PARSE_FAILURE) (103: INVALID_GROUP) (254: COMMAND_ABORT_FAILURE) (255: COMMAND_ABORT_SUCCESS)]	UINT1	N/A	%d			
das.das_alarm	das_alarm_error	Error alert	STRING	N/A	%s			event
das.das_alarm	das_alarm_fatal	Fatal alert	STRING	N/A	%s			event
das.das_alarm	das_alarm_info	Information alarm	STRING	N/A	%s			event
das.das_alarm	das_alarm_warning	Warning alarm	STRING	N/A	%s			event
das.das_alert	das_alert_error	Error alert	STRING	N/A	%s			event
das.das_alert	das_alert_fatal	Fatal alert	STRING	N/A	%s			event
das.das_alert	das_alert_info	Information alert	STRING	N/A	%s			event
das.das_alert	das_alert_warning	Warning alert	STRING	N/A	%s			event
das.disable_arc_resp	information	information (for disable_archive command response)	STRING	N/A	%s			
das.disable_arc_resp	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			
das.disable_arc_resp	response	response (for disable_archive command), [ENUM: (1: COMMAND_ACKNOWLEDGED) (2: COMMAND_INITIATED) (3: COMMAND_COMPLETED) (4: COMMAND_REJECTED_INVALID_ARGUMENT) (5: COMMAND_REJECTED_INVALID_HEADER_SYNC) (6: COMMAND_REJECTED_INVALID_HEADER_SIZE) (7: COMMAND_REJECTED_INVALID_HEADER_MSGID) (8: COMMAND_REJECTED_INVALID_HEADER_PKTTYPE) (100: FILE_WRITE_FAILURE) (101: FILE_READ_FAILURE) (102: FILE_PARSE_FAILURE) (103: INVALID_GROUP) (254: COMMAND_ABORT_FAILURE) (255: COMMAND_ABORT_SUCCESS)]	UINT1	N/A	%d			
das.disable_group_resp	information	information (for disable_group command response)	STRING	N/A	%s			
das.disable_group_resp	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			
das.disable_group_resp	response	response (for disable_group command), [ENUM: (1: COMMAND_ACKNOWLEDGED) (2: COMMAND_INITIATED) (3: COMMAND_COMPLETED) (4: COMMAND_REJECTED_INVALID_ARGUMENT) (5: COMMAND_REJECTED_INVALID_HEADER_SYNC) (6: COMMAND_REJECTED_INVALID_HEADER_SIZE) (7: COMMAND_REJECTED_INVALID_HEADER_MSGID) (8: COMMAND_REJECTED_INVALID_HEADER_PKTTYPE) (100: FILE_WRITE_FAILURE) (101: FILE_READ_FAILURE) (102: FILE_PARSE_FAILURE) (103: INVALID_GROUP) (254: COMMAND_ABORT_FAILURE) (255: COMMAND_ABORT_SUCCESS)]	UINT1	N/A	%d			

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
das.dss	inlet_outlet_press_diff	Inlet /Outlet pressure differential (DSS)	FLOAT4	pounds per square inch differential	%f			1
das.dss	inlet_press	Inlet Pressure (DSS)	FLOAT4	pounds per square inch absolute	%f			1
das.dss	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.enable_arc_resp	information	information (for enable_archive command response)	STRING	N/A	%s			event
das.enable_arc_resp	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			event
das.enable_arc_resp	response	response (for enable_archive command), [ENUM: (1: COMMAND_ACKNOWLEDGED) (2: COMMAND_INITIATED) (3: COMMAND_COMPLETED) (4: COMMAND_REJECTED_INVALID_ARGUMENT) (5: COMMAND_REJECTED_INVALID_HEADER_SYNC) (6: COMMAND_REJECTED_INVALID_HEADER_SIZE) (7: COMMAND_REJECTED_INVALID_HEADER_MSGID) (8: COMMAND_REJECTED_INVALID_HEADER_PKTTYPE) (100: FILE_WRITE_FAILURE) (101: FILE_READ_FAILURE) (102: FILE_PARSE_FAILURE) (103: INVALID_GROUP) (254: COMMAND_ABORT_FAILURE) (255: COMMAND_ABORT_SUCCESS)]	UINT1	N/A	%d			event
das.enable_group_resp	information	information (for enable_group command response)	STRING	N/A	%s			event
das.enable_group_resp	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			event
das.enable_group_resp	response	response (for enable_group command), [ENUM: (1: COMMAND_ACKNOWLEDGED) (2: COMMAND_INITIATED) (3: COMMAND_COMPLETED) (4: COMMAND_REJECTED_INVALID_ARGUMENT) (5: COMMAND_REJECTED_INVALID_HEADER_SYNC) (6: COMMAND_REJECTED_INVALID_HEADER_SIZE) (7: COMMAND_REJECTED_INVALID_HEADER_MSGID) (8: COMMAND_REJECTED_INVALID_HEADER_PKTTYPE) (100: FILE_WRITE_FAILURE) (101: FILE_READ_FAILURE) (102: FILE_PARSE_FAILURE) (103: INVALID_GROUP) (254: COMMAND_ABORT_FAILURE) (255: COMMAND_ABORT_SUCCESS)]	UINT1	N/A	%d			event
das.error	message	message (Error)	STRING	N/A	%s			event
das.error	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			event
das.fms_1_10hz	baro_alt	barometric altitude (FMS 1)	FLOAT4	feet	%f			1
das.fms_1_10hz	cross_track_dist	cross-track distance	FLOAT4	nautical miles	%f			1
das.fms_1_10hz	dist_to_dest	distance to destination (FMS 1)	FLOAT4	nautical miles	%f			1
das.fms_1_10hz	dist_to_next_wypt	distance to next waypoint (FMS 1)	FLOAT4	nautical miles	%f			1
das.fms_1_10hz	est_time_to_dest	estimated time to destination (FMS 1)	FLOAT4	minutes	%f			1
das.fms_1_10hz	est_time_to_next_wypt	estimated time to next waypoint (FMS 1)	FLOAT4	minutes	%f			1
das.fms_1_10hz	ground_speed	ground speed (FMS 1)	FLOAT4	knots	%f			1
das.fms_1_10hz	mag_dev	magnetic deviation	FLOAT4	degrees	%f			1
das.fms_1_10hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.fms_2_10hz	baro_alt	barometric altitude (FMS 2)	FLOAT4	feet	%f			1

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
das.fms_2_10hz	cross_track_dist	cross-track distance	FLOAT4	nautical miles	%f			1
das.fms_2_10hz	dist_to_dest	distance to destination (FMS 2)	FLOAT4	nautical miles	%f			1
das.fms_2_10hz	dist_to_next_wypt	distance to next waypoint (FMS 2)	FLOAT4	nautical miles	%f			1
das.fms_2_10hz	est_time_to_dest	estimated time to destination (FMS 2)	FLOAT4	minutes	%f			1
das.fms_2_10hz	est_time_to_next_wypt	estimated time to next waypoint (FMS 2)	FLOAT4	minutes	%f			1
das.fms_2_10hz	ground_speed	ground speed (FMS 2)	FLOAT4	knots	%f			1
das.fms_2_10hz	mag_dev	magnetic deviation	FLOAT4	degrees	%f			1
das.fms_2_10hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.fms_3_10hz	baro_alt	barometric altitude (FMS 3)	FLOAT4	feet	%f			1
das.fms_3_10hz	cross_track_dist	cross-track distance	FLOAT4	nautical miles	%f			1
das.fms_3_10hz	dist_to_dest	distance to destination (FMS 3)	FLOAT4	nautical miles	%f			1
das.fms_3_10hz	dist_to_next_wypt	distance to next waypoint (FMS 3)	FLOAT4	nautical miles	%f			1
das.fms_3_10hz	est_time_to_dest	estimated time to destination (FMS 3)	FLOAT4	minutes	%f			1
das.fms_3_10hz	est_time_to_next_wypt	estimated time to next waypoint (FMS 3)	FLOAT4	minutes	%f			1
das.fms_3_10hz	ground_speed	ground speed (FMS 3)	FLOAT4	knots	%f			1
das.fms_3_10hz	mag_dev	magnetic deviation	FLOAT4	degrees	%f			1
das.fms_3_10hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.gps_1_10hz	gps_lat	gps latitude (GPS 1)	FLOAT8	degrees	%.13lf			1
das.gps_1_10hz	gps_lon	gps longitude (GPS 1)	FLOAT8	degrees	%.13lf			1
das.gps_1_10hz	gps_msl_alt	gps MSL altitude (GPS 1)	FLOAT4	feet	%f			1
das.gps_1_10hz	gps_vert_speed	gps vertical speed (GPS 1)	FLOAT4	feet per minute	%f			1
das.gps_1_10hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.gps_1_10hz	sampletime	timestamp from GPS LRU (GPS 1)	FLOAT8	seconds	%.3ymdz			1
das.gps_2_10hz	gps_lat	gps latitude (GPS 2)	FLOAT8	degrees	%.13lf			1
das.gps_2_10hz	gps_lon	gps longitude (GPS 2)	FLOAT8	degrees	%.13lf			1
das.gps_2_10hz	gps_msl_alt	gps MSL altitude (GPS 2)	FLOAT4	feet	%f			1
das.gps_2_10hz	gps_vert_speed	gps vertical speed (GPS 2)	FLOAT4	feet per minute	%f			1
das.gps_2_10hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.gps_2_10hz	sampletime	timestamp from GPS LRU (GPS 2)	FLOAT8	seconds	%.3ymdz			1
das.hall_effect	hall_effect_1	Hall effect #1	BINARRAY, [Number of Element: 40, Representation: FLOAT4]	N/A	%f			50

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
das.hall_effect	hall_effect_2	Hall effect #2	BINARRAY, [Number of Element: 40, Representation: FLOAT4]	N/A	%f			50
das.hall_effect	hall_effect_3	Hall effect #3	BINARRAY, [Number of Element: 40, Representation: FLOAT4]	N/A	%f			50
das.hall_effect	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			50
das.health_status	dadc_stat	DADC Stat, [ENUM: (0: No_data) (1: DADC#1_active) (2: DADC#2_active)]	UINT1	N/A	%d			1
das.health_status	das_proxy_stat	DAS Proxy Stat, [ENUM: (0: Not_Connected) (1: Connected)]	UINT1	N/A	%d			1
das.health_status	fms_stat	FMS Stat, [ENUM: (0: No_data) (1: FMS#1_active) (2: FMS#2_active) (4: FMS#3_active)]	UINT1	N/A	%d			1
das.health_status	fv_stat	FV Stat, [ENUM: (0: Not_Connected) (1: Connected)]	UINT1	N/A	%d			1
das.health_status	go_nogo	Go/NoGo, [ENUM: (0: NoGo) (1: Go)]	UINT4	N/A	%d			1
das.health_status	gps_stat	GPS Stat, [ENUM: (0: No_data) (1: GPS#1_active) (2: GPS#2_active)]	UINT1	N/A	%d			1
das.health_status	ic_1080_stat	IC 1080 Stat, [ENUM: (0: No_data) (1: IC_1080#1_active) (2: IC_1080#2_active)]	UINT1	N/A	%d			1
das.health_status	ins_stat	INS Stat, [ENUM: (0: No_data) (1: INS#1_active) (2: INS#2_active) (4: INS#3_active)]	UINT1	N/A	%d			1
das.health_status	instro_stat	Instro Stat, [ENUM: (0: Not_Connected) (1: Connected)]	UINT1	N/A	%d			1
das.health_status	internal_stat	Internal Stat, [ENUM: (0: No_fault) (1: PBIT_fail) (2: CBIT_fail) (4: SW_Audit_fail) (8: NodeB_High_Temp) (16: Mem_Low) (32: Disk_fail) (64: Error_detected)]	UINT1	N/A	%d			1
das.health_status	mem_reserve	Memory Reserve	FLOAT4	percent	%f			1
das.health_status	nfs_stat	NFS Stat, [ENUM: (0: None_mounted) (1: modstor_mounted) (2: das_archive_mounted) (4: das_log_mounted) (8: das_config_mounted)]	UINT1	N/A	%d			1
das.health_status	node_a_temp	Node A temperature	FLOAT4	degrees Celsius	%f			1
das.health_status	node_b_temp	Node B temperature	FLOAT4	degrees Celsius	%f			1
das.health_status	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.health_status	proc_reserve	Processing Reserve	FLOAT4	percent	%f			1
das.health_status	time_stat	Time Stat, [ENUM: (0: No_Time) (1: SNTP_Synced) (2: IRIG_Lock)]	UINIT1	N/A	%d			1
das.hepa_filter	hepa_filter_input_press	HEPA Filter Input Pressure	FLOAT4	pounds per square inch absolute	%f			1
das.hepa_filter	hepa_filter_output_press	HEPA Filter Output Pressure	FLOAT4	pounds per square inch absolute	%f			1
das.hepa_filter	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.ic1080_10hz	baro_alt	baro altitude (Left Displayed ADC) (IC1080 1)	FLOAT4	feet	%f			10
das.ic1080_10hz	indicated_airspeed	indicated airspeed (Left Displayed ADC) (IC1080 1)	FLOAT4	knots	%f			10
das.ic1080_10hz	lat_accel	lateral acceleration (IC1080 1)	FLOAT4	gravities	%f			10
das.ic1080_10hz	lon_accel	longitudinal acceleration (IC1080 1)	FLOAT4	gravities	%f			10
das.ic1080_10hz	norm_accel	normal acceleration (IC1080 1)	FLOAT4	gravities	%f			10
das.ic1080_10hz	pitch	pitch (Left Displayed ATT/HDG Source) (IC1080 1)	FLOAT4	degrees	%f			10
das.ic1080_10hz	pitch_sampletime	pitch sampletime (IC1080 1)	FLOAT8	seconds	%.3ymdz			10

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
das.ic1080_10hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			10
das.ic1080_10hz	true_airspeed	true airspeed (Left Displayed ADC) (IC1080 1)	FLOAT4	knots	%f			10
das.ic1080_15hz	adc_vert_speed	adc vertical speed (Left Displayed ADC) (IC1080 1)	FLOAT4	feet per minute	%f			15
das.ic1080_15hz	baro_correction	baro correction	FLOAT4	inches of mercury	%f			15
das.ic1080_15hz	baro_set_type	baro set type, [ENUM: (0: inches of mercury) (1: hectopascals)]	UINT1	N/A	%d			15
das.ic1080_15hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			15
das.ic1080_15hz	press_alt	Pressure altitude (Left Displayed ADC) (IC1080 1)	FLOAT4	feet	%f			15
das.ic1080_1hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.ic1080_1hz	wind_angle	wind angle (IC1080 1)	FLOAT4	degrees	%f			1
das.ic1080_1hz	wind_speed	wind speed (IC1080 1)	FLOAT4	knots	%f			1
das.ic1080_2_10hz	baro_alt	baro altitude (Left Displayed ADC) (IC1080 2)	FLOAT4	feet	%f			10
das.ic1080_2_10hz	indicated_airspeed	indicated airspeed (Left Displayed ADC) (IC1080 2)	FLOAT4	knots	%f			10
das.ic1080_2_10hz	lat_accel	lateral acceleration (IC1080 2)	FLOAT4	gravities	%f			10
das.ic1080_2_10hz	lon_accel	longitudinal acceleration (IC1080 2)	FLOAT4	gravities	%f			10
das.ic1080_2_10hz	norm_accel	normal acceleration (IC1080 2)	FLOAT4	gravities	%f			10
das.ic1080_2_10hz	pitch	pitch (Left Displayed ATT/HDG Source) (IC1080 2)	FLOAT4	degrees	%f			10
das.ic1080_2_10hz	pitch_sampletime	pitch sampletime (IC1080 2)	FLOAT8	seconds	%.3ymdz			10
das.ic1080_2_10hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			10
das.ic1080_2_10hz	true_airspeed	true airspeed (Left Displayed ADC) (IC1080 2)	FLOAT4	knots	%f			10
das.ic1080_2_15hz	adc_vert_speed	adc vertical speed (Left Displayed ADC) (IC1080 2)	FLOAT4	feet per minute	%f			15
das.ic1080_2_15hz	baro_correction	baro correction	FLOAT4	inches of mercury	%f			15
das.ic1080_2_15hz	baro_set_type	baro set type, [ENUM: (0: inches of mercury) (1: hectopascals)]	UINT1	N/A	%d			15
das.ic1080_2_15hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			15
das.ic1080_2_15hz	press_alt	Pressure altitude (Left Displayed ADC) (IC1080 2)	FLOAT4	feet	%f			15
das.ic1080_2_1hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.ic1080_2_1hz	wind_angle	wind angle (IC1080 2)	FLOAT4	degrees	%f			1
das.ic1080_2_1hz	wind_speed	wind speed (IC1080 2)	FLOAT4	knots	%f			1
das.ic1080_2_2hz	ground_speed	ground speed (IC1080 2)	FLOAT4	knots	%f			2
das.ic1080_2_2hz	ground_speed_fms	ground speed - FMS (Left & Right Displayed FMS) (IC1080 2)	FLOAT4	knots	%f			2
das.ic1080_2_2hz	lat_fms_1	latitude - FMS 1 (geodetic) (Left Displayed FMS1) (IC1080 2)	FLOAT4	degrees	%f			2
das.ic1080_2_2hz	lat_fms_1_sampletime	latitude - FMS 1 sampletime (IC1080 2)	FLOAT8	seconds	%.3ymdz			2
das.ic1080_2_2hz	lat_fms_2	latitude - FMS 2 (geodetic) (Right Displayed FMS2) (IC1080 2)	FLOAT4	degrees	%f			2
das.ic1080_2_2hz	lat_fms_2_sampletime	latitude - FMS 2 sampletime (IC1080 2)	FLOAT8	seconds	%.3ymdz			2
das.ic1080_2_2hz	lon_fms_1	longitude - FMS 1 (geodetic) (Left Displayed FMS1) (IC1080 2): West is negative and East is positive with respect to the Prime Meridian	FLOAT4	degrees	%f			2
das.ic1080_2_2hz	lon_fms_1_sampletime	longitude - FMS 1 sampletime (IC1080 2)	FLOAT8	seconds	%.3ymdz			2
das.ic1080_2_2hz	lon_fms_2	longitude - FMS 2 (geodetic) (Right Displayed FMS2) (IC1080 2): West is negative and East is positive with respect to the Prime Meridian	FLOAT4	degrees	%f			2

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
das.ic1080_2_2hz	lon_fms_2_sampletime	longitude - FMS 2 sampletime (IC1080 2)	FLOAT8	seconds	%.3ymdz			2
das.ic1080_2_2hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			2
das.ic1080_2_2hz	static_air_temp	static air temperature (Left Displayed ADC) (IC1080 2)	FLOAT4	degrees Celsius	%f			2
das.ic1080_2_2hz	true_heading	true heading (Left Displayed ATT/HDG Source) (IC1080 2)	FLOAT4	degrees	%f			2
das.ic1080_2_2hz	true_track_angle	true track angle (IC1080 2)	FLOAT4	degrees	%f			2
das.ic1080_2_5hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			5
das.ic1080_2_5hz	roll	roll (IC1080 2)	FLOAT4	degrees	%f			5
das.ic1080_2_5hz	roll_sampletime	roll sampletime (IC1080 2)	FLOAT8	seconds	%.3ymdz			5
das.ic1080_2hz	east_velocity_filtered	east velocity filtered	FLOAT4	knots	%f			2
das.ic1080_2hz	ground_speed	ground speed (IC1080 1)	FLOAT4	knots	%f			2
das.ic1080_2hz	ground_speed_fms	ground speed - FMS (Left & Right Displayed FMS) (IC1080 1)	FLOAT4	knots	%f			2
das.ic1080_2hz	ground_speed_unfiltered	ground speed unfiltered	FLOAT4	knots	%f			2
das.ic1080_2hz	lat_fms_1	latitude - FMS 1 (geodetic) (Left Displayed FMS1) (IC1080 1)	FLOAT4	degrees	%f			2
das.ic1080_2hz	lat_fms_1_sampletime	latitude - FMS 1 sampletime (IC1080 1)	FLOAT8	seconds	%.3ymdz			2
das.ic1080_2hz	lat_fms_2	latitude - FMS 2 (geodetic) (Right Displayed FMS2) (IC1080 1)	FLOAT4	degrees	%f			2
das.ic1080_2hz	lat_fms_2_sampletime	latitude - FMS 2 sampletime (IC1080 1)	FLOAT8	seconds	%.3ymdz			2
das.ic1080_2hz	lon_fms_1	longitude - FMS 1 (geodetic) (Left Displayed FMS1) (IC1080 1): West is negative and East is positive with respect to the Prime Meridian	FLOAT4	degrees	%f			2
das.ic1080_2hz	lon_fms_1_sampletime	longitude - FMS 1 sampletime (IC1080 1)	FLOAT8	seconds	%.3ymdz			2
das.ic1080_2hz	lon_fms_2	longitude - FMS 2 (geodetic) (Right Displayed FMS2) (IC1080 1): West is negative and East is positive with respect to the Prime Meridian	FLOAT4	degrees	%f			2
das.ic1080_2hz	lon_fms_2_sampletime	longitude - FMS 2 sampletime (IC1080 1)	FLOAT8	seconds	%.3ymdz			2
das.ic1080_2hz	north_velocity_filtered	north velocity filtered	FLOAT4	knots	%f			2
das.ic1080_2hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			2
das.ic1080_2hz	static_air_temp	static air temperature (Left Displayed ADC) (IC1080 1)	FLOAT4	degrees Celsius	%f			2
das.ic1080_2hz	true_heading	true heading (Left Displayed ATT/HDG Source) (IC1080 1)	FLOAT4	degrees	%f			2
das.ic1080_2hz	true_track_angle	true track angle (IC1080 1)	FLOAT4	degrees	%f			2
das.ic1080_2hz	true_track_angle_unfiltered	true track angle unfiltered	FLOAT4	degrees	%f			2
das.ic1080_5hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			5
das.ic1080_5hz	roll	roll (IC1080 1)	FLOAT4	degrees	%f			5
das.ic1080_5hz	roll_sampletime	roll sampletime (IC1080 1)	FLOAT8	seconds	%.3ymdz			5
das.info	message	message (Informational)	STRING	N/A	%s			event
das.info	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			event
das.ins_1_12hz	east_west_vel	east-west velocity(INS 1)	FLOAT4	knots	%f			1
das.ins_1_12hz	hybrid_east_west_vel	hybrid east-west velocity (INS 1)	FLOAT4	knots	%f			1
das.ins_1_12hz	hybrid_lat	hybrid latitude (INS 1)	FLOAT8	degrees	%.13lf			1
das.ins_1_12hz	hybrid_lat_sampletime	sampletime (hybrid latitude timestamp - INS 1)	FLOAT8	seconds	%.3ymdz			1

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
das.ins_1_12hz	hybrid_lon	hybrid longitude (INS 1)	FLOAT8	degrees	%.13lf			1
das.ins_1_12hz	hybrid_lon_sampletime	sampletime (hybrid longitude timestamp - INS 1)	FLOAT8	seconds	%.3ymdz			1
das.ins_1_12hz	hybrid_north_south_vel	hybrid north-south velocity (INS 1)	FLOAT4	knots	%f			1
das.ins_1_12hz	lat	latitude (geodetic) (INS 1)	FLOAT4	degrees	%f			1
das.ins_1_12hz	lon	longitude (geodetic) (INS 1)	FLOAT4	degrees	%f			1
das.ins_1_12hz	north_south_vel	north-south velocity (INS 1)	FLOAT4	knots	%f			1
das.ins_1_12hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.ins_1_12hz	wind_angle	wind angle (INS 1)	FLOAT4	degrees	%f			1
das.ins_1_12hz	wind_speed	wind speed (INS 1)	FLOAT4	knots	%f			1
das.ins_1_1hz	hybrid_hor_fom	hybrid horizontal fom (INS 1)	FLOAT4	nautical miles	%f			1
das.ins_1_1hz	hybrid_vert_fom	hybrid vertical fom (INS 1)	FLOAT4	feet	%f			1
das.ins_1_1hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.ins_1_25hz	drift_angle	drift angle (INS 1)	FLOAT4	degrees	%f			1
das.ins_1_25hz	geodetic_alt	geodetic altitude (INS 1)	FLOAT4	feet	%f			1
das.ins_1_25hz	hybrid_alt	hybrid altitude (INS 1)	FLOAT4	feet	%f			1
das.ins_1_25hz	hybrid_alt_sampletime	sampletime (hybrid altitude timestamp - INS 1)	FLOAT8	seconds	%.3ymdz			1
das.ins_1_25hz	hybrid_flight_path_angle	hybrid flight path angle (INS 1)	FLOAT4	degrees	%f			1
das.ins_1_25hz	hybrid_ground_speed	hybrid ground speed (INS 1)	FLOAT4	knots	%f			1
das.ins_1_25hz	hybrid_true_heading	hybrid true heading (INS 1)	FLOAT4	degrees	%f			1
das.ins_1_25hz	hybrid_true_track_angle	hybrid true track angle (INS 1)	FLOAT4	degrees	%f			1
das.ins_1_25hz	hybrid_vert_vel	hybrid vertical velocity (INS 1)	FLOAT4	feet per minute	%f			1
das.ins_1_25hz	mag_heading	magnetic heading (INS 1)	FLOAT4	degrees	%f			1
das.ins_1_25hz	mag_track_angle	magnetic track angle (INS 1)	FLOAT4	degrees	%f			1
das.ins_1_25hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.ins_1_25hz	true_heading	true heading (INS 1)	FLOAT4	degrees	%f			1
das.ins_1_25hz	true_track_angle	true track angle (INS 1)	FLOAT4	degrees	%f			1
das.ins_1_50hz	inertial_vert_speed	inertial vertical speed (INS 1)	FLOAT4	feet per minute	%f			1
das.ins_1_50hz	lat_accel	lateral acceleration (INS 1)	FLOAT4	gravities	%f			1
das.ins_1_50hz	lon_accel	longitudinal acceleration (INS 1)	FLOAT4	gravities	%f			1
das.ins_1_50hz	norm_accel	normal acceleration (INS 1)	FLOAT4	gravities	%f			1
das.ins_1_50hz	pitch	pitch (INS 1)	FLOAT4	degrees	%f			1
das.ins_1_50hz	pitch_rate	pitch rate (INS 1)	FLOAT4	degrees per second	%f			1
das.ins_1_50hz	pitch_sampletime	sampletime (pitch timestamp - INS 1)	FLOAT8	seconds	%.3ymdz			1
das.ins_1_50hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.ins_1_50hz	roll	roll (INS 1)	FLOAT4	degrees	%f			1
das.ins_1_50hz	roll_rate	roll rate (INS 1)	FLOAT4	degrees per second	%f			1
das.ins_1_50hz	roll_sampletime	sampletime (roll timestamp - INS 1)	FLOAT8	seconds	%.3ymdz			1
das.ins_1_50hz	yaw_rate	yaw rate (INS 1)	FLOAT4	degrees per second	%f			1

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
das.ins_2_12hz	east_west_vel	east-west velocity(INS 2)	FLOAT4	knots	%f			1
das.ins_2_12hz	hybrid_east_west_vel	hybrid east-west velocity (INS 2)	FLOAT4	knots	%f			1
das.ins_2_12hz	hybrid_lat	hybrid latitude (INS 2)	FLOAT8	degrees	%.13lf			1
das.ins_2_12hz	hybrid_lat_sampletime	sampletime (hybrid latitude timestamp - INS 2)	FLOAT8	seconds	%.3ymdz			1
das.ins_2_12hz	hybrid_lon	hybrid longitude (INS 2)	FLOAT8	degrees	%.13lf			1
das.ins_2_12hz	hybrid_lon_sampletime	sampletime (hybrid longitude timestamp - INS 2)	FLOAT8	seconds	%.3ymdz			1
das.ins_2_12hz	hybrid_north_south_vel	hybrid north-south velocity (INS 2)	FLOAT4	knots	%f			1
das.ins_2_12hz	lat	latitude (geodetic) (INS 2)	FLOAT4	degrees	%f			1
das.ins_2_12hz	lon	longitude (geodetic) (INS 2)	FLOAT4	degrees	%f			1
das.ins_2_12hz	north_south_vel	north-south velocity (INS 2)	FLOAT4	knots	%f			1
das.ins_2_12hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.ins_2_12hz	wind_angle	wind angle (INS 2)	FLOAT4	degrees	%f			1
das.ins_2_12hz	wind_speed	wind speed (INS 2)	FLOAT4	knots	%f			1
das.ins_2_1hz	hybrid_hor_fom	hybrid horizontal fom (INS 2)	FLOAT4	nautical miles	%f			1
das.ins_2_1hz	hybrid_vert_fom	hybrid vertical fom (INS 2)	FLOAT4	feet	%f			1
das.ins_2_1hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.ins_2_25hz	drift_angle	drift angle (INS 2)	FLOAT4	degrees	%f			1
das.ins_2_25hz	geodetic_alt	geodetic altitude (INS 2)	FLOAT4	feet	%f			1
das.ins_2_25hz	hybrid_alt	hybrid altitude (INS 2)	FLOAT4	feet	%f			1
das.ins_2_25hz	hybrid_alt_sampletime	sampletime (hybrid altitude timestamp - INS 2)	FLOAT8	seconds	%.3ymdz			1
das.ins_2_25hz	hybrid_flight_path_angle	hybrid flight path angle (INS 2)	FLOAT4	degrees	%f			1
das.ins_2_25hz	hybrid_ground_speed	hybrid ground speed (INS 2)	FLOAT4	knots	%f			1
das.ins_2_25hz	hybrid_true_heading	hybrid true heading (INS 2)	FLOAT4	degrees	%f			1
das.ins_2_25hz	hybrid_true_track_angle	hybrid true track angle (INS 2)	FLOAT4	degrees	%f			1
das.ins_2_25hz	hybrid_vert_vel	hybrid vertical velocity (INS 2)	FLOAT4	feet per minute	%f			1
das.ins_2_25hz	mag_heading	magnetic heading (INS 2)	FLOAT4	degrees	%f			1
das.ins_2_25hz	mag_track_angle	magnetic track angle (INS 2)	FLOAT4	degrees	%f			1
das.ins_2_25hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.ins_2_25hz	true_heading	true heading (INS 2)	FLOAT4	degrees	%f			1
das.ins_2_25hz	true_track_angle	true track angle (INS 2)	FLOAT4	degrees	%f			1
das.ins_2_50hz	inertial_vert_speed	inertial vertical speed (INS 2)	FLOAT4	feet per minute	%f			1
das.ins_2_50hz	lat_accel	lateral acceleration (INS 2)	FLOAT4	gravities	%f			1
das.ins_2_50hz	lon_accel	longitudinal acceleration (INS 2)	FLOAT4	gravities	%f			1
das.ins_2_50hz	norm_accel	normal acceleration (INS 2)	FLOAT4	gravities	%f			1
das.ins_2_50hz	pitch	pitch (INS 2)	FLOAT4	degrees	%f			1
das.ins_2_50hz	pitch_rate	pitch rate (INS 2)	FLOAT4	degrees per second	%f			1
das.ins_2_50hz	pitch_sampletime	sampletime (pitch timestamp - INS 2)	FLOAT8	seconds	%.3ymdz			1
das.ins_2_50hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.ins_2_50hz	roll	roll (INS 2)	FLOAT4	degrees	%f			1

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
das.ins_2_50hz	roll_rate	roll rate (INS 2)	FLOAT4	degrees per second	%f			1
das.ins_2_50hz	roll_sampletime	sampletime (roll timestamp - INS 2)	FLOAT8	seconds	%.3ymdz			1
das.ins_2_50hz	yaw_rate	yaw rate (INS 2)	FLOAT4	degrees per second	%f			1
das.ins_3_12hz	east_west_vel	east-west velocity(INS 3)	FLOAT4	knots	%f			1
das.ins_3_12hz	hybrid_east_west_vel	hybrid east-west velocity (INS 3)	FLOAT4	knots	%f			1
das.ins_3_12hz	hybrid_lat	hybrid latitude (INS 3)	FLOAT8	degrees	%.13lf			1
das.ins_3_12hz	hybrid_lat_sampletime	sampletime (hybrid latitude timestamp - INS 3)	FLOAT8	seconds	%.3ymdz			1
das.ins_3_12hz	hybrid_lon	hybrid longitude (INS 3)	FLOAT8	degrees	%.13lf			1
das.ins_3_12hz	hybrid_lon_sampletime	sampletime (hybrid longitude timestamp - INS 3)	FLOAT8	seconds	%.3ymdz			1
das.ins_3_12hz	hybrid_north_south_vel	hybrid north-south velocity (INS 3)	FLOAT4	knots	%f			1
das.ins_3_12hz	lat	latitude (geodetic) (INS 3)	FLOAT4	degrees	%f			1
das.ins_3_12hz	lon	longitude (geodetic) (INS 3)	FLOAT4	degrees	%f			1
das.ins_3_12hz	north_south_vel	north-south velocity (INS 3)	FLOAT4	knots	%f			1
das.ins_3_12hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.ins_3_12hz	wind_angle	wind angle (INS 3)	FLOAT4	degrees	%f			1
das.ins_3_12hz	wind_speed	wind speed (INS 3)	FLOAT4	knots	%f			1
das.ins_3_1hz	hybrid_hor_fom	hybrid horizontal fom (INS 3)	FLOAT4	nautical miles	%f			1
das.ins_3_1hz	hybrid_vert_fom	hybrid vertical fom (INS 3)	FLOAT4	feet	%f			1
das.ins_3_1hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.ins_3_25hz	drift_angle	drift angle (INS 3)	FLOAT4	degrees	%f			1
das.ins_3_25hz	geodetic_alt	geodetic altitude (INS 3)	FLOAT4	feet	%f			1
das.ins_3_25hz	hybrid_alt	hybrid altitude (INS 3)	FLOAT4	feet	%f			1
das.ins_3_25hz	hybrid_alt_sampletime	sampletime (hybrid altitude timestamp - INS 3)	FLOAT8	seconds	%.3ymdz			1
das.ins_3_25hz	hybrid_flight_path_angle	hybrid flight path angle (INS 3)	FLOAT4	degrees	%f			1
das.ins_3_25hz	hybrid_ground_speed	hybrid ground speed (INS 3)	FLOAT4	knots	%f			1
das.ins_3_25hz	hybrid_true_heading	hybrid true heading (INS 3)	FLOAT4	degrees	%f			1
das.ins_3_25hz	hybrid_true_track_angle	hybrid true track angle (INS 3)	FLOAT4	degrees	%f			1
das.ins_3_25hz	hybrid_vert_vel	hybrid vertical velocity (INS 1)	FLOAT4	feet per minute	%f			1
das.ins_3_25hz	mag_heading	magnetic heading (INS 3)	FLOAT4	degrees	%f			1
das.ins_3_25hz	mag_track_angle	magnetic track angle (INS 3)	FLOAT4	degrees	%f			1
das.ins_3_25hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.ins_3_25hz	true_heading	true heading (INS 3)	FLOAT4	degrees	%f			1
das.ins_3_25hz	true_track_angle	true track angle (INS 3)	FLOAT4	degrees	%f			1
das.ins_3_50hz	inertial_vert_speed	inertial vertical speed (INS 3)	FLOAT4	feet per minute	%f			1
das.ins_3_50hz	lat_accel	lateral acceleration (INS 3)	FLOAT4	gravities	%f			1
das.ins_3_50hz	lon_accel	longitudinal acceleration (INS 3)	FLOAT4	gravities	%f			1
das.ins_3_50hz	norm_accel	normal acceleration (INS 3)	FLOAT4	gravities	%f			1
das.ins_3_50hz	pitch	pitch (INS 3)	FLOAT4	degrees	%f			1

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
das.ins_3_50hz	pitch_rate	pitch rate (INS 3)	FLOAT4	degrees per second	%f			1
das.ins_3_50hz	pitch_sampletime	sampletime (pitch timestamp - INS 3)	FLOAT8	seconds	%.3ymdz			1
das.ins_3_50hz	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.ins_3_50hz	roll	roll (INS 3)	FLOAT4	degrees	%f			1
das.ins_3_50hz	roll_rate	roll rate (INS 3)	FLOAT4	degrees per second	%f			1
das.ins_3_50hz	roll_sampletime	sampletime (roll timestamp - INS 3)	FLOAT8	seconds	%.3ymdz			1
das.ins_3_50hz	yaw_rate	yaw rate (INS 3)	FLOAT4	degrees per second	%f			1
das.nasmyth_env	nasmyth_humidity	Nasmyth Humidity (close to SI as possible)	FLOAT4	percent	%f			1
das.nasmyth_env	nasmyth_tube_entrance_temp	Nasmyth tube entrance temperature	FLOAT4	degrees Fahrenheit	%f			1
das.nasmyth_env	nasmyth_tube_temp	Nasmyth tube temperature (all the way up)	FLOAT4	degrees Fahrenheit	%f			1
das.nasmyth_env	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.nasmyth_micro	nasmyth_micro_1	Nasmyth Microphone #1	BINARRAY, [Number of Element: 20, Representation: FLOAT4]	gravities	%f			50
das.nasmyth_micro	nasmyth_micro_2	Nasmyth Microphone #2	BINARRAY, [Number of Element: 20, Representation: FLOAT4]	gravities	%f			50
das.nasmyth_micro	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			50
das.pbit_results	last_result_test_1	last result - Test 1 (Clock), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_10	last result - Test 10 (MV64360 Enet2), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_11	last result - Test 11 (MV64360 MPSC1), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_12	last result - Test 12 (MV64360 MPSC2), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_13	last result - Test 13 (MV64360 SRAM), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_14	last result - Test 14 (MV64360 DMA), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_15	last result - Test 15 (MV64360 Timer), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_16	last result - Test 16 (MV64360 PIC), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_17	last result - Test 17 (SDRAM), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_18	last result - Test 18 (FLASH A), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
das.pbit_results	last_result_test_19	last result - Test 19 (FLASH B), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_2	last result - Test 2 (CPU), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_20	last result - Test 20 (FPGA), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_21	last result - Test 21 (FPGA DPRAM), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_22	last result - Test 22 (FPGA GPIO), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_23	last result - Test 23 (FPGA WD), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_24	last result - Test 24 (FPGA PIC), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_25	last result - Test 25 (PCI), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_26	last result - Test 26 (RTC), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_27	last result - Test 27 (UART A), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_28	last result - Test 28 (UART B), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_29	last result - Test 29 (PMC 1), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_3	last result - Test 3 (L1 Inst Cache), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_30	last result - Test 30 (PMC 2), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_31	last result - Test 31 (CPU Temp), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_32	last result - Test 32 (Board Temp), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_33	last result - Test 33 (NV resistor), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_34	last result - Test 34 (SPD EEPROM), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_35	last result - Test 35 (User EEPROM), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_36	last result - Test 36 (Chipset EEPROM), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_37	last result - Test 37 (VME64), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_38	last result - Test 38 (SATA), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_39	last result - Test 39 (ADC), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_4	last result - Test 4 (L1 Data Cache), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_40	last result - Test 40 (USER 1), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_41	last result - Test 41 (USER 2), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
das.pbit_results	last_result_test_42	last result - Test 42 (USER 3), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_43	last result - Test 43 (USER 4), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_5	last result - Test 5 (L2 Cache), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_6	last result - Test 6 (L3 Cache), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_7	last result - Test 7 (MV64360), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_8	last result - Test 8 (MV64360 TWSI), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	last_result_test_9	last result - Test 9 (MV64360 Enet1), [ENUM: (0: PASS) (1: FAIL) (254: NOT_IMPLEMENTED) (253: NOT_RUN) (252: SKIPPED)]	UINT1	N/A	%d			on connect
das.pbit_results	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			on connect
das.pbit_results	total_errors_test_1	total errors - Test 1 (Clock)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_10	total errors - Test 10 (MV64360 Enet2)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_11	total errors - Test 11 (MV64360 MPSC1)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_12	total errors - Test 12 (MV64360 MPSC2)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_13	total errors - Test 13 (MV64360 SRAM)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_14	total errors - Test 14 (MV64360 DMA)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_15	total errors - Test 15 (MV64360 Timer)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_16	total errors - Test 16 (MV64360 PIC)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_17	total errors - Test 17 (SDRAM)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_18	total errors - Test 18 (FLASH A)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_19	total errors - Test 19 (FLASH B)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_2	total errors - Test 2 (CPU)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_20	total errors - Test 20 (FPGA)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_21	total errors - Test 21 (FPGA DPRAM)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_22	total errors - Test 22 (FPGA GPIO)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_23	total errors - Test 23 (FPGA WD)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_24	total errors - Test 24 (FPGA PIC)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_25	total errors - Test 25 (PCI)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_26	total errors - Test 26 (RTC)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_27	total errors - Test 27 (UART A)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_28	total errors - Test 28 (UART B)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_29	total errors - Test 29 (PMC 1)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_3	total errors - Test 3 (L1 Inst Cache)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_30	total errors - Test 30 (PMC 2)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_31	total errors - Test 31 (CPU Temp)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_32	total errors - Test 32 (Board Temp)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_33	total errors - Test 33 (NV resistor)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_34	total errors - Test 34 (SPD EEPROM)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_35	total errors - Test 35 (User EEPROM)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_36	total errors - Test 36 (Chipset EEPROM)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_37	total errors - Test 37 (VME64)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_38	total errors - Test 38 (SATA)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_39	total errors - Test 39 (ADC)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_4	total errors - Test 4 (L1 Data Cache)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_40	total errors - Test 40 (USER 1)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_41	total errors - Test 41 (USER 2)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_42	total errors - Test 42 (USER 3)	UINT4	N/A	%d			on connect

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
das.pbit_results	total_errors_test_43	total errors - Test 43 (USER 4)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_5	total errors - Test 5 (L2 Cache)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_6	total errors - Test 6 (L3 Cache)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_7	total errors - Test 7 (MV64360)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_8	total errors - Test 8 (MV64360 TWSI)	UINT4	N/A	%d			on connect
das.pbit_results	total_errors_test_9	total errors - Test 9 (MV64360 Enet1)	UINT4	N/A	%d			on connect
das.pbit_results	total_stores	total stores	UINT4	N/A	%d			on connect
das.pds	ac_obs_bus_1_power_draw	AC Observatory Bus 1 power draw	FLOAT4	kilo volt amperes	%f			1
das.pds	ac_obs_bus_2_power_draw	AC Observatory Bus 2 power draw	FLOAT4	kilo volt amperes	%f			1
das.pds	ac_obs_bus_3_power_draw	AC Observatory Bus 3 power draw	FLOAT4	kilo volt amperes	%f			1
das.pds	ac_obs_bus_4_power_draw	AC Observatory Bus 4 power draw	FLOAT4	kilo volt amperes	%f			1
das.pds	dc_current_power_bus_1	DC Current Power Bus 1	FLOAT4	amperes dc	%f			1
das.pds	dc_current_power_bus_2	DC Current Power Bus 2	FLOAT4	amperes dc	%f			1
das.pds	dc_current_power_bus_3	DC Current Power Bus 3	FLOAT4	amperes dc	%f			1
das.pds	dc_current_power_bus_3_to_si	DC Current Power Bus 3 to SI	FLOAT4	amperes dc	%f			1
das.pds	dc_current_power_bus_4	DC Current Power Bus 4	FLOAT4	amperes dc	%f			1
das.pds	dc_obs_bus_1_power_draw	DC Observatory Bus 1 power draw	FLOAT4	kilowatts	%f			1
das.pds	dc_obs_bus_2_power_draw	DC Observatory Bus 2 power draw	FLOAT4	kilowatts	%f			1
das.pds	dc_obs_bus_3_power_draw	DC Observatory Bus 3 power draw	FLOAT4	kilowatts	%f			1
das.pds	dc_obs_bus_4_power_draw	DC Observatory Bus 4 power draw	FLOAT4	kilowatts	%f			1
das.pds	dc_voltage_power_bus_1	DC Voltage Power Bus 1	FLOAT4	volts dc	%f			1
das.pds	dc_voltage_power_bus_2	DC Voltage Power Bus 2	FLOAT4	volts dc	%f			1
das.pds	dc_voltage_power_bus_3	DC Voltage Power Bus 3	FLOAT4	volts dc	%f			1
das.pds	dc_voltage_power_bus_3_to_si	DC Voltage Power Bus 3 to SI	FLOAT4	volts dc	%f			1
das.pds	dc_voltage_power_bus_4	DC Voltage Power Bus 4	FLOAT4	volts dc	%f			1
das.pds	freq_conv_1_rms_current	Frequency Converter 1 RMS Current	FLOAT4	amperes rms	%f			1
das.pds	freq_conv_1_rms_voltage	Frequency Converter 1 RMS Voltage	FLOAT4	volts rms	%f			1
das.pds	freq_conv_2_rms_current	Frequency Converter 2 RMS Current	FLOAT4	amperes rms	%f			1
das.pds	freq_conv_2_rms_voltage	Frequency Converter 2 RMS Voltage	FLOAT4	volts rms	%f			1
das.pds	freq_conv_3_rms_current	Frequency Converter 3 RMS Current	FLOAT4	amperes rms	%f			1
das.pds	freq_conv_3_rms_voltage	Frequency Converter 3 RMS Voltage	FLOAT4	volts rms	%f			1
das.pds	freq_conv_4_rms_current	Frequency Converter 4 RMS Current	FLOAT4	amperes rms	%f			1
das.pds	freq_conv_4_rms_voltage	Frequency Converter 4 RMS Voltage	FLOAT4	volts rms	%f			1
das.pds	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.pds	power_bus_1_rms_current_phase_a	Power Bus 1 RMS Current Phase A	FLOAT4	amperes rms	%f			1
das.pds	power_bus_1_rms_current_phase_b	Power Bus 1 RMS Current Phase B	FLOAT4	amperes rms	%f			1

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
das.pds	power_bus_1_rms_current_phase_c	Power Bus 1 RMS Current Phase C	FLOAT4	amperes rms	%f			1
das.pds	power_bus_1_rms_voltage_phase_a	Power Bus 1 RMS Voltage Phase A	FLOAT4	volts rms	%f			1
das.pds	power_bus_1_rms_voltage_phase_b	Power Bus 1 RMS Voltage Phase B	FLOAT4	volts rms	%f			1
das.pds	power_bus_1_rms_voltage_phase_c	Power Bus 1 RMS Voltage Phase C	FLOAT4	volts rms	%f			1
das.pds	power_bus_2_rms_current_phase_a	Power Bus 2 RMS Current Phase A	FLOAT4	amperes rms	%f			1
das.pds	power_bus_2_rms_current_phase_b	Power Bus 2 RMS Current Phase B	FLOAT4	amperes rms	%f			1
das.pds	power_bus_2_rms_current_phase_c	Power Bus 2 RMS Current Phase C	FLOAT4	amperes rms	%f			1
das.pds	power_bus_2_rms_voltage_phase_a	Power Bus 2 RMS Voltage Phase A	FLOAT4	volts rms	%f			1
das.pds	power_bus_2_rms_voltage_phase_b	Power Bus 2 RMS Voltage Phase B	FLOAT4	volts rms	%f			1
das.pds	power_bus_2_rms_voltage_phase_c	Power Bus 2 RMS Voltage Phase C	FLOAT4	volts rms	%f			1
das.pds	power_bus_3_rms_current_phase_a	Power Bus 3 RMS Current Phase A	FLOAT4	amperes rms	%f			1
das.pds	power_bus_3_rms_current_phase_b	Power Bus 3 RMS Current Phase B	FLOAT4	amperes rms	%f			1
das.pds	power_bus_3_rms_current_phase_c	Power Bus 3 RMS Current Phase C	FLOAT4	amperes rms	%f			1
das.pds	power_bus_3_rms_voltage_phase_a	Power Bus 3 RMS Voltage Phase A	FLOAT4	volts rms	%f			1
das.pds	power_bus_3_rms_voltage_phase_b	Power Bus 3 RMS Voltage Phase B	FLOAT4	volts rms	%f			1
das.pds	power_bus_3_rms_voltage_phase_c	Power Bus 3 RMS Voltage Phase C	FLOAT4	volts rms	%f			1
das.pds	power_bus_4_rms_current_phase_a	Power Bus 4 RMS Current Phase A	FLOAT4	amperes rms	%f			1
das.pds	power_bus_4_rms_current_phase_b	Power Bus 4 RMS Current Phase B	FLOAT4	amperes rms	%f			1
das.pds	power_bus_4_rms_current_phase_c	Power Bus 4 RMS Current Phase C	FLOAT4	amperes rms	%f			1
das.pds	power_bus_4_rms_voltage_phase_a	Power Bus 4 RMS Voltage Phase A	FLOAT4	volts rms	%f			1
das.pds	power_bus_4_rms_voltage_phase_b	Power Bus 4 RMS Voltage Phase B	FLOAT4	volts rms	%f			1
das.pds	power_bus_4_rms_voltage_phase_c	Power Bus 4 RMS Voltage Phase C	FLOAT4	volts rms	%f			1
das.pds	rms_current_for_cryo_comp_1	RMS Current for cryocooler compressor #1	FLOAT4	amperes rms	%f			1
das.pds	rms_current_for_cryo_comp_2	RMS Current for cryocooler compressor #2	FLOAT4	amperes rms	%f			1
das.pds	rms_current_for_vps_pump_1	RMS Current for VPS Pump #1	FLOAT4	amperes rms	%f			1
das.pds	rms_current_for_vps_pump_2	RMS Current for VPS Pump #2	FLOAT4	amperes rms	%f			1
das.pds	rms_voltage_for_cryo_comp_1	RMS Voltage for cryocooler compressor #1	FLOAT4	volts rms	%f			1
das.pds	rms_voltage_for_cryo_comp_2	RMS Voltage for cryocooler compressor #2	FLOAT4	volts rms	%f			1
das.pds	rms_voltage_for_vps_pump_1	RMS Voltage for VPS Pump #1	FLOAT4	volts rms	%f			1
das.pds	rms_voltage_for_vps_pump_2	RMS Voltage for VPS Pump #2	FLOAT4	volts rms	%f			1
das.pds	total_obs_power_draw	Total Observatory power draw	FLOAT4	kilo volt amperes	%f			1
das.pds	ups_a_logic_fault	UPS A - Logic Fault	BYTE	discrete	%d			1
das.pds	ups_a_low_battery	UPS A - Low Battery	BYTE	discrete	%d			1
das.pds	ups_a_over_temp	UPS A - Over Temperature	BYTE	discrete	%d			1
das.pds	ups_a_overload	UPS A - Overload	BYTE	discrete	%d			1
das.pds	ups_a_rms_current	UPS A RMS Current	FLOAT4	amperes rms	%f			1
das.pds	ups_a_rms_voltage	UPS A RMS Voltage	FLOAT4	volts rms	%f			1

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
das.pds	ups_a_ups_off	UPS A - UPS Off	BYTE	discrete	%d			1
das.pds	ups_a_utility_off	UPS A - Utility Off	BYTE	discrete	%d			1
das.pds	ups_b_logic_fault	UPS B - Logic Fault	BYTE	discrete	%d			1
das.pds	ups_b_low_battery	UPS B - Low Battery	BYTE	discrete	%d			1
das.pds	ups_b_over_temp	UPS B - Over Temperature	BYTE	discrete	%d			1
das.pds	ups_b_overload	UPS B - Overload	BYTE	discrete	%d			1
das.pds	ups_b_rms_current	UPS B RMS Current	FLOAT4	amperes rms	%f			1
das.pds	ups_b_rms_voltage	UPS B RMS Voltage	FLOAT4	volts rms	%f			1
das.pds	ups_b_ups_off	UPS B - UPS Off	BYTE	discrete	%d			1
das.pds	ups_b_utility_off	UPS B - Utility Off	BYTE	discrete	%d			1
das.pds	ups_c_logic_fault	UPS C - Logic Fault	BYTE	discrete	%d			1
das.pds	ups_c_low_battery	UPS C - Low Battery	BYTE	discrete	%d			1
das.pds	ups_c_over_temp	UPS C - Over Temperature	BYTE	discrete	%d			1
das.pds	ups_c_overload	UPS C - Overload	BYTE	discrete	%d			1
das.pds	ups_c_rms_current	UPS C RMS Current	FLOAT4	amperes rms	%f			1
das.pds	ups_c_rms_voltage	UPS C RMS Voltage	FLOAT4	volts rms	%f			1
das.pds	ups_c_ups_off	UPS C - UPS Off	BYTE	discrete	%d			1
das.pds	ups_c_utility_off	UPS C - Utility Off	BYTE	discrete	%d			1
das.pds	ups_d_rms_current	UPS D RMS Current	FLOAT4	amperes rms	%f			1
das.pds	ups_d_rms_voltage	UPS D RMS Voltage	FLOAT4	volts rms	%f			1
das.rd_cfg_file_resp	information	information (for read_config_file command response)	STRING	N/A	%s			event
das.rd_cfg_file_resp	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			event
das.rd_cfg_file_resp	response	response (for read_config_file command), [ENUM: (1: COMMAND_ACKNOWLEDGED) (2: COMMAND_INITIATED) (3: COMMAND_COMPLETED) (4: COMMAND_REJECTED_INVALID_ARGUMENT) (5: COMMAND_REJECTED_INVALID_HEADER_SYNC) (6: COMMAND_REJECTED_INVALID_HEADER_SIZE) (7: COMMAND_REJECTED_INVALID_HEADER_MSGID) (8: COMMAND_REJECTED_INVALID_HEADER_PKTTYPE) (100: FILE_WRITE_FAILURE) (101: FILE_READ_FAILURE) (102: FILE_PARSE_FAILURE) (103: INVALID_GROUP) (254: COMMAND_ABORT_FAILURE) (255: COMMAND_ABORT_SUCCESS)]	UINT1	N/A	%d			event
das.reboot_resp	information	information (for reboot command response)	STRING	N/A	%s			event
das.reboot_resp	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			event
das.reboot_resp	response	response (for reboot command), [ENUM: (1: COMMAND_ACKNOWLEDGED) (2: COMMAND_INITIATED) (3: COMMAND_COMPLETED) (4: COMMAND_REJECTED_INVALID_ARGUMENT) (5: COMMAND_REJECTED_INVALID_HEADER_SYNC) (6: COMMAND_REJECTED_INVALID_HEADER_SIZE) (7: COMMAND_REJECTED_INVALID_HEADER_MSGID) (8: COMMAND_REJECTED_INVALID_HEADER_PKTTYPE) (100: FILE_WRITE_FAILURE) (101: FILE_READ_FAILURE) (102: FILE_PARSE_FAILURE) (103: INVALID_GROUP) (254: COMMAND_ABORT_FAILURE) (255: COMMAND_ABORT_SUCCESS)]	UINT1	N/A	%d			event
das.shutdown_resp	information	information (for shutdown command response)	STRING	N/A	%s			event
das.shutdown_resp	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			event

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
das.shutdown_resp	response	response (for shutdown command), [ENUM: (1: COMMAND_ACKNOWLEDGED) (2: COMMAND_INITIATED) (3: COMMAND_COMPLETED) (4: COMMAND_REJECTED_INVALID_ARGUMENT) (5: COMMAND_REJECTED_INVALID_HEADER_SYNC) (6: COMMAND_REJECTED_INVALID_HEADER_SIZE) (7: COMMAND_REJECTED_INVALID_HEADER_MSGID) (8: COMMAND_REJECTED_INVALID_HEADER_PKTTYPE) (100: FILE_WRITE_FAILURE) (101: FILE_READ_FAILURE) (102: FILE_PARSE_FAILURE) (103: INVALID_GROUP) (254: COMMAND_ABORT_FAILURE) (255: COMMAND_ABORT_SUCCESS)]	UINT1	N/A	%d			event
das.si_flange_acc	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			50
das.si_flange_acc	si_mounting_flange_accel_x	SI Mounting Flange Accelerometer - X	BINARRAY, [Number of Element: 40, Representation: FLOAT4]	gravities	%f			50
das.si_flange_acc	si_mounting_flange_accel_y	SI Mounting Flange Accelerometer - Y	BINARRAY, [Number of Element: 40, Representation: FLOAT4]	gravities	%f			50
das.si_flange_acc	si_mounting_flange_accel_z	SI Mounting Flange Accelerometer - Z	BINARRAY, [Number of Element: 40, Representation: FLOAT4]	gravities	%f			50
das.si_patch	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			50
das.si_patch	si_patch_analog_signal_1	SI patch analog signal #1	BINARRAY, [Number of Element: 40, Representation: FLOAT4]	TBD	%f			50
das.si_patch	si_patch_analog_signal_2	SI patch analog signal #2	BINARRAY, [Number of Element: 40, Representation: FLOAT4]	TBD	%f			50
das.si_patch	si_patch_analog_signal_3	SI patch analog signal #3	BINARRAY, [Number of Element: 40, Representation: FLOAT4]	TBD	%f			50
das.telescope_env	ffi_aft	FFI AFT	FLOAT4	degrees Fahrenheit	%f			1
das.telescope_env	ffi_facing_aperture	FFI Facing Aperture	FLOAT4	degrees Fahrenheit	%f			1
das.telescope_env	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.telescope_env	primary_mirror_backside_edge_center	Primary mirror backside edge center (C10)	FLOAT4	degrees Fahrenheit	%f			1
das.telescope_env	primary_mirror_backside_edge_fwd	Primary mirror backside edge fwd (L2)	FLOAT4	degrees Fahrenheit	%f			1

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
das.telescope_env	primary_mirror_backside_edge_left	Primary mirror backside edge left	FLOAT4	degrees Fahrenheit	%f			1
das.telescope_env	primary_mirror_backside_edge_right	Primary mirror backside edge right (J7)	FLOAT4	degrees Fahrenheit	%f			1
das.telescope_env	secondary_mirror_temp	secondary mirror temperature (below aft spider)	FLOAT4	degrees Fahrenheit	%f			1
das.telescope_env	spider_temp_1	spider temperature - 1	FLOAT4	degrees Fahrenheit	%f			1
das.telescope_env	spider_temp_2	spider temperature - 2	FLOAT4	degrees Fahrenheit	%f			1
das.telescope_env	spider_temps_3	spider temperatures - 3	FLOAT4	degrees Fahrenheit	%f			1
das.telescope_env	spider_temps_4	spider temperatures - 4	FLOAT4	degrees Fahrenheit	%f			1
das.telescope_env	tertiary_mirror_temp_aft	tertiary mirror temperature - Aft	FLOAT4	degrees Fahrenheit	%f			1
das.telescope_env	tertiary_mirror_temp_fwd	tertiary mirror temperature - Fwd	FLOAT4	degrees Fahrenheit	%f			1
das.telescope_env	vis_air_bottles_press	VIS air bottles pressure	FLOAT4	pounds per square inch absolute	%f			1
das.telescope_env	wfi_aft	WFI AFT	FLOAT4	degrees Fahrenheit	%f			1
das.telescope_env	wfi_facing_aperture	WFI Facing Aperture	FLOAT4	degrees Fahrenheit	%f			1
das.vacuum_pump	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			1
das.vacuum_pump	vac_pump_1_flow_rate	Vacuum Pump 1 Flow Rate	FLOAT4	N/A	%f			1
das.vacuum_pump	vac_pump_1_press	Vacuum Pump 1 Pressure	FLOAT4	torrs	%f			1
das.vacuum_pump	vac_pump_2_flow_rate	Vacuum Pump 2 Flow Rate	FLOAT4	N/A	%f			1
das.vacuum_pump	vac_pump_2_press	Vacuum Pump 2 Pressure	FLOAT4	torrs	%f			1
das.vacuum_pump	vac_pump_3_press	Vacuum Pump 3 Pressure	FLOAT4	torrs	%f			1
das.version	build_indicator	Software build indicator	UINT1	N/A	%d			on connect
das.version	checksums_valid	Pass or fail of all checksums matching expected values	UINT1	N/A	%d			on connect
das.version	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			on connect
das.version	rev	Software revision	UINT1	N/A	%d			on connect
das.version	subrev	Software sub-revision	UINT1	N/A	%d			on connect
das.warning	message	message (Warning)	STRING	N/A	%s			event
das.warning	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			event
das.wr_cfg_file_resp	information	information (for write_config_file command response)	STRING	N/A	%s			event
das.wr_cfg_file_resp	pkt_timestamp	pkt_timestamp	FLOAT8	seconds	%.3ymdz			event

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
das.wr_cfg_file_resp	response	response (for write_config_file command), [ENUM: (1: COMMAND_ACKNOWLEDGED) (2: COMMAND_INITIATED) (3: COMMAND_COMPLETED) (4: COMMAND_REJECTED_INVALID_ARGUMENT) (5: COMMAND_REJECTED_INVALID_HEADER_SYNC) (6: COMMAND_REJECTED_INVALID_HEADER_SIZE) (7: COMMAND_REJECTED_INVALID_HEADER_MSGID) (8: COMMAND_REJECTED_INVALID_HEADER_PKTTYPE) (100: FILE_WRITE_FAILURE) (101: FILE_READ_FAILURE) (102: FILE_PARSE_FAILURE) (103: INVALID_GROUP) (254: COMMAND_ABORT_FAILURE) (255: COMMAND_ABORT_SUCCESS)]	UINT1	N/A	%d			event
dither								
dither	commands	Contains the last SCL command sent to dither	STRING	N/A	%s			event
dither	current	Current dither position	STRING	N/A	%s			
dither	next	Next dither position	STRING	N/A	%s			
dither	oper_state	Operating state	INT4	N/A	0x%2.2X			
dither	rcs_id	Version information for the XML file that defines this data source	STRING	N/A	%s			event
dither	sequence	Current sequence of dither positions.	STRING	N/A	%s			
dither.archive_info	archive_file_name	Archive file name	STRING	N/A	%s			event
dither.archive_info	archive_status	[ENUM: (0: false) (1: true)]	BOOL4	N/A	%s	0	1	
dither.archive_info	current_bytes_archived	Total bytes written to the current ark file	FLOAT8	bytes	%.0lf	0		
dither.archive_info	total_bytes_archived	Total bytes written since this data source starts	FLOAT8	bytes	%.0lf	0		
dither.archive_nsamples	current_bytes_archived_nsample		INT4	N/A	%d	0		event
dither.archive_nsamples	total_bytes_archived_nsample		INT4	N/A	%d	0		event
dither.dither_alarm	dither_alarm_error	Serious problems requiring user confirmation	STRING	N/A	%s			event
dither.dither_alarm	dither_alarm_fatal	Fatal problems requiring user confirmation	STRING	N/A	%s			event
dither.dither_alarm	dither_alarm_info	Normal events requiring user confirmation	STRING	N/A	%s			event
dither.dither_alarm	dither_alarm_warning	Abnormal events requiring user confirmation	STRING	N/A	%s			event
dither.dither_alert	dither_alert_error	Non-recoverable problems which are not fatal	STRING	N/A	%s			event
dither.dither_alert	dither_alert_fatal	Fatal problems	STRING	N/A	%s			event
dither.dither_alert	dither_alert_info	Informational messages (events)	STRING	N/A	%s			event
dither.dither_alert	dither_alert_warning	Events requiring user action for recovery	STRING	N/A	%s			event
dither.points	all_names	ASCII list of the dither names	STRING	N/A	%s			
dither.points	last_changed	Last dither position that was added or deleted	STRING	N/A	%s			
dither.points	last_distributed	Last dither point that was distributed	STRING	N/A	%s			
dither.points	number	Number of dither positions currently defined	INT4	N/A	%d	0		
dither.points.ANY	delta_dec	Delta declination for the dither point	FLOAT8	arcseconds	%lf			
dither.points.ANY	delta_el	Delta telescope elevation for the dither point	FLOAT8	arcseconds	%lf			
dither.points.ANY	delta_ra	Delta Right Ascension for the dither point	FLOAT8	sidereal seconds	%lf			
dither.points.ANY	delta_xel	Delta cross-elevation for the dither point	FLOAT8	arcseconds	%lf			
dither.points.ANY	delta_xffi	Change in FFI X-pixels for the dither point	FLOAT8	pixels	%lf			
dither.points.ANY	delta_xfpi	Change in FPI X-pixels for the dither point	FLOAT8	pixels	%lf			
dither.points.ANY	delta_xsi	Change in SI x-pixels for the dither point	FLOAT8	pixels	%lf			
dither.points.ANY	delta_xwfi	Change in WFI X-pixels for the dither point	FLOAT8	pixels	%lf			
dither.points.ANY	delta_yffi	Change in FFI Y-pixels for the dither point	FLOAT8	pixels	%lf			
dither.points.ANY	delta_yfpi	Change in FPI Y-pixels for the dither point	FLOAT8	pixels	%lf			
dither.points.ANY	delta_ysi	Change in SI y-pixels for the dither point	FLOAT8	pixels	%lf			
dither.points.ANY	delta_ywfi	Change in WFI Y-pixels for the dither point	FLOAT8	pixels	%lf			
dither.responses	define	Last SCL response sent back to the issuer of the define command	STRING	N/A	%s			event
dither.responses	goto	Last SCL response sent back to the issuer of the goto command	STRING	N/A	%s			event
dither.responses	insert	Last SCL response sent back to the issuer of the insert command	STRING	N/A	%s			event
dither.responses	next	Last SCL response sent back to the issuer of the next command	STRING	N/A	%s			event

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
dither.responses	quiet	Last SCL response sent back to the issuer of the quiet command	STRING	N/A	%s			event
dither.responses	resume	Last SCL response sent back to the issuer of the resume command	STRING	N/A	%s			event
dither.responses	set	Last SCL response sent back to the issuer of the set command	STRING	N/A	%s			event
dither.responses	suspend	Last SCL response sent back to the issuer of the suspend command	STRING	N/A	%s			event
dither.responses	verbose	Last SCL response sent back to the issuer of the verbose command	STRING	N/A	%s			event
fltexec								
fltexec	oper_state	Operating state	INT4	N/A	0x%2.2X			event
fltexec	rcs_id	Version information for the XML file that defines this data source	STRING	N/A	%s			event
fltexec.archive_info	archive_file_name	Archive file name	STRING	N/A	%s			event
fltexec.archive_info	archive_status	[ENUM: (0: false) (1: true)]	BOOL4	N/A	%s	0	1	event
fltexec.archive_info	current_bytes_archived	Total bytes written to the current ark file	FLOAT8	bytes	%.0lf	0		event
fltexec.archive_info	total_bytes_archived	Total bytes written since this data source starts	FLOAT8	bytes	%.0lf	0		event
fltexec.fltexec_alarm	fltexec_alarm_error	Serious problem requiring user confirmation	STRING	N/A	%s			event
fltexec.fltexec_alarm	fltexec_alarm_fatal	Fatal problem requiring user confirmation	STRING	N/A	%s			event
fltexec.fltexec_alarm	fltexec_alarm_info	Normal event requiring user confirmation	STRING	N/A	%s			event
fltexec.fltexec_alarm	fltexec_alarm_warning	Abnormal event requiring user confirmation	STRING	N/A	%s			event
fltexec.fltexec_alert	fltexec_alert_error	Non-recoverable problem which is not fatal	STRING	N/A	%s			event
fltexec.fltexec_alert	fltexec_alert_fatal	Fatal problem	STRING	N/A	%s			event
fltexec.fltexec_alert	fltexec_alert_info	Informational message (event)	STRING	N/A	%s			event
fltexec.fltexec_alert	fltexec_alert_warning	Warning message (may require user action for recovery)	STRING	N/A	%s			event
fltexec.fltexec_data	heading_state	Heading state [ENUM: (1: engaged) (2: released)]	UINT4	N/A	%d	1	2	event
fltexec.fltexec_data	leg_seq	Current leg sequence number, used for user display	INT4	N/A	%d			event
fltexec.fltexec_data	obj_dec	Current object declination	FLOAT8	degrees	%lf			event
fltexec.fltexec_data	obj_name	Name of the astronomical object being observed	STRING	N/A	%s			event
fltexec.fltexec_data	obj_ra	Current object Right Ascension	FLOAT8	sidereal hours	%lf			event
fltexec.fltexec_data	obj_thdg_err	True heading error	FLOAT8	degrees	%.1lf			event
fltexec.fltexec_data	planned_alt	Expected altitude	FLOAT8	feet	%.0lf			event
fltexec.fltexec_data	time_left	Time left on current observing leg	FLOAT8	seconds	%.0lf			event
nod								
nod	amp2	Size of the secondary throw (between X and B)	FLOAT8	arcseconds	%lf	0		
nod	amplitude	Size of the primary throw (between X and A)	FLOAT8	arcseconds	%lf	0		
nod	commands	Contains the last SCL command sent to nod	STRING	N/A	%s			event
nod	coord_sys	Coordinate system for pos_angle	STRING	N/A	%s			
nod	current	Current nod position	STRING	N/A	%s			
nod	next	Next nod position	STRING	N/A	%s			
nod	oper_state	Operating state	INT4	N/A	0x%2.2X			
nod	pos_angle	Angle of the nod axis	FLOAT8	degrees	%lf	-180	360	
nod	position	Which beam the telescope is in	STRING	N/A	%s			
nod	profile	Number of nod beams	INT4	N/A	%i	2	3	
nod	rcs_id	Version information for the XML file that defines this data source	STRING	N/A	%s			event
nod	track_pos_a	Name of position to use for tracking when at beam A	STRING	N/A	%s			
nod	track_pos_b	Name of position to use for tracking when at beam B	STRING	N/A	%s			
nod	track_pos_x	Name of position to use for tracking when at beam X	STRING	N/A	%s			
nod.archive_info	archive_file_name	Archive file name	STRING	N/A	%s			event
nod.archive_info	archive_status	[ENUM: (0: false) (1: true)]	BOOL4	N/A	%s	0	1	
nod.archive_info	current_bytes_archived	Total bytes written to the current ark file	FLOAT8	bytes	%.0lf	0		
nod.archive_info	total_bytes_archived	Total bytes written since this data source starts	FLOAT8	bytes	%.0lf	0		
nod.archive_nsamples	current_bytes_archived_nsample		INT4	N/A	%d	0		event
nod.archive_nsamples	total_bytes_archived_nsample		INT4	N/A	%d	0		event

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
nod.beams	all_names	ASCII list of the nod names	STRING	N/A	%s			
nod.beams	last_changed	Last beam that was added or deleted	STRING	N/A	%s			
nod.beams	last_distributed	Last beam that was distributed	STRING	N/A	%s			
nod.beams	number	Number of beam positions currently defined	INT4	N/A	%d	0		
nod.beams.ANY	delta_dec	Delta declination for the plus beam	FLOAT8	degrees	%lf			
nod.beams.ANY	delta_el	Change in fine-drive elevation for the plus beam	FLOAT8	degrees	%lf			
nod.beams.ANY	delta_ra	Delta Right Ascension for the plus beam	FLOAT8	sidereal hours	%lf			
nod.beams.ANY	delta_xel	Change in fine-drive cross-elevation for the plus beam	FLOAT8	degrees	%lf			
nod.beams.ANY	delta_xffi	Change in FFI X-pixels for the plus beam	FLOAT8	pixels	%lf			
nod.beams.ANY	delta_xfpi	Change in FPI X-pixels for the plus beam	FLOAT8	pixels	%lf			
nod.beams.ANY	delta_xsi	Change in SI x-pixels for the plus beam	FLOAT8	pixels	%lf			
nod.beams.ANY	delta_xwfi	Change in WFI X-pixels for the plus beam	FLOAT8	pixels	%lf			
nod.beams.ANY	delta_yffi	Change in FFI Y-pixels for the plus beam	FLOAT8	pixels	%lf			
nod.beams.ANY	delta_yfpi	Change in FPI Y-pixels for the plus beam	FLOAT8	pixels	%lf			
nod.beams.ANY	delta_ysi	Change in SI y-pixels for the plus beam	FLOAT8	pixels	%lf			
nod.beams.ANY	delta_ywfi	Change in WFI Y-pixels for the plus beam	FLOAT8	pixels	%lf			
nod.define	amp2	Size of the secondary throw (between X and B)	FLOAT8	arcseconds	%lf	0		
nod.define	amplitude	Size of the primary throw (between X and A)	FLOAT8	arcseconds	%lf	0		
nod.define	coord_sys	Coordinate system for pos_angle	STRING	N/A	%s			
nod.define	pos_angle	Angle of the nod axis	FLOAT8	degrees	%lf			
nod.define	profile	Number of nod beams	INT4	N/A	%i	2	3	
nod.define	track_pos_a	Name of position to use for tracking when at beam A	STRING	N/A	%s			
nod.define	track_pos_b	Name of position to use for tracking when at beam B	STRING	N/A	%s			
nod.define	track_pos_x	Name of position to use for tracking when at beam X	STRING	N/A	%s			
nod.nod_alarm	nod_alarm_error	Serious problems requiring user confirmation	STRING	N/A	%s			event
nod.nod_alarm	nod_alarm_fatal	Fatal problems requiring user confirmation	STRING	N/A	%s			event
nod.nod_alarm	nod_alarm_info	Normal events requiring user confirmation	STRING	N/A	%s			event
nod.nod_alarm	nod_alarm_warning	Abnormal events requiring user confirmation	STRING	N/A	%s			event
nod.nod_alert	nod_alert_error	Non-recoverable problems which are not fatal	STRING	N/A	%s			event
nod.nod_alert	nod_alert_fatal	Fatal problems	STRING	N/A	%s			event
nod.nod_alert	nod_alert_info	Informational messages (events)	STRING	N/A	%s			event
nod.nod_alert	nod_alert_warning	Events requiring user action for recovery	STRING	N/A	%s			event
nod.responses	correct	Last SCL response sent back to the issuer of the correct command	STRING	N/A	%s			event
nod.responses	define	Last SCL response sent back to the issuer of the define command	STRING	N/A	%s			event
nod.responses	goto	Last SCL response sent back to the issuer of the goto command	STRING	N/A	%s			event
nod.responses	next	Last SCL response sent back to the issuer of the next command	STRING	N/A	%s			event
nod.responses	quiet	Last SCL response sent back to the issuer of the quiet command	STRING	N/A	%s			event
nod.responses	resume	Last SCL response sent back to the issuer of the resume command	STRING	N/A	%s			event
nod.responses	set	Last SCL response sent back to the issuer of the set command	STRING	N/A	%s			event
nod.responses	suspend	Last SCL response sent back to the issuer of the suspend command	STRING	N/A	%s			event
nod.responses	verbose	Last SCL response sent back to the issuer of the verbose command	STRING	N/A	%s			event
ntp								
ntp	rcs_id	Version information for the XML file that defines this data source	STRING	N/A	%s			event
ntp.archive_info	archive_file_name	Archive file name	STRING	N/A	%s			event
ntp.archive_info	archive_status	[ENUM: (0: false) (1: true)]	BOOL4	N/A	%s	0	1	
ntp.archive_info	current_bytes_archived	Total bytes written to the current ark file	FLOAT8	bytes	%.0lf	0		
ntp.archive_info	total_bytes_archived	Total bytes written since this data source starts	FLOAT8	bytes	%.0lf	0		
ntp.ntp	gps_antenna_problem	GPS Antenna Problem true, GPS Antenna OK false [NUMBER: (0: false) (1: true)]	BOOL4	N/A	%s			
ntp.ntp	ntp_in_holdover_mode	In Holdover true, No longer in Holdover false [NUMBER: (0: false) (1: true)]	BOOL4	N/A	%s			
ntp.ntp	ntp_in_sync	In Sync true, Not In Sync false [NUMBER: (0: false) (1: true)]	BOOL4	N/A	%s			
ntp.ntp_alarm	ntp_alarm_error	Error alert	STRING	N/A	%s			event

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
ntp.ntp_alarm	ntp_alarm_fatal	Fatal alert	STRING	N/A	%s			event
ntp.ntp_alarm	ntp_alarm_info	Information alarm	STRING	N/A	%s			event
ntp.ntp_alarm	ntp_alarm_warning	Warning alarm	STRING	N/A	%s			event
ntp.ntp_alert	ntp_alert_error	Error alert	STRING	N/A	%s			event
ntp.ntp_alert	ntp_alert_fatal	Fatal alert	STRING	N/A	%s			event
ntp.ntp_alert	ntp_alert_info	Information alert	STRING	N/A	%s			event
ntp.ntp_alert	ntp_alert_warning	Warning alert	STRING	N/A	%s			event
rien								
rien	commands	Contains the last SCL command sent to rien	STRING	N/A	%s			event
rien	oper_state	Operating state	INT4	N/A	0x%2.2X			
rien	rcs_id	Version information for the XML file that defines this data source	STRING	N/A	%s			event
rien.archive_info	archive_file_name	Archive file name	STRING	N/A	%s			event
rien.archive_info	archive_status	[ENUM: (0: false) (1: true)]	BOOL4	N/A	%s	0	1	
rien.archive_info	current_bytes_archived	Total bytes written to the current ark file	FLOAT8	bytes	%.0lf	0		
rien.archive_info	total_bytes_archived	Total bytes written since this data source starts	FLOAT8	bytes	%.0lf	0		
rien.archive_nsamples	current_bytes_archived_nsample		INT4	N/A	%d	0		event
rien.archive_nsamples	total_bytes_archived_nsample		INT4	N/A	%d	0		event
rien.responses	mode	Last SCL response sent back to the issuer of the mode command	STRING	N/A	%s			event
rien.responses	quiet	Last SCL response sent back to the issuer of the quiet command	STRING	N/A	%s			event
rien.responses	restore	Last SCL response sent back to the issuer of the restore command	STRING	N/A	%s			event
rien.responses	resume	Last SCL response sent back to the issuer of the resume command	STRING	N/A	%s			event
rien.responses	save	Last SCL response sent back to the issuer of the save command	STRING	N/A	%s			event
rien.responses	set	Last SCL response sent back to the issuer of the set command	STRING	N/A	%s			event
rien.responses	si_align	Last SCL response sent back to the issuer of the si_align command	STRING	N/A	%s			event
rien.responses	si_boresight	Last SCL response sent back to the issuer of the si_boresight command	STRING	N/A	%s			event
rien.responses	sibs_change	Last SCL response sent back to the issuer of the sibs_change command	STRING	N/A	%s			event
rien.responses	suspend	Last SCL response sent back to the issuer of the suspend command	STRING	N/A	%s			event
rien.responses	verbose	Last SCL response sent back to the issuer of the verbose command	STRING	N/A	%s			event
rien.rien_alarm	rien_alarm_error	Serious problem requiring user confirmation	STRING	N/A	%s			event
rien.rien_alarm	rien_alarm_fatal	Fatal problem requiring user confirmation	STRING	N/A	%s			event
rien.rien_alarm	rien_alarm_info	Normal event requiring user confirmation	STRING	N/A	%s			event
rien.rien_alarm	rien_alarm_warning	Abnormal event requiring user confirmation	STRING	N/A	%s			event
rien.rien_alert	rien_alert_error	Non-recoverable problem which is not fatal	STRING	N/A	%s			event
rien.rien_alert	rien_alert_fatal	Fatal problem	STRING	N/A	%s			event
rien.rien_alert	rien_alert_info	Informational message (event)	STRING	N/A	%s			event
rien.rien_alert	rien_alert_warning	Warning message (may require user action for recovery)	STRING	N/A	%s			event
rien.rien_mode_1	fpi	Focal Plane Imager present in light path, [ENUM: (0: false) (1: true)]	BOOL4	N/A	%s			
rien.rien_mode_1	rotation_si	SI rotation relative to the mounting flange	FLOAT8	degrees	%lf			
rien.rien_mode_1	ta_focus_position_si	TA Focus position on the z axis of SIMFRF	FLOAT8	millimeters	%lf	-300	900	
rien.rien_mode_1	x_pixel_boresight_si	Column of array in which lies the SI Boresight	FLOAT8	pixels	%lf	-1023	1024	
rien.rien_mode_1	x_pixel_max_si	SI X-axis maximum pixel number	INT4	pixels	%d			
rien.rien_mode_1	x_pixel_min_si	SI X-axis minimum pixel number	INT4	pixels	%d			
rien.rien_mode_1	x_pixel_origin_ta	Column of array in which lies the TA optical axis	FLOAT8	pixels	%lf			
rien.rien_mode_1	x_scale_si	SI x-axis pixel scale factor	FLOAT8	arcseconds per millimeter	%lf			
rien.rien_mode_1	x_sep_si	SI x-axis pixel separation (center to center)	FLOAT8	micrometers	%lf			
rien.rien_mode_1	y_pixel_boresight_si	Row of array in which lies the SI Boresight	FLOAT8	pixels	%lf	-1023	1024	
rien.rien_mode_1	y_pixel_max_si	SI Y-axis maximum pixel number	INT4	pixels	%d			

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
rien.rien_mode_1	y_pixel_min_si	SI Y-axis minimum pixel number	INT4	pixels	%d			
rien.rien_mode_1	y_pixel_origin_ta	Row of array in which lies the TA optical axis	FLOAT8	pixels	%lf			
rien.rien_mode_1	y_scale_si	SI y-axis pixel scale factor	FLOAT8	arcseconds per millimeter	%lf			
rien.rien_mode_1	y_sep_si	SI y-axis pixel separation (center to center)	FLOAT8	micrometers	%lf			
rien.rien_mode_2	fpi	Focal Plane Imager present in light path, [ENUM: (0: false) (1: true)]	BOOL4	N/A	%s			
rien.rien_mode_2	rotation_si	SI rotation relative to the mounting flange	FLOAT8	degrees	%lf			
rien.rien_mode_2	ta_focus_position_si	TA Focus position on the z axis of SIMFRF	FLOAT8	millimeters	%lf	-300	900	
rien.rien_mode_2	x_pixel_boresight_si	Column of array in which lies the SI Boresight	FLOAT8	pixels	%lf	-1023	1024	
rien.rien_mode_2	x_pixel_max_si	SI X-axis maximum pixel number	INT4	pixels	%d			
rien.rien_mode_2	x_pixel_min_si	SI X-axis minimum pixel number	INT4	pixels	%d			
rien.rien_mode_2	x_pixel_origin_ta	Column of array in which lies the TA optical axis	FLOAT8	pixels	%lf			
rien.rien_mode_2	x_scale_si	SI x-axis pixel scale factor	FLOAT8	arcseconds per millimeter	%lf			
rien.rien_mode_2	x_sep_si	SI x-axis pixel separation	FLOAT8	micrometers	%lf			
rien.rien_mode_2	y_pixel_boresight_si	Row of array in which lies the SI Boresight	FLOAT8	pixels	%lf	-1023	1024	
rien.rien_mode_2	y_pixel_max_si	SI Y-axis maximum pixel number	INT4	pixels	%d			
rien.rien_mode_2	y_pixel_min_si	SI X-axis minimum pixel number	INT4	pixels	%d			
rien.rien_mode_2	y_pixel_origin_ta	Row of array in which lies the TA optical axis	FLOAT8	pixels	%lf			
rien.rien_mode_2	y_scale_si	SI y-axis pixel scale factor	FLOAT8	arcseconds per millimeter	%lf			
rien.rien_mode_2	y_sep_si	SI y-axis pixel separation	FLOAT8	micrometers	%lf			
rien.save	file	Last file written by the save command	STRING	N/A	%s			
rien.si_align	images	Number of images to use in the alignment process if position is in AOI	UINT4	N/A	%d			
rien.si_boresight	images	Number of images to use in the boresighting process if position is in AOI	UINT4	N/A	%d			
rien.si_config	analog_chops	Additional chop positions for the RIEN instrument	STRING	N/A	%s			
rien.si_config	current_mode	Current configuration mode for the SI, [ENUM: (1: rien_mode_1) (2: rien_mode_2) (3: rien_mode_3) (4: rien_mode_4)]	STRING	N/A	%s			
<session>								
<session>	commands	Contains the last SCL command sent to the session	STRING	N/A	%s			event
<session>	comment	User comment	STRING	N/A	%s			
<session>	dropped_responses	The fraction of intermediate responses that are dropped	FLOAT8	percent	%lf			
<session>	obj_limit	The limit of object handles in a session node; above this value, login and subscribes are rejected	INT4	percent	%d			
<session>	rcs_id	Version information for the XML file that defines this data source	STRING	N/A	%s			event
<session>.archive_info	archive_file_name	Archive file name	STRING	N/A	%s			event
<session>.archive_info	archive_status	[ENUM: (0: false) (1: true)]	BOOL4	N/A	%s	0	1	
<session>.archive_info	current_bytes_archived	Total bytes written to the current ark file	FLOAT8	bytes	%.0lf	0		
<session>.archive_info	total_bytes_archived	Total bytes written since this data source starts	FLOAT8	bytes	%.0lf	0		
<session>.archive_nsamples	current_bytes_archived_nsample		INT4	N/A	%d	0		event
<session>.archive_nsamples	total_bytes_archived_nsample		INT4	N/A	%d	0		event
<session>.user_alarms	user_alarm_error	Serious problem requiring user confirmation	STRING	N/A	%s			event
<session>.user_alarms	user_alarm_fatal	Fatal problem requiring user confirmation	STRING	N/A	%s			event
<session>.user_alarms	user_alarm_info	Normal event requiring user confirmation	STRING	N/A	%s			event
<session>.user_alarms	user_alarm_warning	Abnormal event requiring user confirmation	STRING	N/A	%s			event

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
<session>.user_alerts	user_alert_error	Non-recoverable problem which is not fatal	STRING	N/A	%s			event
<session>.user_alerts	user_alert_fatal	Fatal problem	STRING	N/A	%s			event
<session>.user_alerts	user_alert_info	Informational message (event)	STRING	N/A	%s			event
<session>.user_alerts	user_alert_warning	Warning message (may require user action for recovery)	STRING	N/A	%s			event
<session>.user_environment	attr	Global attributes	STRING	N/A	%s			
<session>.user_environment	host	Hostname for this session	STRING	N/A	%s			
<session>.user_environment	logged_in	Whether or not this user is logged in., [NUMBER: (0: no) (1: yes)]	BOOL4	N/A	%s	0	1	
<session>.user_environment	logpath	Log path	STRING	N/A	%s			
<session>.user_environment	mission_id	Mission ID of FLT	STRING	N/A	%s			
<session>.user_environment	new_data_source	Last data source this session received notification for	STRING	N/A	%s			
<session>.user_environment	pid	Process ID for this session	INT4	N/A	%d			
<session>.user_environment	resp_format	Specifies output format for SI04 responses (valid values are: LEGACY, BINARY, FORMATTED), [ENUM: (0: legacy) (1: binary) (2: formatted)]	STRING	N/A	%s			
<session>.user_environment	role	Role of this user	STRING	N/A	%s			
<session>.user_environment	session_name	Assigned name for this session	STRING	N/A	%s			
<session>.user_environment	showlabels	Whether or not labels are displayed in responses (global)., [NUMBER: (0: no) (1: yes)]	BOOL4	N/A	%s	0	1	
<session>.user_environment	showunits	Whether or not units are displayed in responses (global)., [NUMBER: (0: no) (1: yes)]	BOOL4	N/A	%s	0	1	
<session>.user_environment	username	User who owns this session	STRING	N/A	%s			
<session>.user_environment	verbose	Verbosity level, [NUMBER: (0: none) (1: normal) (2: all)]	STRING	N/A	%s			
sma								
sma	chop_symmetry	An indicator of chopper symmetry [ENUM: (0: no_chop) (2: asymmetric) (4: non_zero_tilt) (5: 2_point_symmetric) (9: 3_point_symmetric) (6: asymmetric_plus) (10: asymmetric_minus) (7: extreme_asymmetric_plus) (11: extreme_asymmetric_minus) (15: contrived)]	STRING	N/A	%s			
sma	chop_tcm_decenter_weighting	Controls the amount that the TCM participates in tip and tilt (1.0=maximum extent possible)	FLOAT8	N/A	%lf	0.0	1.0	
sma	collimation_r	The FCM collimation setting for R	FLOAT8	arcseconds	%lf	-1800	1800	
sma	collimation_s	The FCM collimation setting for S	FLOAT8	arcseconds	%lf	-1800	1800	
sma	collimation_tilt	The FCM collimation setting for tilt	FLOAT8	arcseconds	%lf	-1800	1800	
sma	collimation_tip	The FCM collimation setting for tip	FLOAT8	arcseconds	%lf	-1800	1800	
sma	commands	Contains the last SCL command sent to sma	STRING	N/A	%s			event
sma	focus_coef_a	Influence coefficient for T1 temperature used in determination of the optimum SMA focus for a given temperature	FLOAT8	N/A	%lf			
sma	focus_coef_b	Influence coefficient for T2 temperature used in determination of the optimum SMA focus for a given temperature	FLOAT8	N/A	%lf			
sma	focus_coef_c	Influence coefficient for T3 temperature used in determination of the optimum SMA focus for a given temperature	FLOAT8	N/A	%lf			
sma	focus_coef_k	Influence coefficient for arbitrary HK parameter X used in determination of the optimum SMA focus for a given temperature	FLOAT8	N/A	%lf			

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
sma	focus_coef_q	Influence coefficient for arbitrary HK parameter X used in determination of the optimum SMA focus for a given temperature	FLOAT8	N/A	%lf			
sma	focus_delta	Delta applied to calculated focus change due to temperature . This value is computed from the coefficients and realtime values following the form: delta = a*T1 + b*T2 + c*T3 + q*X and is used in computation of a calculated FCM T axis via: FCM_tcalc = offset + Z0 + K*Zsi + delta	FLOAT8	micrometers	%lf	-5000	5000	
sma	focus_fcm_t_calc	Calculated SMA focus for a given temperature based on an equation FCM_tcalc = Z0 + K*Zsi + a*T1 + b*T2 + c*T3 + q*X.	FLOAT8	micrometers	%lf			
sma	focus_param_t1	Name of housekeeping parameter which will provide T1 temperature value used in determination of the optimum SMA focus for a given temperature	STRING	N/A	%s			
sma	focus_param_t2	Name of housekeeping parameter which will provide T2 temperature value used in determination of the optimum SMA focus for a given temperature	STRING	N/A	%s			
sma	focus_param_t3	Name of housekeeping parameter which will provide T3 temperature value used in determination of the optimum SMA focus for a given temperature	STRING	N/A	%s			
sma	focus_param_x	Name of housekeeping parameter which will provide a value used in determination of the optimum SMA focus for a given temperature	STRING	N/A	%s			
sma	focus_param_value_t1	Actual values of parameters defined by corresponding sma.focus_param_t1	FLOAT8	N/A	%lf			
sma	focus_param_value_t2	Actual values of parameters defined by corresponding sma.focus_param_t2	FLOAT8	N/A	%lf			
sma	focus_param_value_t3	Actual values of parameters defined by corresponding sma.focus_param_t3	FLOAT8	N/A	%lf			
sma	focus_param_value_x	Actual values of parameters defined by corresponding sma.focus_param_x	FLOAT8	N/A	%lf			
sma	focus_total_offset	Total offset applied to calculated focus change due to temperature. This value is computed via use of the sma.focus temp_focus=here argument or, alternatively, is an accumulation of the offsets provided by the operator via the sma.focus offset=XX command.	FLOAT8	N/A	%lf	-5000	5000	
sma	oper_state	Operating state	INT4	N/A	0x%2.2X			
sma	rcs_id	Version information for the XML file that defines this data source	STRING	N/A	%s			event
sma	sky_amp2	Calculated second amplitude on the sky	FLOAT8	arcseconds	%lf	-1125	1125	
sma	sky_amplitude	Calculated amplitude on the sky	FLOAT8	arcseconds	%lf	-1125	1125	
sma	sky_angle	Calculated angle in the sky_coord_sys reference frame	FLOAT8	degrees	%lf	0	360	
sma	sky_coord_sys	Reference frame for which MCCS computes SMA parameters	STRING	N/A	%s			
sma	sky_tilt	Calculated tilt in the sky_coord_sys reference frame	FLOAT8	arcseconds	%lf	-1125	1125	
sma	sky_tip	Calculated tip in the sky_coord_sys reference frame	FLOAT8	arcseconds	%lf	-1125	1125	
sma.archive_info	archive_file_name	Archive file name	STRING	N/A	%s			event
sma.archive_info	archive_status	[ENUM: (0: false) (1: true)]	BOOL4	N/A	%s	0	1	
sma.archive_info	current_bytes_archived	Total bytes written to the current ark file	FLOAT8	bytes	%.0lf	0		
sma.archive_info	total_bytes_archived	Total bytes written since this data source starts	FLOAT8	bytes	%.0lf	0		
sma.archive_nsamples	current_bytes_archived_nsample		INT4	N/A	%d	0		event
sma.archive_nsamples	total_bytes_archived_nsample		INT4	N/A	%d	0		event
sma.chop	coord_sys	User-supplied coordinate system for the sma.chop command, [ENUM: (10: erf) (11: sirf) (12: wfirf) (13: ffirf) (14: fpirf)]	STRING	N/A	%s			
sma.chop	frequency	User-supplied frequency for the sma.chop command	FLOAT8	hertz	%lf	0	20	
sma.chop	phase	User-supplied phase for the sma.chop command	FLOAT8	milliseconds	%lf	0	1000	
sma.chop	pos_ang	User-supplied angle for the sma.chop command	FLOAT8	degrees	%lf			
sma.chop	profile	User-supplied profile for the sma.chop command	UINT4	N/A	%d	0		
sma.chop	settling_time	User-supplied settling time for the sma.chop command	FLOAT8	milliseconds	%lf	5	100	
sma.chop	sync_src	User-supplied synchronization source for the sma.chop command, [ENUM: (0: internal) (1: external) (3: analog)]	STRING	N/A	%s			
sma.chop	user_amp2	User-supplied second amplitude for the sma.chop command	FLOAT8	arcseconds	%lf			
sma.chop	user_amplitude	User-supplied amplitude for the sma.chop command	FLOAT8	arcseconds	%lf			
sma.chop	user_angle	User-supplied angle for the sma.chop command	FLOAT8	degrees	%lf			

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
sma.chop	user_tilt	User-supplied tilt center for the sma.chop command	FLOAT8	arcseconds	%lf			
sma.chop	user_tip	User-supplied tip center for the sma.chop command	FLOAT8	arcseconds	%lf			
sma.focus	mode	Instrument mode assumed by focus	STRING	N/A	%s			
sma.focus	nominal_fcm_t	FCM T axis value to put TA focal-plane at the SI flange when all temperatures are zero.	FLOAT8	micrometers	%lf			
sma.focus	offset	Offset applied to calculated focus change due to temperature.	FLOAT8	micrometers	%lf	-5000	5000	
sma.focus	position	Last (and default) T position for the sma.focus command	FLOAT8	micrometers	%lf	-5000	5000	
sma.points.minus	delta_dec	Delta declination for the minus beam	FLOAT8	degrees	%lf			
sma.points.minus	delta_el	Change in fine-drive elevation for the minus beam	FLOAT8	degrees	%lf			
sma.points.minus	delta_ra	Delta Right Ascension for the minus beam	FLOAT8	sidereal hours	%lf			
sma.points.minus	delta_xel	Change in fine-drive cross-elevation for the minus beam	FLOAT8	degrees	%lf			
sma.points.minus	delta_xffi	Change in FFI X-pixels for the minus beam	FLOAT8	pixels	%lf			
sma.points.minus	delta_xfpi	Change in FPI X-pixels for the minus beam	FLOAT8	pixels	%lf			
sma.points.minus	delta_xsi	Change in SI x-pixels for the minus beam	FLOAT8	pixels	%lf			
sma.points.minus	delta_xwfi	Change in WFI X-pixels for the minus beam	FLOAT8	pixels	%lf			
sma.points.minus	delta_yffi	Change in FFI Y-pixels for the minus beam	FLOAT8	pixels	%lf			
sma.points.minus	delta_yfpi	Change in FPI Y-pixels for the minus beam	FLOAT8	pixels	%lf			
sma.points.minus	delta_ysi	Change in SI y-pixels for the minus beam	FLOAT8	pixels	%lf			
sma.points.minus	delta_ywfi	Change in WFI Y-pixels for the minus beam	FLOAT8	pixels	%lf			
sma.points.plus	delta_dec	Delta declination for the plus beam	FLOAT8	degrees	%lf			
sma.points.plus	delta_el	Change in fine-drive elevation for the plus beam	FLOAT8	degrees	%lf			
sma.points.plus	delta_ra	Delta Right Ascension for the plus beam	FLOAT8	sidereal hours	%lf			
sma.points.plus	delta_xel	Change in fine-drive cross-elevation for the plus beam	FLOAT8	degrees	%lf			
sma.points.plus	delta_xffi	Change in FFI X-pixels for the plus beam	FLOAT8	pixels	%lf			
sma.points.plus	delta_xfpi	Change in FPI X-pixels for the plus beam	FLOAT8	pixels	%lf			
sma.points.plus	delta_xsi	Change in SI x-pixels for the plus beam	FLOAT8	pixels	%lf			
sma.points.plus	delta_xwfi	Change in WFI X-pixels for the plus beam	FLOAT8	pixels	%lf			
sma.points.plus	delta_yffi	Change in FFI Y-pixels for the plus beam	FLOAT8	pixels	%lf			
sma.points.plus	delta_yfpi	Change in FPI Y-pixels for the plus beam	FLOAT8	pixels	%lf			
sma.points.plus	delta_ysi	Change in SI y-pixels for the plus beam	FLOAT8	pixels	%lf			
sma.points.plus	delta_ywfi	Change in WFI Y-pixels for the plus beam	FLOAT8	pixels	%lf			
sma.position	r	Last (and default) R offset for the sma.position command	FLOAT8	micrometers	%lf	-5000	5000	
sma.position	s	Last (and default) S offset for the sma.position command	FLOAT8	micrometers	%lf	-5000	5000	
sma.position	tilt	Last (and default) tilt for the sma.position command	FLOAT8	arcseconds	%lf	-1800	1800	
sma.position	tip	Last (and default) tip for the sma.position command	FLOAT8	arcseconds	%lf	-1800	1800	
sma.responses	align_fcm	Last SCL response sent back to the issuer of the align_fcm command	STRING	N/A	%s			event
sma.responses	align_tcm	Last SCL response sent back to the issuer of the align_tcm command	STRING	N/A	%s			event
sma.responses	chop	Last SCL response sent back to the issuer of the chop command	STRING	N/A	%s			event
sma.responses	focus	Last SCL response sent back to the issuer of the focus command	STRING	N/A	%s			event
sma.responses	init	Last SCL response sent back to the issuer of the init command	STRING	N/A	%s			event
sma.responses	park	Last SCL response sent back to the issuer of the park command	STRING	N/A	%s			event
sma.responses	position	Last SCL response sent back to the issuer of the position command	STRING	N/A	%s			event
sma.responses	quiet	Last SCL response sent back to the issuer of the quiet command	STRING	N/A	%s			event
sma.responses	reset	Last SCL response sent back to the issuer of the reset command	STRING	N/A	%s			event
sma.responses	resume	Last SCL response sent back to the issuer of the resume command	STRING	N/A	%s			event
sma.responses	set	Last SCL response sent back to the issuer of the set command	STRING	N/A	%s			event
sma.responses	suspend	Last SCL response sent back to the issuer of the suspend command	STRING	N/A	%s			event

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
sma.responses	verbose	Last SCL response sent back to the issuer of the verbose command	STRING	N/A	%s			event
sma.sma_alarm	sma_alarm_error	Serious problems requiring user confirmation	STRING	N/A	%s			event
sma.sma_alarm	sma_alarm_fatal	Fatal problems requiring user confirmation	STRING	N/A	%s			event
sma.sma_alarm	sma_alarm_info	Normal events requiring user confirmation	STRING	N/A	%s			event
sma.sma_alarm	sma_alarm_warning	Abnormal events requiring user confirmation	STRING	N/A	%s			event
sma.sma_alert	sma_alert_error	Non-recoverable problems which are not fatal	STRING	N/A	%s			event
sma.sma_alert	sma_alert_fatal	Fatal problems	STRING	N/A	%s			event
sma.sma_alert	sma_alert_info	Informational messages (events)	STRING	N/A	%s			event
sma.sma_alert	sma_alert_warning	Events requiring user action for recovery	STRING	N/A	%s			event
sma.temp_focus	auto_enable	Governs automatic commanding of the SMA to the computed optimum focus based on a given temperature	BOOL4	N/A	%s	0	1	
sma.temp_focus	auto_max_change	Maximum change in focus allowed in order to command the SMA to the computed optimum focus based on a given temperature	FLOAT8	micrometers	%lf	1	5000	
sma.temp_focus	auto_min_change	Minimum change in focus required in order to command the SMA to the computed optimum focus based on a given temperature	FLOAT8	micrometers	%lf	1	5000	
sma.temp_focus	t_process_msg	Periodic status responses from the automatic temperature based focus	STRING	N/A	%s			event
ta								
ta	commands	Contains the last SCL command sent to the ta commandable	STRING	N/A	%s			event
ta	rcs_id	Version information for the XML file that defines this data source	STRING	N/A	%s			event
ta.archive_info	archive_file_name	Archive file name	STRING	N/A	%s			event
ta.archive_info	archive_status	[ENUM: (0: false) (1: true)]	BOOL4	N/A	%s	0	1	
ta.archive_info	current_bytes_archived	Total bytes written to the current ark file	FLOAT8	bytes	%.0lf	0		
ta.archive_info	total_bytes_archived	Total bytes written since this data source starts	FLOAT8	bytes	%.0lf	0		
ta.archive_nsamples	current_bytes_archived_nsample		INT4	N/A	%d	0		event
ta.archive_nsamples	total_bytes_archived_nsample		INT4	N/A	%d	0		event
ta.responses	cmd_timeout	Last SCL response sent back to the issuer of the cmd_timeout command	STRING	N/A	%s			event
ta.responses	poll	Last SCL response sent back to the issuer of the poll command	STRING	N/A	%s			event
ta.ta_alarm	ta_alarm_error	Serious problem requiring user confirmation	STRING	N/A	%s			event
ta.ta_alarm	ta_alarm_fatal	Fatal problem requiring user confirmation	STRING	N/A	%s			event
ta.ta_alarm	ta_alarm_info	Normal event requiring user confirmation	STRING	N/A	%s			event
ta.ta_alarm	ta_alarm_warning	Abnormal event requiring user confirmation	STRING	N/A	%s			event
ta.ta_alert	ta_alert_error	Non-recoverable problem which is not fatal	STRING	N/A	%s			event
ta.ta_alert	ta_alert_fatal	Fatal problem	STRING	N/A	%s			event
ta.ta_alert	ta_alert_info	Informational message (event)	STRING	N/A	%s			event
ta.ta_alert	ta_alert_warning	Warning message (may require user action for recovery)	STRING	N/A	%s			event
ta_ffi								
ta_ffi	commands	Contains the last SCL command sent to the ta_ffi commandable	STRING	N/A	%s			event
ta_ffi	rcs_id	Version information for the XML file that defines this data source	STRING	N/A	%s			event
ta_ffi.archive_info	archive_file_name	Archive file name	STRING	N/A	%s			event
ta_ffi.archive_info	archive_status	[ENUM: (0: false) (1: true)]	BOOL4	N/A	%s	0	1	
ta_ffi.archive_info	current_bytes_archived	Total bytes written to the current ark file	FLOAT8	bytes	%.0lf	0		
ta_ffi.archive_info	total_bytes_archived	Total bytes written since this data source starts	FLOAT8	bytes	%.0lf	0		
ta_ffi.archive_nsamples	current_bytes_archived_nsample		INT4	N/A	%d	0		event
ta_ffi.archive_nsamples	total_bytes_archived_nsample		INT4	N/A	%d	0		event
ta_ffi.ccd_image_data	exposure_duration	Duration of exposure period	INT4	milliseconds	%d	1	10000	
ta_ffi.ccd_image_data	exposure_start_time	Start time of exposure	TIME8	seconds	%d.%09d	0	2147483647	

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
ta_ffi.ccd_image_data	filter_position	Filter position, [ENUM: (-1: UNDEFINED) (0: HOME) (1: CLOSED) (2: FILTER_DAY) (3: FILTER_ND1) (4: FILTER_ND2) (5: FILTER_ND3) (6: CLEAR) (12: FILTER_DAY_U) (13: FILTER_ND1_U) (14: FILTER_ND2_U) (15: FILTER_ND3_U) (16: FILTER_U) (22: FILTER_DAY_G) (23: FILTER_ND1_G) (24: FILTER_ND2_G) (25: FILTER_ND3_G) (26: FILTER_G) (32: FILTER_DAY_R) (33: FILTER_ND1_R) (34: FILTER_ND2_R) (35: FILTER_ND3_R) (36: FILTER_R) (42: FILTER_DAY_I) (43: FILTER_ND1_I) (44: FILTER_ND2_I) (45: FILTER_ND3_I) (46: FILTER_I) (52: FILTER_DAY_Z) (53: FILTER_ND1_Z) (54: FILTER_ND2_Z) (55: FILTER_ND3_Z) (56: FILTER_Z)]	INT4	N/A	%d	-1	56	
ta_ffi.ccd_image_data	focal_length	Focal length of camera optics	FLOAT4	millimeters	%f	100	100000	
ta_ffi.ccd_image_data	focus_position	Focus position. Only valid for FPI. Valid range is -300.0 to 300.0 (or -1000 if unknown).	FLOAT4	millimeters	%f	-300	300	
ta_ffi.ccd_image_data	image_size_transfer	Image size (in Bytes) of current image	INT4	N/A	%d	0	2147483647	
ta_ffi.ccd_image_data	image_sn	Unique identity of the image, with ranges as follows: WFI: 10000...19999, FFI: 20000...29999, FPI: 30000...39999	INT4	N/A	%d	10000	39999	
ta_ffi.ccd_image_data	image_type	Indicates whether the data block is a CCD image, a darkfield image or a flatfield image, [ENUM: (0: CCD_IMAGE) (1: FLATFIELD) (2: DARKFIELD) (3: TEST)]	INT4	N/A	%d	0	3	
ta_ffi.ccd_image_data	image_use	Reports how the image is used, [ENUM: (0: NONE) (1: MONITOR) (2: NORMAL) (3: DARK_ACQUISITION) (4: TEST)]	INT4	N/A	%d	1	4	
ta_ffi.ccd_image_data	packet	data packet as it was received from the TA	BINARY	N/A	%s			
ta_ffi.ccd_image_data	pixel_format	Pixel format of current image, [ENUM: (0: 8_BIT) (1: 14_BIT) (2: 16_BIT)]	INT4	N/A	%d	0	2	
ta_ffi.ccd_image_data	pixel_grouping	Pixel grouping of the current image, [ENUM: (0: 1x1) (1: 2x2) (2: 4x4)]	INT4	N/A	%d	0	2	
ta_ffi.ccd_image_data	pixel_n	Image pixel data. Each pixel is two bytes in size, so if the pixel_format is defined to be 14 bit, the two high bits (14 and 15) are always 0, whereas if the pixel_format is defined to be 8 bit, the high byte is filled with 0s. Also, for any given data packet, there are n pixels, i.e., the image size (see image_size_transfer) may vary with every packet. If the pixel_format is defined to be 16-bit, both bytes are fully used.	BINARY	N/A	%u			
ta_ffi.ccd_image_data	pkt_timestamp	time data packet was constructed by TA	TIME8	seconds	%d.%09d	0	2147483647	
ta_ffi.ccd_image_data	reticle	Reticle brightness. Only valid for FPI.	FLOAT4	N/A	%f	0	1	
ta_ffi.fits_data	fits	Contains the FITS image data (ASCII header followed by the image data in binary)	BINARY	N/A	%s			
ta_ffi.responses	debug	Last SCL response sent back to the issuer of the debug command	STRING	N/A	%s			event
ta_ffi.responses	image_to_hk	Last SCL response sent back to the issuer of the image_to_hk command	STRING	N/A	%s			event
ta_ffi.responses	log	Last SCL response sent back to the issuer of the log command	STRING	N/A	%s			event
ta_ffi.responses	retransmit_packets	Last SCL response sent back to the issuer of the retransmit_packets command	STRING	N/A	%s			event
ta_ffi.responses	show_test_images	Last SCL response sent back to the issuer of the show_test_images command	STRING	N/A	%s			event
ta_ffi.ta_ffi_alarm	ta_ffi_alarm_error	Serious problem requiring user confirmation	STRING	N/A	%s			event
ta_ffi.ta_ffi_alarm	ta_ffi_alarm_fatal	Fatal problem requiring user confirmation	STRING	N/A	%s			event
ta_ffi.ta_ffi_alarm	ta_ffi_alarm_info	Normal event requiring user confirmation	STRING	N/A	%s			event
ta_ffi.ta_ffi_alarm	ta_ffi_alarm_warning	Abnormal event requiring user confirmation	STRING	N/A	%s			event
ta_ffi.ta_ffi_alert	ta_ffi_alert_error	Non-recoverable problem which is not fatal	STRING	N/A	%s			event
ta_ffi.ta_ffi_alert	ta_ffi_alert_fatal	Fatal problem	STRING	N/A	%s			event
ta_ffi.ta_ffi_alert	ta_ffi_alert_info	Informational message (event)	STRING	N/A	%s			event
ta_ffi.ta_ffi_alert	ta_ffi_alert_warning	Warning message (may require user action for recovery)	STRING	N/A	%s			event
ta_fpi								
ta_fpi	commands	Contains the last SCL command sent to the ta_fpi commandable	STRING	N/A	%s			event
ta_fpi	rcs_id	Version information for the XML file that defines this data source	STRING	N/A	%s			event
ta_fpi.archive_info	archive_file_name	Archive file name	STRING	N/A	%s			event
ta_fpi.archive_info	archive_status	[ENUM: (0: false) (1: true)]	BOOL4	N/A	%s	0	1	
ta_fpi.archive_info	current_bytes_archived	Total bytes written to the current ark file	FLOAT8	bytes	%.0lf	0		

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
ta_fpi.archive_info	total_bytes_archived	Total bytes written since this data source starts	FLOAT8	bytes	%.0lf	0		
ta_fpi.archive_nsamples	current_bytes_archived_nsample		INT4	N/A	%d	0		event
ta_fpi.archive_nsamples	total_bytes_archived_nsample		INT4	N/A	%d	0		event
ta_fpi.ccd_image_data	exposure_duration	Duration of exposure period	INT4	milliseconds	%d	1	10000	
ta_fpi.ccd_image_data	exposure_start_time	Start time of exposure	TIME8	seconds	%d.%09d	0	2147483647	
ta_fpi.ccd_image_data	filter_position	Filter position, [ENUM: (-1: UNDEFINED) (0: HOME) (1: CLOSED) (2: FILTER_DAY) (3: FILTER_ND1) (4: FILTER_ND2) (5: FILTER_ND3) (6: CLEAR) (12: FILTER_DAY_U) (13: FILTER_ND1_U) (14: FILTER_ND2_U) (15: FILTER_ND3_U) (16: FILTER_U) (22: FILTER_DAY_G) (23: FILTER_ND1_G) (24: FILTER_ND2_G) (25: FILTER_ND3_G) (26: FILTER_G) (32: FILTER_DAY_R) (33: FILTER_ND1_R) (34: FILTER_ND2_R) (35: FILTER_ND3_R) (36: FILTER_R) (42: FILTER_DAY_I) (43: FILTER_ND1_I) (44: FILTER_ND2_I) (45: FILTER_ND3_I) (46: FILTER_I) (52: FILTER_DAY_Z) (53: FILTER_ND1_Z) (54: FILTER_ND2_Z) (55: FILTER_ND3_Z) (56: FILTER_Z)]	INT4	N/A	%d	-1	56	
ta_fpi.ccd_image_data	focal_length	Focal length of camera optics	FLOAT4	millimeters	%f	100	100000	
ta_fpi.ccd_image_data	focus_position	Focus position. Only valid for FPI. Valid range is -300.0 to 300.0 (or -1000 if unknown).	FLOAT4	millimeters	%f	-300	300	
ta_fpi.ccd_image_data	image_size_transfer	Image size (in Bytes) of current image	INT4	N/A	%d	0	2147483647	
ta_fpi.ccd_image_data	image_sn	Unique identity of the image, with ranges as follows: WFI: 10000...19999, FFI: 20000...29999, FPI: 30000...39999	INT4	N/A	%d	10000	39999	
ta_fpi.ccd_image_data	image_type	Indicates whether the data block is a CCD image, a darkfield image or a flatfield image, [ENUM: (0: CCD_IMAGE) (1: FLATFIELD) (2: DARKFIELD) (3: TEST)]	INT4	N/A	%d	0	3	
ta_fpi.ccd_image_data	image_use	Reports how the image is used, [ENUM: (0: NONE) (1: MONITOR) (2: NORMAL) (3: DARK_ACQUISITION) (4: TEST)]	INT4	N/A	%d	1	4	
ta_fpi.ccd_image_data	packet	data packet as it was received from the TA	BINARY	N/A	%s			
ta_fpi.ccd_image_data	pixel_format	Pixel format of current image, [ENUM: (0: 8_BIT) (1: 14_BIT) (2: 16_BIT)]	INT4	N/A	%d	0	2	
ta_fpi.ccd_image_data	pixel_grouping	Pixel grouping of the current image, [ENUM: (0: 1x1) (1: 2x2) (2: 4x4)]	INT4	N/A	%d	0	2	
ta_fpi.ccd_image_data	pixel_n	Image pixel data. Each pixel is two bytes in size, so if the pixel_format is defined to be 14 bit, the two high bits (14 and 15) are always 0, whereas if the pixel_format is defined to be 8 bit, the high byte is filled with 0s. Also, for any given data packet, there are n pixels, i.e., the image size (see image_size_transfer) may vary with every packet. If the pixel_format is defined to be 16-bit, both bytes are fully used.	BINARY	N/A	%u			
ta_fpi.ccd_image_data	pkt_timestamp	time data packet was constructed by TA	TIME8	seconds	%d.%09d	0	2147483647	
ta_fpi.ccd_image_data	reticle	Reticle brightness. Only valid for FPI.	FLOAT4	N/A	%f	0	1	
ta_fpi.fits_data	fits	Contains the FITS image data (ASCII header followed by the image data in binary)	BINARY	N/A	%s			
ta_fpi.responses	debug	Last SCL response sent back to the issuer of the debug command	STRING	N/A	%s			event
ta_fpi.responses	image_to_hk	Last SCL response sent back to the issuer of the image_to_hk command	STRING	N/A	%s			event
ta_fpi.responses	log	Last SCL response sent back to the issuer of the log command	STRING	N/A	%s			event
ta_fpi.responses	retransmit_packets	Last SCL response sent back to the issuer of the retransmit_packets command	STRING	N/A	%s			event
ta_fpi.responses	show_test_images	Last SCL response sent back to the issuer of the show_test_images command	STRING	N/A	%s			event
ta_fpi.ta_fpi_alarm	ta_fpi_alarm_error	Serious problem requiring user confirmation	STRING	N/A	%s			event
ta_fpi.ta_fpi_alarm	ta_fpi_alarm_fatal	Fatal problem requiring user confirmation	STRING	N/A	%s			event
ta_fpi.ta_fpi_alarm	ta_fpi_alarm_info	Normal event requiring user confirmation	STRING	N/A	%s			event
ta_fpi.ta_fpi_alarm	ta_fpi_alarm_warning	Abnormal event requiring user confirmation	STRING	N/A	%s			event
ta_fpi.ta_fpi_alert	ta_fpi_alert_error	Non-recoverable problem which is not fatal	STRING	N/A	%s			event
ta_fpi.ta_fpi_alert	ta_fpi_alert_fatal	Fatal problem	STRING	N/A	%s			event
ta_fpi.ta_fpi_alert	ta_fpi_alert_info	Informational message (event)	STRING	N/A	%s			event
ta_fpi.ta_fpi_alert	ta_fpi_alert_warning	Warning message (may require user action for recovery)	STRING	N/A	%s			event

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
ta_mcp								
ta_mcp	commands	Contains the last SCL command sent to the MCP	STRING	N/A	%s			event
ta_mcp	rcs_id	Version information for the XML file that defines this data source	STRING	N/A	%s			event
ta_mcp.archive_info	archive_file_name	Archive file name	STRING	N/A	%s			event
ta_mcp.archive_info	archive_status		BOOL4	N/A	%s	0	1	
ta_mcp.archive_info	current_bytes_archived	Total bytes written to the current ark file	FLOAT8	bytes	%.0lf	0		
ta_mcp.archive_info	total_bytes_archived	Total bytes written since this data source starts	FLOAT8	bytes	%.0lf	0		
ta_mcp.archive_nsamples	current_bytes_archived_nsample		INT4	N/A	%d	0		event
ta_mcp.archive_nsamples	mcp_hk_anu_nsample		INT4	N/A	%d	0		event
ta_mcp.archive_nsamples	mcp_hk_pdu_nsample		INT4	N/A	%d	0		event
ta_mcp.archive_nsamples	mcp_hk_pms_nsample		INT4	N/A	%d	0		event
ta_mcp.archive_nsamples	mcp_hk_pwc_nsample		INT4	N/A	%d	0		event
ta_mcp.archive_nsamples	mcp_hk_states_nsample		INT4	N/A	%d	0		event
ta_mcp.archive_nsamples	total_bytes_archived_nsample		INT4	N/A	%d	0		event
ta_mcp.responses	mcp_annunciator_confirm	Last SCL response sent back to the issuer of the mcp_annunciator_confirm command	STRING	N/A	%s			event
ta_mcp.responses	mcp_annunciator_test	Last SCL response sent back to the issuer of the mcp_annunciator_test command	STRING	N/A	%s			event
ta_mcp.responses	mcp_annunciator_test	Last SCL response sent back to the issuer of the mcp_built_in_test command	STRING	N/A	%s			event
ta_mcp.responses	mcp_config_get	Last SCL response sent back to the issuer of the mcp_config_get command	STRING	N/A	%s			event
ta_mcp.responses	mcp_config_load	Last SCL response sent back to the issuer of the mcp_config_load command	STRING	N/A	%s			event
ta_mcp.responses	mcp_config_save	Last SCL response sent back to the issuer of the mcp_config_save command	STRING	N/A	%s			event
ta_mcp.responses	mcp_config_set	Last SCL response sent back to the issuer of the mcp_config_set command	STRING	N/A	%s			event
ta_mcp.responses	mcp_hk_rate	Last SCL response sent back to the issuer of the mcp_hk_rate command	STRING	N/A	%s			event
ta_mcp.responses	mcp_image_qc_init	Last SCL response sent back to the issuer of the mcp_image_qc_init command	STRING	N/A	%s			event
ta_mcp.responses	mcp_image_quality_compensation	Last SCL response sent back to the issuer of the mcp_image_quality_compensation command	STRING	N/A	%s			event
ta_mcp.responses	mcp_pdu_switch	Last SCL response sent back to the issuer of the mcp_pdu_switch command	STRING	N/A	%s			event
ta_mcp.responses	mcp_pwc_gate_valve_close	Last SCL response sent back to the issuer of the mcp_pwc_gate_valve_close command	STRING	N/A	%s			event
ta_mcp.responses	mcp_pwc_gate_valve_open	Last SCL response sent back to the issuer of the mcp_pwc_gate_valve_open command	STRING	N/A	%s			event
ta_mcp.responses	mcp_set_hk_limits	Last SCL response sent back to the issuer of the mcp_set_hk_limits command	STRING	N/A	%s			event
ta_mcp.responses	mcp_sw_version	Last SCL response sent back to the issuer of the mcp_sw_version command	STRING	N/A	%s			event
ta_mcp.responses	mcp_test_command	Last SCL response sent back to the issuer of the mcp_test_command command	STRING	N/A	%s			event
ta_mcp.responses	mcp_tsc_hk_rate	Last SCL response sent back to the issuer of the mcp_tsc_hk_rate command	STRING	N/A	%s			event
ta_mcp.responses	send_packet	Last SCL response sent back to the issuer of the send_packet command	STRING	N/A	%s			event
ta_pos								
ta_pos	commands	Contains the last SCL command sent to ta_pos	STRING	N/A	%s			event
ta_pos	leg_duration	Total duration of the current flight leg	FLOAT8	seconds	%lf	0	86400	
ta_pos	observation_time	Total duration of the current observation	FLOAT8	seconds	%lf	0	86400	
ta_pos	oper_state	Operating state	INT4	N/A	0x%2.2X			
ta_pos	rcs_id	Version information for the XML file that defines this data source	STRING	N/A	%s			event
ta_pos	rewind_target	The LOS that will be used by the ta_pos.rewind command when no “los” argument is provided. Generally, this LOS is near the soft limit, except when observing slow rotators in which case this LOS will be zero.	FLOAT4	degrees	%f			
ta_pos	rewind_trigger	Point at which a ta_pos.rewind will be required	FLOAT8	degrees	%lf			
ta_pos	rof_wait_iter	Maximum number of steps (within trackers ROF Calculation) to wait after commanding the TA to centroid track. 0 means dont wait for racking algorithm. During each step, the MCCS will wait one exposure time for the imager in question. (largest exposure time used if imagers differ). The wait will exit as soon as the proper state is achieved (ROF_LOOP).	INT4	N/A	%d	0		

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
ta_pos	time_of_last_rewind	The time that the last successful integrated SCL rewind occurred (i.e., ta_pos.rewind).	FLOAT8	seconds	%.3ymdz			
ta_pos	time_until_rewind	Time remaining until a ta_pos.rewind will be required, assuming target is tracked perfectly by the aircraft (i.e., no change in XEL)	FLOAT4	seconds	%.0f	0		
ta_pos	time_until_rewind_ideal	Time remaining if aircraft heading changes will be performed that reduce XEL to an ideal of zero.	~~FLOAT4~~ FLOAT8	seconds	%.0f			
ta_pos	track_wait_iter	Maximum number of steps (within trackers Two Axis Position Calculation) to wait after commanding the TA to centroid track. 0 means dont wait for racking algorithm. During each step, the MCCS will wait one exposure time for the imager in question. The wait will exit as soon as the proper state is achieved (INERT_LOOP or NONEINERT_LOOP).	INT4	N/A	%d	0		
ta_pos.archive_info	archive_file_name	Archive file name	STRING	N/A	%s			event
ta_pos.archive_info	archive_status	[ENUM: (0: false) (1: true)]	BOOL4	N/A	%s	0	1	
ta_pos.archive_info	current_bytes_archived	Total bytes written to the current ark file	FLOAT8	bytes	%.0lf	0		
ta_pos.archive_info	total_bytes_archived	Total bytes written since this data source starts	FLOAT8	bytes	%.0lf	0		
ta_pos.archive_nsamples	current_bytes_archived_nsample		INT4	N/A	%d	0		event
ta_pos.archive_nsamples	time_until_rewind_nsample		INT4	N/A	%d	0		event
ta_pos.archive_nsamples	total_bytes_archived_nsample		INT4	N/A	%d	0		event
ta_pos.autofocus	aoi	Name of the AOI to use for focussing	STRING	N/A	%s			
ta_pos.autofocus	timeout	Default timeout for autofocus	UINT4	seconds	%d	1	90	
ta_pos.fpi_focus	mode	Instrument mode during last fpi_focus	STRING	N/A	%s			
ta_pos.pattern	coord_sys	Default reference frame for pattern movement	STRING	N/A	%s			
ta_pos.pattern	mode	Positioning mode for pattern command, [ENUM: (1: absolute) (2: relative)]	STRING	N/A	%s			
ta_pos.pattern	next_id	Next ID to send with pattern command	INT4	N/A	%d			
ta_pos.pattern	rate	Scan rate for pattern command	FLOAT8	arcseconds per second	%lf	0	3609.63411	
ta_pos.pattern	sample_spacing	Distance between sample points along the curve – set to 0.0 to allow the pattern command to choose a distance equal to 0.5sec * rate.	FLOAT8	arcseconds	%lf	0	1000	
ta_pos.pattern	type	Scan type for pattern command [ENUM: (1: step) (2: smooth) (3: linear)]	STRING	N/A	%s			
ta_pos.responses	autofocus	Last SCL response sent back to the issuer of the autofocus command	STRING	N/A	%s			event
ta_pos.responses	fpi_focus	Last SCL response sent back to the issuer of the fpi_focus command	STRING	N/A	%s			event
ta_pos.responses	goto	Last SCL response sent back to the issuer of the goto command	STRING	N/A	%s			event
ta_pos.responses	imager_align	Last SCL response sent back to the issuer of the imager_align command	STRING	N/A	%s			event
ta_pos.responses	offset	Last SCL response sent back to the issuer of the offset command	STRING	N/A	%s			event
ta_pos.responses	pattern	Last SCL response sent back to the issuer of the pattern command	STRING	N/A	%s			event
ta_pos.responses	quiet	Last SCL response sent back to the issuer of the quiet command	STRING	N/A	%s			event
ta_pos.responses	resume	Last SCL response sent back to the issuer of the resume command	STRING	N/A	%s			event
ta_pos.responses	rewind	Last SCL response sent back to the issuer of the rewind command	STRING	N/A	%s			event
~~ta_pos.responses~~	~~scan~~	~~Last SCL response sent back to the issuer of the scan command~~	~~STRING~~	~~N/A~~	~~%s~~			~~event~~
ta_pos.responses	set	Last SCL response sent back to the issuer of the set command	STRING	N/A	%s			event
ta_pos.responses	stop	Last SCL response sent back to the issuer of the stop command	STRING	N/A	%s			event
~~ta_pos.responses~~	~~stow~~	~~Last SCL response sent back to the issuer of the stow command~~	~~STRING~~	~~N/A~~	~~%s~~			~~event~~
ta_pos.responses	suspend	Last SCL response sent back to the issuer of the suspend command	STRING	N/A	%s			event
ta_pos.responses	track	Last SCL response sent back to the issuer of the track command	STRING	N/A	%s			event
ta_pos.responses	tweak	Last SCL response sent back to the issuer of the tweak command	STRING	N/A	%s			event
ta_pos.responses	tweak_define	Last SCL response sent back to the issuer of the tweak_define command	STRING	N/A	%s	0		event
ta_pos.responses	verbose	Last SCL response sent back to the issuer of the verbose command	STRING	N/A	%s			event
ta_pos.rewind	beam	The chop image of the rewind center [ENUM: (0: OFF) (1: MIDPOINT) (2: PLUS_BEAM) (3: MINUS_BEAM)]	INT4	N/A	%d			
ta_pos.rewind	center	Where to rotate around	STRING	N/A	%s			
ta_pos.rewind	chop_sys	Reference frame in which chopping is constant	STRING	N/A	%s			
ta_pos.rewind	los_margin	Margin between LOS soft and hard limits	FLOAT8	degrees	%lf	0	2.8	
ta_pos.rewind	nod_sys	Reference frame in which nodding is constant	STRING	N/A	%s			

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
ta_pos.rewind	quantum_time	Minimum time between rewinds	FLOAT8	seconds	%lf			
ta_pos.rewind	total_time	Total time for the current observation	FLOAT8	seconds	%lf			
ta_pos.ta_pos_alarm	ta_pos_alarm_error	Serious problems requiring user confirmation	STRING	N/A	%s			event
ta_pos.ta_pos_alarm	ta_pos_alarm_fatal	Fatal problems requiring user confirmation	STRING	N/A	%s			event
ta_pos.ta_pos_alarm	ta_pos_alarm_info	Normal events requiring user confirmation	STRING	N/A	%s			event
ta_pos.ta_pos_alarm	ta_pos_alarm_warning	Abnormal events requiring user confirmation	STRING	N/A	%s			event
ta_pos.ta_pos_alert	ta_pos_alert_error	Non-recoverable problems which are not fatal	STRING	N/A	%s			event
ta_pos.ta_pos_alert	ta_pos_alert_fatal	Fatal problems	STRING	N/A	%s			event
ta_pos.ta_pos_alert	ta_pos_alert_info	Informational messages (events)	STRING	N/A	%s			event
ta_pos.ta_pos_alert	ta_pos_alert_warning	Events requiring user action for recovery	STRING	N/A	%s			event
ta_pos.track	centroid	Object being used for tracking (AOI)	STRING	N/A	%s			
ta_pos.track	limb1	First AOI used for limb tracking	STRING	N/A	%s			
ta_pos.track	limb2	Second AOI used for limb tracking	STRING	N/A	%s			
ta_pos.track	rof1	One of two objects being used for ROF tracking	STRING	N/A	%s			
ta_pos.track	rof2	One of two objects being used for ROF tracking	STRING	N/A	%s			
ta_pos.track	state	Currently commanded track state, [ENUM: (0: off) (0: none)-(3: rof) (5: limb) (41: offset) (43: rof+offset)]	INT4	N/A	0x%X			
ta_pos.tweak	coord_sys	Coordinate system currently used by the tweaker box, [ENUM: (0: irf) (1: taicrf) (2: erf) (3: sirf) (4: fpirf) (5: wfirf) (6: ffirf)]	STRING	N/A	%s			
ta_pos.tweak	mode	Type of movement currently used by the tweaker box, [ENUM: (0: NotSet) (1: scan) (2: offset) (4: step)]	STRING	N/A	%s			
ta_pos.tweak	x_off	Horizontal offset or step from tweaker box button push	FLOAT8	sidereal seconds	%lf	0		
ta_pos.tweak	x_rate	Horizontal scan rate from tweaker box button push	FLOAT8	sidereal seconds per second	%lf	0		
ta_pos.tweak	y_off	Vertical offset or step from tweaker box button push	FLOAT8	arcseconds	%lf	0		
ta_pos.tweak	y_rate	Vertical scan rate from tweaker box button push	FLOAT8	arcseconds per second	%lf	0		
ta_scs								
ta_scs	commands	Contains the last SCL command sent to the SMCU	STRING	N/A	%s			event
ta_scs	rcs_id	Version information for the XML file that defines this data source	STRING	N/A	%s			event
ta_scs.archive_info	archive_file_name	Archive file name	STRING	N/A	%s			event
ta_scs.archive_info	archive_status		BOOL4	N/A	%s	0	1	
ta_scs.archive_info	current_bytes_archived	Total bytes written to the current ark file	FLOAT8	bytes	%.0lf	0		
ta_scs.archive_info	total_bytes_archived	Total bytes written since this data source starts	FLOAT8	bytes	%.0lf	0		
ta_scs.archive_nsamples	current_bytes_archived_nsample		INT4	N/A	%d	0		event
ta_scs.archive_nsamples	total_bytes_archived_nsample		INT4	N/A	%d	0		event
ta_scs.archive_nsamples	tsc_mcs_hk_nsample		INT4	N/A	%d	0		event
ta_scs.responses	scs_align_set	Last SCL response sent back to the issuer of the scs_align_set command	STRING	N/A	%s			event
ta_scs.responses	scs_built_in_test	Last SCL response sent back to the issuer of the scs_built_in_test command	STRING	N/A	%s			event
ta_scs.responses	scs_calibration	Last SCL response sent back to the issuer of the scs_calibration command	STRING	N/A	%s			event
ta_scs.responses	scs_config_get	Last SCL response sent back to the issuer of the scs_config_get command	STRING	N/A	%s			event
ta_scs.responses	scs_config_load	Last SCL response sent back to the issuer of the scs_config_load command	STRING	N/A	%s			event
ta_scs.responses	scs_config_save	Last SCL response sent back to the issuer of the scs_config_save command	STRING	N/A	%s			event
ta_scs.responses	scs_config_set	Last SCL response sent back to the issuer of the scs_config_set command	STRING	N/A	%s			event
ta_scs.responses	scs_fcm_collimation	Last SCL response sent back to the issuer of the scs_fcm_collimation command	STRING	N/A	%s			event
ta_scs.responses	scs_fcm_collimation_sky	Last SCL response sent back to the issuer of the scs_fcm_collimation_sky command	STRING	N/A	%s			event
ta_scs.responses	scs_fcm_focus	Last SCL response sent back to the issuer of the scs_fcm_focus command	STRING	N/A	%s			event
ta_scs.responses	scs_fcm_pos_abs	Last SCL response sent back to the issuer of the scs_fcm_pos_abs command	STRING	N/A	%s			event
ta_scs.responses	scs_fcm_stop	Last SCL response sent back to the issuer of the scs_fcm_stop command	STRING	N/A	%s			event

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
ta_scs.responses	scs_mode_active	Last SCL response sent back to the issuer of the scs_mode_active command	STRING	N/A	%s			event
ta_scs.responses	scs_mode_parked	Last SCL response sent back to the issuer of the scs_mode_parked command	STRING	N/A	%s			event
ta_scs.responses	scs_mode_shutdown	Last SCL response sent back to the issuer of the scs_mode_shutdown command	STRING	N/A	%s			event
ta_scs.responses	scs_mode_standby	Last SCL response sent back to the issuer of the scs_mode_standby command	STRING	N/A	%s			event
ta_scs.responses	scs_status_subscribe	Last SCL response sent back to the issuer of the scs_status_subscribe command	STRING	N/A	%s			event
ta_scs.responses	scs_tcm_analog_in	Last SCL response sent back to the issuer of the scs_tcm_analog_in command	STRING	N/A	%s			event
ta_scs.responses	scs_tcm_chop	Last SCL response sent back to the issuer of the scs_tcm_chop command	STRING	N/A	%s			event
ta_scs.responses	scs_tcm_chop_sky	Last SCL response sent back to the issuer of the scs_tcm_chop_sky command	STRING	N/A	%s			event
ta_scs.responses	scs_tcm_chop_wf_select	Last SCL response sent back to the issuer of the scs_tcm_chop_wf_select command	STRING	N/A	%s			event
ta_scs.responses	scs_tcm_fbc_mode	Last SCL response sent back to the issuer of the scs_tcm_fbc_mode command	STRING	N/A	%s			event
ta_scs.responses	scs_tcm_param_set	Last SCL response sent back to the issuer of the scs_tcm_param_set command	STRING	N/A	%s			event
ta_scs.responses	scs_tcm_pos_abs	Last SCL response sent back to the issuer of the scs_tcm_pos_abs command	STRING	N/A	%s			event
ta_scs.responses	scs_tcm_pos_abs_sky	Last SCL response sent back to the issuer of the scs_tcm_pos_abs_sky command	STRING	N/A	%s			event
ta_scs.responses	scs_tcm_stop	Last SCL response sent back to the issuer of the scs_tcm_stop command	STRING	N/A	%s			event
ta_scs.responses	scs_test_command	Last SCL response sent back to the issuer of the scs_test_command command	STRING	N/A	%s			event
ta_scs.responses	send_packet	Last SCL response sent back to the issuer of the send_packet command	STRING	N/A	%s			event
ta_state								
ta_state	coarse_deg_el	Coarse drive elevation	FLOAT8	degrees	%.8lf	15	70	
ta_state	commands	Contains the last SCL command sent to ta_state	STRING	N/A	%s			event
ta_state	fbc_el_stdev	sigma(EL) for the FBC correction	FLOAT8	N/A	%lf			
ta_state	fbc_xel_stdev	sigma(XEL) for the FBC correction	FLOAT8	N/A	%lf			
ta_state	fd_deg_el	Fine drive elevation	FLOAT8	degrees	%lf	-3	3	
ta_state	fd_deg_los	Fine drive Line of Sight	FLOAT8	degrees	%lf	-3	3	
ta_state	fd_deg_xel	Fine drive cross-elevation	FLOAT8	degrees	%lf	-3	3	
ta_state	fd_lock_time	Time when the fine drive became locked	FLOAT8	seconds	%lf	0		
ta_state	irf_deg_el	TARF-IRF elevation	FLOAT8	degrees	%lf			
ta_state	irf_deg_los	TARF-IRF drive Line of Sight	FLOAT8	degrees	%lf			
ta_state	irf_deg_xel	TARF-IRF drive cross-elevation	FLOAT8	degrees	%lf			
ta_state	oper_state	Operating state	INT4	N/A	0x%2.2X			
ta_state	position	Position the telescope was last pointed to	STRING	N/A	%s			
ta_state	rcs_id	Version information for the XML file that defines this data source	STRING	N/A	%s			event
ta_state	servo_mode	Simplification of ta_tsc.tsc_status	INT4	N/A	%d			
ta_state	ta_deg_el	Total elevation	FLOAT8	degrees	%lf	15	70	
ta_state	ta_deg_los	Total drive Line of Sight	FLOAT8	degrees	%lf	-3	3	
ta_state	ta_deg_xel	Total drive cross-elevation	FLOAT8	degrees	%lf	-3	3	
ta_state	tsc_status	String representation of ta_tsc.tsc_status	STRING	N/A	%s			
ta_state.archive_info	archive_file_name	Archive file name	STRING	N/A	%s			event
ta_state.archive_info	archive_status	[ENUM: (0: false) (1: true)]	BOOL4	N/A	%s	0	1	
ta_state.archive_info	current_bytes_archived	Total bytes written to the current ark file	FLOAT8	bytes	%.0lf	0		
ta_state.archive_info	total_bytes_archived	Total bytes written since this data source starts	FLOAT8	bytes	%.0lf	0		
ta_state.archive_nsamples	coarse_deg_el_nsample		INT4	N/A	%d	0		event
ta_state.archive_nsamples	current_bytes_archived_nsample		INT4	N/A	%d	0		event
ta_state.archive_nsamples	dec_nsample		INT4	N/A	%d	0		event
ta_state.archive_nsamples	fd_deg_el_nsample		INT4	N/A	%d	0		event
ta_state.archive_nsamples	fd_deg_los_nsample		INT4	N/A	%d	0		event
ta_state.archive_nsamples	fd_deg_xel_nsample		INT4	N/A	%d	0		event
ta_state.archive_nsamples	irf_deg_el_nsample		INT4	N/A	%d	0		event
ta_state.archive_nsamples	irf_deg_los_nsample		INT4	N/A	%d	0		event
ta_state.archive_nsamples	irf_deg_xel_nsample		INT4	N/A	%d	0		event
ta_state.archive_nsamples	ra_nsample		INT4	N/A	%d	0		event

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
ta_state.archive_nsamples	rof_nsample		INT4	N/A	%d	0		event
ta_state.archive_nsamples	ta_deg_el_nsample		INT4	N/A	%d	0		event
ta_state.archive_nsamples	ta_deg_los_nsample		INT4	N/A	%d	0		event
ta_state.archive_nsamples	ta_deg_xel_nsample		INT4	N/A	%d	0		event
ta_state.archive_nsamples	total_bytes_archived_nsample		INT4	N/A	%d	0		event
~~ta_state.archive_nsamples~~	~~vpa_nsample~~		~~INT4~~	~~N/A~~	~~%d~~	~~0~~		~~event~~
ta_state.camera.ffi	adcu	Number of bits in the output data, [ENUM: (8: 8) (14: 14) (16:16)]	INT4	N/A	%d	8	16	
ta_state.camera.ffi	binning	Binning factor for camera CCD array, [ENUM: (1: 1x1) (1: 1) (2: 2x2) (2: 2) (4: 4x4) (4: 4)]	INT4	unity	%d	1	4	
ta_state.camera.ffi	integration	Integration time for each camera image	FLOAT8	seconds	%lf	0.01	10	
ta_state.camera.fpi	adcu	Number of bits in the output data, [ENUM: (8: 8) (14: 14) (16:16)]	INT4	N/A	%d	8	16	
ta_state.camera.fpi	binning	Binning factor for camera CCD array, [ENUM: (1: 1x1) (1: 1) (2: 2x2) (2: 2) (4: 4x4) (4: 4)]	INT4	unity	%d	1	4	
ta_state.camera.fpi	integration	Integration time for each camera image	FLOAT8	seconds	%lf	0.01	10	
ta_state.camera.wfi	adcu	Number of bits in the output data, [ENUM: (8: 8) (14: 14) (16:16)]	INT4	N/A	%d	8	16	
ta_state.camera.wfi	binning	Binning factor for camera CCD array, [ENUM: (1: 1x1) (1: 1) (2: 2x2) (2: 2) (4: 4x4) (4: 4)]	INT4	unity	%d	1	4	
ta_state.camera.wfi	integration	Integration time for each camera image	FLOAT8	seconds	%lf	0.01	10	
ta_state.dec	rate	d(dec)/dt for the boresight	FLOAT8	degrees per second	%lf			
ta_state.dec	stdev	sigma(dec) for the boresight	FLOAT8	N/A	%lf			
ta_state.iqc	initialized	Has IQC been initialized?, [ENUM: (0: false) (1: true)]	BOOL4	N/A	%s	0	1	
ta_state.iqc	limit	Margin between IQC set point and triggers	SEXA8	degrees	%sgd	0	55	
ta_state.iqc	monitoring	Is elevation being checked for IQC?, [BIT: (0: user) (1: telescope) (2: sma)]	UINT4	N/A	0x%X	0	7	
ta_state.iqc	set_point	Elevation the last time IQC was used	FLOAT8	degrees	%lf	15	70	
ta_state.iqc	triggered	Is there an outstanding IQC alert?, [ENUM: (0: false) (1: true)]	BOOL4	N/A	%s	0	1	
ta_state.ra	rate	d(RA)/dt for the boresight	FLOAT8	sidereal hours per second	%lf			
ta_state.ra	stdev	sigma(RA) for the boresight	FLOAT8	N/A	%lf			
ta_state.responses	cage	Last SCL response sent back to the issuer of the cage command	STRING	N/A	%s			event
ta_state.responses	camera	Last SCL response sent back to the issuer of the servo command	STRING	N/A	%s			event
ta_state.responses	iqc	Last SCL response sent back to the issuer of the iqc command	STRING	N/A	%s			event
ta_state.responses	quiet	Last SCL response sent back to the issuer of the quiet command	STRING	N/A	%s			event
ta_state.responses	resume	Last SCL response sent back to the issuer of the resume command	STRING	N/A	%s			event
ta_state.responses	servo	Last SCL response sent back to the issuer of the servo command	STRING	N/A	%s			event
ta_state.responses	set	Last SCL response sent back to the issuer of the set command	STRING	N/A	%s			event
ta_state.responses	shutdown	Last SCL response sent back to the issuer of the shutdown command	STRING	N/A	%s			event
ta_state.responses	standby	Last SCL response sent back to the issuer of the standby command	STRING	N/A	%s			event
ta_state.responses	startup	Last SCL response sent back to the issuer of the startup command	STRING	N/A	%s			event
ta_state.responses	suspend	Last SCL response sent back to the issuer of the suspend command	STRING	N/A	%s			event
ta_state.responses	verbose	Last SCL response sent back to the issuer of the verbose command	STRING	N/A	%s			event
ta_state.rof	rate	d(rof)/dt for the boresight	FLOAT8	degrees per second	%lf			
ta_state.rof	stdev	sigma(rof) for the boresight	FLOAT8	N/A	%lf			
ta_state.servo	compensation	Default compensation for ta_state.servo command, [ENUM: (1: FBC_OFF) (2: FBC_QS) (3: FBC_DY) (4: FBC_ON)]	INT4	N/A	%d			
ta_state.ta_state_alarm	ta_state_alarm_error	Serious problems requiring user confirmation	STRING	N/A	%s			event
ta_state.ta_state_alarm	ta_state_alarm_fatal	Fatal problems requiring user confirmation	STRING	N/A	%s			event
ta_state.ta_state_alarm	ta_state_alarm_info	Normal events requiring user confirmation	STRING	N/A	%s			event
ta_state.ta_state_alarm	ta_state_alarm_warning	Abnormal events requiring user confirmation	STRING	N/A	%s			event
ta_state.ta_state_alert	ta_state_alert_error	Non-recoverable problems which are not fatal	STRING	N/A	%s			event
ta_state.ta_state_alert	ta_state_alert_fatal	Fatal problems	STRING	N/A	%s			event

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
ta_state.ta_state_alert	ta_state_alert_info	Informational messages (events)	STRING	N/A	%s			event
ta_state.ta_state_alert	ta_state_alert_warning	Warning message (may require user action for recovery)	STRING	N/A	%s			event
ta_trc								
ta_trc	commands	Contains the last SCL command sent to the Tracker	STRING	N/A	%s			event
ta_trc	rcs_id	Version information for the XML file that defines this data source	STRING	N/A	%s			event
ta_trc.archive_info	archive_file_name	Archive file name	STRING	N/A	%s			event
ta_trc.archive_info	archive_status		BOOL4	N/A	%s	0	1	
ta_trc.archive_info	current_bytes_archived	Total bytes written to the current ark file	FLOAT8	bytes	%.0lf	0		
ta_trc.archive_info	total_bytes_archived	Total bytes written since this data source starts	FLOAT8	bytes	%.0lf	0		
ta_trc.archive_nsamples	current_bytes_archived_nsample		INT4	N/A	%d	0		event
ta_trc.archive_nsamples	ffi_aoi_bs_table_nsample		INT4	N/A	%d	0		event
ta_trc.archive_nsamples	ffi_control_table_nsample		INT4	N/A	%d	0		event
ta_trc.archive_nsamples	ffi_img_hist_table_nsample		INT4	N/A	%d	0		event
ta_trc.archive_nsamples	fpi_aoi_bs_table_nsample		INT4	N/A	%d	0		event
ta_trc.archive_nsamples	fpi_control_table_nsample		INT4	N/A	%d	0		event
ta_trc.archive_nsamples	fpi_img_hist_table_nsample		INT4	N/A	%d	0		event
ta_trc.archive_nsamples	total_bytes_archived_nsample		INT4	N/A	%d	0		event
ta_trc.archive_nsamples	trc_alignment_table_nsample		INT4	N/A	%d	0		event
ta_trc.archive_nsamples	trc_boresight_table_nsample		INT4	N/A	%d	0		event
ta_trc.archive_nsamples	trc_centroid_table_nsample		INT4	N/A	%d	0		event
ta_trc.archive_nsamples	trc_distribution_table_nsample		INT4	N/A	%d	0		event
ta_trc.archive_nsamples	trc_limb_table_nsample		INT4	N/A	%d	0		event
ta_trc.archive_nsamples	trc_main_aoi_table_nsample		INT4	N/A	%d	0		event
ta_trc.archive_nsamples	trc_process_info_nsample		INT4	N/A	%d	0		event
ta_trc.archive_nsamples	trc_rof_table_nsample		INT4	N/A	%d	0		event
ta_trc.archive_nsamples	trc_status_table_nsample		INT4	N/A	%d	0		event
ta_trc.archive_nsamples	trc_tascu_comm_table_nsample		INT4	N/A	%d	0		event
ta_trc.archive_nsamples	trc_temperature_table_nsample		INT4	N/A	%d	0		event
ta_trc.archive_nsamples	trc_tracking_table_nsample		INT4	N/A	%d	0		event
ta_trc.archive_nsamples	trc_watchdog_nsample		INT4	N/A	%d	0		event
ta_trc.archive_nsamples	wfi_aoi_bs_table_nsample		INT4	N/A	%d	0		event
ta_trc.archive_nsamples	wfi_control_table_nsample		INT4	N/A	%d	0		event
ta_trc.archive_nsamples	wfi_img_hist_table_nsample		INT4	N/A	%d	0		event
ta_trc.responses	send_packet	Last SCL response sent back to the issuer of the send_packet command	STRING	N/A	%s			event
ta_trc.responses	trc_aoi_clear	Last SCL response sent back to the issuer of the trc_aoi_clear command	STRING	N/A	%s			event
ta_trc.responses	trc_aoi_correction	Last SCL response sent back to the issuer of the trc_aoi_correction command	STRING	N/A	%s			event
ta_trc.responses	trc_aoi_define	Last SCL response sent back to the issuer of the trc_aoi_define command	STRING	N/A	%s			event
ta_trc.responses	trc_aoi_define_irf	Last SCL response sent back to the issuer of the trc_aoi_define_irf command	STRING	N/A	%s			event
ta_trc.responses	trc_aoi_define_irf_time	Last SCL response sent back to the issuer of the trc_aoi_define_irf_time command	STRING	N/A	%s			event
ta_trc.responses	trc_aoi_velocity	Last SCL response sent back to the issuer of the trc_aoi_velocity command	STRING	N/A	%s			event
ta_trc.responses	trc_bit	Last SCL response sent back to the issuer of the trc_bit command	STRING	N/A	%s			event
ta_trc.responses	trc_boresight_clear	Last SCL response sent back to the issuer of the trc_boresight_clear command	STRING	N/A	%s			event
ta_trc.responses	trc_boresight_define_pixel	Last SCL response sent back to the issuer of the trc_boresight_define_pixel command	STRING	N/A	%s			event
ta_trc.responses	trc_boresight_define_smarf	Last SCL response sent back to the issuer of the trc_boresight_define_smarf command	STRING	N/A	%s			event
ta_trc.responses	trc_boresight_define_tarf	Last SCL response sent back to the issuer of the trc_boresight_define_tarf command	STRING	N/A	%s			event
ta_trc.responses	trc_camera_integration	Last SCL response sent back to the issuer of the trc_camera_integration command	STRING	N/A	%s			event
ta_trc.responses	trc_camera_setup	Last SCL response sent back to the issuer of the trc_camera_setup command	STRING	N/A	%s			event

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
ta_trc.responses	trc_camera_speed	Last SCL response sent back to the issuer of the trc_camera_speed command	STRING	N/A	%s			event
ta_trc.responses	trc_camera_subframe	Last SCL response sent back to the issuer of the trc_camera_subframe command	STRING	N/A	%s			event
ta_trc.responses	trc_camera_sync	Last SCL response sent back to the issuer of the trc_camera_sync command	STRING	N/A	%s			event
ta_trc.responses	trc_camera_test	Last SCL response sent back to the issuer of the trc_camera_test command	STRING	N/A	%s			event
ta_trc.responses	trc_config_get	Last SCL response sent back to the issuer of the trc_config_get command	STRING	N/A	%s			event
ta_trc.responses	trc_config_load	Last SCL response sent back to the issuer of the trc_config_load command	STRING	N/A	%s			event
ta_trc.responses	trc_config_save	Last SCL response sent back to the issuer of the trc_config_save command	STRING	N/A	%s			event
ta_trc.responses	trc_config_set	Last SCL response sent back to the issuer of the trc_config_set command	STRING	N/A	%s			event
ta_trc.responses	trc_darkfield_acquisition	Last SCL response sent back to the issuer of the trc_darkfield_acquisition command	STRING	N/A	%s			event
ta_trc.responses	trc_darkfield_get	Last SCL response sent back to the issuer of the trc_darkfield_get command	STRING	N/A	%s			event
ta_trc.responses	trc_darkfield_load	Last SCL response sent back to the issuer of the trc_darkfield_load command	STRING	N/A	%s			event
ta_trc.responses	trc_darkfield_store	Last SCL response sent back to the issuer of the trc_darkfield_store command	STRING	N/A	%s			event
ta_trc.responses	trc_drift_limit	Last SCL response sent back to the issuer of the trc_drift_limit command	STRING	N/A	%s			event
ta_trc.responses	trc_filter_home	Last SCL response sent back to the issuer of the trc_filter_home command	STRING	N/A	%s			event
ta_trc.responses	trc_filter_select	Last SCL response sent back to the issuer of the trc_filter_select command	STRING	N/A	%s			event
ta_trc.responses	trc_flatfield_get	Last SCL response sent back to the issuer of the trc_flatfield_get command	STRING	N/A	%s			event
ta_trc.responses	trc_focus_auto	Last SCL response sent back to the issuer of the trc_focus_auto command	STRING	N/A	%s			event
ta_trc.responses	trc_focus_goto	Last SCL response sent back to the issuer of the trc_focus_goto command	STRING	N/A	%s			event
ta_trc.responses	trc_focus_move	Last SCL response sent back to the issuer of the trc_focus_move command	STRING	N/A	%s			event
ta_trc.responses	trc_focus_position	Last SCL response sent back to the issuer of the trc_focus_position command	STRING	N/A	%s			event
ta_trc.responses	trc_housekeeping_define	Last SCL response sent back to the issuer of the trc_housekeeping_define command	STRING	N/A	%s			event
ta_trc.responses	trc_inertial_set	Last SCL response sent back to the issuer of the trc_inertial_set command	STRING	N/A	%s			event
ta_trc.responses	trc_inertial_set_time	Last SCL response sent back to the issuer of the trc_inertial_set_time command	STRING	N/A	%s			event
ta_trc.responses	trc_init	Last SCL response sent back to the issuer of the trc_init command	STRING	N/A	%s			event
ta_trc.responses	trc_los_limit_move	Last SCL response sent back to the issuer of the trc_los_limit_move command	STRING	N/A	%s			event
ta_trc.responses	trc_los_limit_set	Last SCL response sent back to the issuer of the trc_los_limit_set command	STRING	N/A	%s			event
ta_trc.responses	trc_mode_positioning	Last SCL response sent back to the issuer of the trc_mode_positioning command	STRING	N/A	%s			event
ta_trc.responses	trc_mode_tracking_centroid_offset	Last SCL response sent back to the issuer of the trc_mode_tracking_centroid_offset command	STRING	N/A	%s			event
ta_trc.responses	trc_mode_tracking_centroid_offset_irf	Last SCL response sent back to the issuer of the trc_mode_tracking_centroid_offset_irf command	STRING	N/A	%s			event
ta_trc.responses	trc_mode_tracking_centroid_offset_irf_time	Last SCL response sent back to the issuer of the trc_mode_tracking_centroid_offset_irf_time command	STRING	N/A	%s			event
ta_trc.responses	trc_mode_tracking_centroid_offset_time	Last SCL response sent back to the issuer of the trc_mode_tracking_centroid_offset_time command	STRING	N/A	%s			event
ta_trc.responses	trc_mode_tracking_centroid_on_axis	Last SCL response sent back to the issuer of the trc_mode_tracking_centroid_on_axis command	STRING	N/A	%s			event
ta_trc.responses	trc_mode_tracking_limb	Last SCL response sent back to the issuer of the trc_mode_tracking_limb command	STRING	N/A	%s			event
ta_trc.responses	trc_mtbs_centroid	Last SCL response sent back to the issuer of the trc_mtbs_centroid command	STRING	N/A	%s			event
ta_trc.responses	trc_mtbs_inertial	Last SCL response sent back to the issuer of the trc_mtbs_inertial command	STRING	N/A	%s			event
ta_trc.responses	trc_mtbs_inertial_time	Last SCL response sent back to the issuer of the trc_mtbs_inertial_time command	STRING	N/A	%s			event
ta_trc.responses	trc_reticle_show	Last SCL response sent back to the issuer of the trc_reticle_show command	STRING	N/A	%s			event
ta_trc.responses	trc_rof_calculate	Last SCL response sent back to the issuer of the trc_rof_calculate command	STRING	N/A	%s			event
ta_trc.responses	trc_rof_calculate_irf	Last SCL response sent back to the issuer of the trc_rof_calculate_irf command	STRING	N/A	%s			event
ta_trc.responses	trc_rof_calculate_irf_time	Last SCL response sent back to the issuer of the trc_rof_calculate_irf_time command	STRING	N/A	%s			event
ta_trc.responses	trc_test_command	Last SCL response sent back to the issuer of the trc_test_command command	STRING	N/A	%s			event
ta_trc.responses	trc_test_image	Last SCL response sent back to the issuer of the trc_test_image command	STRING	N/A	%s			event

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
ta_trc.responses	trc_tweak_position_move	Last SCL response sent back to the issuer of the trc_tweak_position_move command	STRING	N/A	%s			event
ta_trc.responses	trc_tweak_position_set	Last SCL response sent back to the issuer of the trc_tweak_position_set command	STRING	N/A	%s			event
ta_trc.responses	trc_version_get	Last SCL response sent back to the issuer of the trc_version_get command	STRING	N/A	%s			event
ta_tsc								
ta_tsc	commands	Contains the last SCL command sent to the TASCU	STRING	N/A	%s			event
ta_tsc	rcs_id	Version information for the XML file that defines this data source	STRING	N/A	%s			event
ta_tsc.archive_info	archive_file_name	Archive file name	STRING	N/A	%s			event
ta_tsc.archive_info	archive_status		BOOL4	N/A	%s	0	1	
ta_tsc.archive_info	current_bytes_archived	Total bytes written to the current ark file	FLOAT8	bytes	%.0lf	0		
ta_tsc.archive_info	total_bytes_archived	Total bytes written since this data source starts	FLOAT8	bytes	%.0lf	0		
ta_tsc.archive_nsamples	current_bytes_archived_nsample		INT4	N/A	%d	0		event
ta_tsc.archive_nsamples	total_bytes_archived_nsample		INT4	N/A	%d	0		event
ta_tsc.archive_nsamples	tsc_mcs_hk_nsample		INT4	N/A	%d	0		event
ta_tsc.responses	send_packet	Last SCL response sent back to the issuer of the send_packet command	STRING	N/A	%s			event
ta_tsc.responses	tsc_atc_bias	Last SCL response sent back to the issuer of the tsc_atc_bias command	STRING	N/A	%s			event
ta_tsc.responses	tsc_atc_pos_corr	Last SCL response sent back to the issuer of the tsc_atc_pos_corr command	STRING	N/A	%s			event
ta_tsc.responses	tsc_atc_pos_set	Last SCL response sent back to the issuer of the tsc_atc_pos_set command	STRING	N/A	%s			event
ta_tsc.responses	tsc_atc_traj_est	Last SCL response sent back to the issuer of the tsc_atc_traj_est command	STRING	N/A	%s			event
ta_tsc.responses	tsc_auto_bal_control	Last SCL response sent back to the issuer of the tsc_auto_bal_control command	STRING	N/A	%s			event
ta_tsc.responses	tsc_bd_param_set	Last SCL response sent back to the issuer of the tsc_bd_param_set command	STRING	N/A	%s			event
ta_tsc.responses	tsc_bd_pos_abs	Last SCL response sent back to the issuer of the tsc_bd_pos_abs command	STRING	N/A	%s			event
ta_tsc.responses	tsc_bd_pos_rel	Last SCL response sent back to the issuer of the tsc_bd_pos_rel command	STRING	N/A	%s			event
ta_tsc.responses	tsc_bd_slew	Last SCL response sent back to the issuer of the tsc_bd_slew command	STRING	N/A	%s			event
ta_tsc.responses	tsc_bd_stop	Last SCL response sent back to the issuer of the tsc_bd_stop command	STRING	N/A	%s			event
ta_tsc.responses	tsc_cd_couple	Last SCL response sent back to the issuer of the tsc_cd_couple command	STRING	N/A	%s			event
ta_tsc.responses	tsc_cd_param_set	Last SCL response sent back to the issuer of the tsc_cd_param_set command	STRING	N/A	%s			event
ta_tsc.responses	tsc_cd_pos_abs	Last SCL response sent back to the issuer of the tsc_cd_pos_abs command	STRING	N/A	%s			event
ta_tsc.responses	tsc_cd_pos_rel	Last SCL response sent back to the issuer of the tsc_cd_pos_rel command	STRING	N/A	%s			event
ta_tsc.responses	tsc_cd_slew	Last SCL response sent back to the issuer of the tsc_cd_slew command	STRING	N/A	%s			event
ta_tsc.responses	tsc_cd_stop	Last SCL response sent back to the issuer of the tsc_cd_stop command	STRING	N/A	%s			event
ta_tsc.responses	tsc_fd_center	Last SCL response sent back to the issuer of the tsc_fd_center command	STRING	N/A	%s			event
ta_tsc.responses	tsc_fd_fbc_mode	Last SCL response sent back to the issuer of the tsc_fd_fbc_mode command	STRING	N/A	%s			event
ta_tsc.responses	tsc_fd_irf_velocity	Last SCL response sent back to the issuer of the tsc_fd_irf_velocity command	STRING	N/A	%s			event
ta_tsc.responses	tsc_fd_offset_lissajous	Last SCL response sent back to the issuer of the tsc_fd_offset_lissajous command	STRING	N/A	%s			event
ta_tsc.responses	tsc_fd_param_set	Last SCL response sent back to the issuer of the tsc_fd_param_set command	STRING	N/A	%s			event
ta_tsc.responses	tsc_fd_pos_abs	Last SCL response sent back to the issuer of the tsc_fd_pos_abs command	STRING	N/A	%s			event
ta_tsc.responses	tsc_fd_pos_abs_time	Last SCL response sent back to the issuer of the tsc_fd_pos_abs_time command	STRING	N/A	%s			event
ta_tsc.responses	tsc_fd_pos_rel	Last SCL response sent back to the issuer of the tsc_fd_pos_rel command	STRING	N/A	%s			event
ta_tsc.responses	tsc_fd_prog_bias	Last SCL response sent back to the issuer of the tsc_fd_prog_bias command	STRING	N/A	%s			event
ta_tsc.responses	tsc_fd_prog_lissajous	Last SCL response sent back to the issuer of the tsc_fd_prog_lissajous command	STRING	N/A	%s			event
ta_tsc.responses	tsc_fd_prog_offset	Last SCL response sent back to the issuer of the tsc_fd_prog_offset command	STRING	N/A	%s			event
ta_tsc.responses	tsc_fd_prog_offset_control	Last SCL response sent back to the issuer of the tsc_fd_prog_offset_control command	STRING	N/A	%s			event
ta_tsc.responses	tsc_fd_prog_track	Last SCL response sent back to the issuer of the tsc_fd_prog_track command	STRING	N/A	%s			event
ta_tsc.responses	tsc_fd_prog_track_control	Last SCL response sent back to the issuer of the tsc_fd_prog_track_control command	STRING	N/A	%s			event
ta_tsc.responses	tsc_fd_slew	Last SCL response sent back to the issuer of the tsc_fd_slew command	STRING	N/A	%s			event
ta_tsc.responses	tsc_fd_stop	Last SCL response sent back to the issuer of the tsc_fd_stop command	STRING	N/A	%s			event
ta_tsc.responses	tsc_fd_trc_offset	Last SCL response sent back to the issuer of the tsc_fd_trc_offset command	STRING	N/A	%s			event

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
ta_tsc.responses	tsc_high_rate_enable	Last SCL response sent back to the issuer of the tsc_high_rate_enable command	STRING	N/A	%s			event
ta_tsc.responses	tsc_mode_caged	Last SCL response sent back to the issuer of the tsc_mode_caged command	STRING	N/A	%s			event
ta_tsc.responses	tsc_mode_shutdown	Last SCL response sent back to the issuer of the tsc_mode_shutdown command	STRING	N/A	%s			event
ta_tsc.responses	tsc_mode_stab_inertial	Last SCL response sent back to the issuer of the tsc_mode_stab_inertial command	STRING	N/A	%s			event
ta_tsc.responses	tsc_mode_stab_local	Last SCL response sent back to the issuer of the tsc_mode_stab_local command	STRING	N/A	%s			event
ta_tsc.responses	tsc_mode_standby	Last SCL response sent back to the issuer of the tsc_mode_standby command	STRING	N/A	%s			event
ta_tsc.responses	tsc_mode_sw_interlock	Last SCL response sent back to the issuer of the tsc_mode_sw_interlock command	STRING	N/A	%s			event
ta_tsc.responses	tsc_ria_ground_mode	Last SCL response sent back to the issuer of the tsc_ria_ground_mode command	STRING	N/A	%s			event
ta_tsc.responses	tsc_test_command	Last SCL response sent back to the issuer of the tsc_test_command command	STRING	N/A	%s			event
ta_tsc.responses	tsc_vis_adjust	Last SCL response sent back to the issuer of the tsc_vis_adjust command	STRING	N/A	%s			event
ta_tsc.responses	tsc_vis_auto_control	Last SCL response sent back to the issuer of the tsc_vis_auto_control command	STRING	N/A	%s			event
ta_tsc.responses	tsc_vis_evacuate	Last SCL response sent back to the issuer of the tsc_vis_evacuate command	STRING	N/A	%s			event
ta_tsc.responses	tsc_vis_ground_adjust	Last SCL response sent back to the issuer of the tsc_vis_ground_adjust command	STRING	N/A	%s			event
ta_tsc.responses	tsc_vis_ground_mode	Last SCL response sent back to the issuer of the tsc_vis_ground_mode command	STRING	N/A	%s			event
ta_tsc.responses	tsc_vis_stop	Last SCL response sent back to the issuer of the tsc_vis_stop command	STRING	N/A	%s			event
ta_wfi								
ta_wfi	commands	Contains the last SCL command sent to the ta_wfi commandable	STRING	N/A	%s			event
ta_wfi	rcs_id	Version information for the XML file that defines this data source	STRING	N/A	%s			event
ta_wfi.archive_info	archive_file_name	Archive file name	STRING	N/A	%s			event
ta_wfi.archive_info	archive_status	[ENUM: (0: false) (1: true)]	BOOL4	N/A	%s	0	1	
ta_wfi.archive_info	current_bytes_archived	Total bytes written to the current ark file	FLOAT8	bytes	%.0lf	0		
ta_wfi.archive_info	total_bytes_archived	Total bytes written since this data source starts	FLOAT8	bytes	%.0lf	0		
ta_wfi.archive_nsamples	current_bytes_archived_nsample		INT4	N/A	%d	0		event
ta_wfi.archive_nsamples	total_bytes_archived_nsample		INT4	N/A	%d	0		event
ta_wfi.ccd_image_data	exposure_duration	Duration of exposure period	INT4	milliseconds	%d	1	10000	
ta_wfi.ccd_image_data	exposure_start_time	Start time of exposure	TIME8	seconds	%d.%09d	0	2147483647	
ta_wfi.ccd_image_data	filter_position	Filter position, [ENUM: (-1: UNDEFINED) (0: HOME) (1: CLOSED) (2: FILTER_DAY) (3: FILTER_ND1) (4: FILTER_ND2) (5: FILTER_ND3) (6: CLEAR) (12: FILTER_DAY_U) (13: FILTER_ND1_U) (14: FILTER_ND2_U) (15: FILTER_ND3_U) (16: FILTER_U) (22: FILTER_DAY_G) (23: FILTER_ND1_G) (24: FILTER_ND2_G) (25: FILTER_ND3_G) (26: FILTER_G) (32: FILTER_DAY_R) (33: FILTER_ND1_R) (34: FILTER_ND2_R) (35: FILTER_ND3_R) (36: FILTER_R) (42: FILTER_DAY_I) (43: FILTER_ND1_I) (44: FILTER_ND2_I) (45: FILTER_ND3_I) (46: FILTER_I) (52: FILTER_DAY_Z) (53: FILTER_ND1_Z) (54: FILTER_ND2_Z) (55: FILTER_ND3_Z) (56: FILTER_Z)]	INT4	N/A	%d	-1	56	
ta_wfi.ccd_image_data	focal_length	Focal length of camera optics	FLOAT4	millimeters	%f	100	100000	
ta_wfi.ccd_image_data	focus_position	Focus position. Only valid for FPI. Valid range is -300.0 to 300.0 (or -1000 if unknown).	FLOAT4	millimeters	%f	-300	300	
ta_wfi.ccd_image_data	image_size_transfer	Image size (in Bytes) of current image	INT4	N/A	%d	0	2147483647	
ta_wfi.ccd_image_data	image_sn	Unique identity of the image, with ranges as follows: WFI: 10000...19999, FFI: 20000...29999, FPI: 30000...39999	INT4	N/A	%d	10000	39999	
ta_wfi.ccd_image_data	image_type	Indicates whether the data block is a CCD image, a darkfield image or a flatfield image, [ENUM: (0: CCD_IMAGE) (1: FLATFIELD) (2: DARKFIELD) (3: TEST)]	INT4	N/A	%d	0	3	
ta_wfi.ccd_image_data	image_use	Reports how the image is used, [ENUM: (0: NONE) (1: MONITOR) (2: NORMAL) (3: DARK_ACQUISITION) (4: TEST)]	INT4	N/A	%d	1	4	
ta_wfi.ccd_image_data	packet	data packet as it was received from the TA	BINARY	N/A	%s			
ta_wfi.ccd_image_data	pixel_format	Pixel format of current image, [ENUM: (0: 8_BIT) (1: 14_BIT) (2: 16_BIT)]	INT4	N/A	%d	0	2	
ta_wfi.ccd_image_data	pixel_grouping	Pixel grouping of the current image, [ENUM: (0: 1x1) (1: 2x2) (2: 4x4)]	INT4	N/A	%d	0	2	

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
ta_wfi.ccd_image_data	pixel_n	Image pixel data. Each pixel is two bytes in size, so if the pixel_format is defined to be 14 bit, the two high bits (14 and 15) are always 0, whereas if the pixel_format is defined to be 8 bit, the high byte is filled with 0s. Also, for any given data packet, there are n pixels, i.e., the image size (see image_size_transfer) may vary with every packet. If the pixel_format is defined to be 16-bit, both bytes are fully used.	BINARY	N/A	%u			
ta_wfi.ccd_image_data	pkt_timestamp	time data packet was constructed by TA	TIME8	seconds	%d.%09d	0	2147483647	
ta_wfi.ccd_image_data	reticle	Reticle brightness. Only valid for FPI.	FLOAT4	N/A	%f	0	1	
ta_wfi.fits_data	fits	Contains the FITS image data (ASCII header followed by the image data in binary)	BINARY	N/A	%s			
ta_wfi.responses	debug	Last SCL response sent back to the issuer of the debug command	STRING	N/A	%s			event
ta_wfi.responses	image_to_hk	Last SCL response sent back to the issuer of the image_to_hk command	STRING	N/A	%s			event
ta_wfi.responses	log	Last SCL response sent back to the issuer of the log command	STRING	N/A	%s			event
ta_wfi.responses	retransmit_packets	Last SCL response sent back to the issuer of the retransmit_packets command	STRING	N/A	%s			event
ta_wfi.responses	show_test_images	Last SCL response sent back to the issuer of the show_test_images command	STRING	N/A	%s			event
ta_wfi.ta_wfi_alarm	ta_wfi_alarm_error	Error alarm	STRING	N/A	%s			event
ta_wfi.ta_wfi_alarm	ta_wfi_alarm_fatal	Fatal alarm	STRING	N/A	%s			event
ta_wfi.ta_wfi_alarm	ta_wfi_alarm_info	Information alarm	STRING	N/A	%s			event
ta_wfi.ta_wfi_alarm	ta_wfi_alarm_warning	Warning alarm	STRING	N/A	%s			event
ta_wfi.ta_wfi_alert	ta_wfi_alert_error	Error alert	STRING	N/A	%s			event
ta_wfi.ta_wfi_alert	ta_wfi_alert_fatal	Fatal alert	STRING	N/A	%s			event
ta_wfi.ta_wfi_alert	ta_wfi_alert_info	Information alert	STRING	N/A	%s			event
ta_wfi.ta_wfi_alert	ta_wfi_alert_warning	Warning alert	STRING	N/A	%s			event
wvm_if								
wvm_if	commands	Contains the last SCL command sent to the WVM	STRING	N/A	%s			event
wvm_if	pitch_angle	Current pitch angle of the aircraft	FLOAT4	degrees	%f			1
wvm_if	pressure_alt	Barometric altitude	FLOAT4	feet	%f			1
wvm_if	rcs_id	Version information for the XML file that defines this data source	STRING	N/A	%s			event
wvm_if	roll_angle	Current roll angle of the aircraft	FLOAT4	degrees	%f			1
wvm_if	roll_angle_sampletime	Time roll angle was measured by the DAS	FLOAT8	seconds	%.3ymdz			1
wvm_if	static_air_temp	Static air temperature	FLOAT4	degrees Celsius	%f			1
wvm_if	water_vapor_along_ta_los	Calculated water vapor along the current telescope line of sight	FLOAT8	micrometers	%f			1
wvm_if	water_vapor_calc_ta_el	Telescope elevation used for water vapor calculation	FLOAT8	degrees	%f			1
wvm_if.archive_info	archive_file_name	Archive file name	STRING	N/A	%s			event
wvm_if.archive_info	archive_status	[ENUM: (0: false) (1: true)]	BOOL4	N/A	%s	0	1	
wvm_if.archive_info	current_bytes_archived	Total bytes written to the current ark file	FLOAT8	bytes	%.0lf	0		
wvm_if.archive_info	total_bytes_archived	Total bytes written since this data source starts	FLOAT8	bytes	%.0lf	0		
wvm_if.archive_nsamples	current_bytes_archived_nsample		INT4	N/A	%d	0		event
wvm_if.archive_nsamples	total_bytes_archived_nsample		INT4	N/A	%d	0		event
wvm_if.responses	calibrate_los	Last SCL response sent back to the issuer of the calibrate_los command	STRING	N/A	%s			event
wvm_if.responses	exit	Last SCL response sent back to the issuer of the exit command	STRING	N/A	%s			event
wvm_if.responses	go_to_operate	Last SCL response sent back to the issuer of the go_to_operate command	STRING	N/A	%s			event
wvm_if.responses	go_to_standby	Last SCL response sent back to the issuer of the go_to_standby command	STRING	N/A	%s			event
wvm_if.responses	quit	Last SCL response sent back to the issuer of the quit command	STRING	N/A	%s			event
wvm_if.responses	run_bit_tests	Last SCL response sent back to the issuer of the run_bit_tests command	STRING	N/A	%s			event
wvm_if.responses	set	Last SCL response sent back to the issuer of the set command	STRING	N/A	%s			event
wvm_if.responses	status	Last SCL response sent back to the issuer of the status command	STRING	N/A	%s			event
wvm_if.wvm_if_alarm	wvm_if_alarm_error	Error alarm	STRING	N/A	%s			event

Prefix	Value Name	Description	Representation	Units	Format	Low Limit	High Limit	Data Rate (Hz)
wvm_if.wvm_if_alarm	wvm_if_alarm_fatal	Fatal alarm	STRING	N/A	%s			event
wvm_if.wvm_if_alarm	wvm_if_alarm_info	Information alarm	STRING	N/A	%s			event
wvm_if.wvm_if_alarm	wvm_if_alarm_warning	Warning alarm	STRING	N/A	%s			event
wvm_if.wvm_if_alert	wvm_if_alert_error	Error alert	STRING	N/A	%s			event
wvm_if.wvm_if_alert	wvm_if_alert_fatal	Fatal alert	STRING	N/A	%s			event
wvm_if.wvm_if_alert	wvm_if_alert_info	Information alert	STRING	N/A	%s			event
wvm_if.wvm_if_alert	wvm_if_alert_warning	Warning alert	STRING	N/A	%s			event
wvm_if.wvmack	ackid	This is normally an echo of the wvmcmd.ackid value. However, in those cases where the Water Vapor Monitor autonomously changes state, then wvmack.ackid=-1, in the case of a command that is not valid in the current state, then wvmack.ackid=-2, and in the case of an unknown command, then wvmack.ackid=-3.	INT4	N/A	%d			
wvm_if.wvmack	command_ack	For valid commands, this acknowledges the command string that was sent to the Water Vapor Monitor from the MCCS	STRING	N/A	%s			
wvm_if.wvmbit	condition	BIT condition of the WVM- nominal, warning, or fatal	STRING	N/A	%s			
wvm_if.wvmbit	time	Time that BIT information was updated by the Water Vapor Monitor in seconds since 1970	FLOAT8	seconds	%lf			
wvm_if.wvmcmd	ackid	Command tracking number assigned by MCCS	INT4	N/A	%d			
wvm_if.wvmcmd	command	Command name	STRING	N/A	%s			
wvm_if.wvmdata	ch_1	Measured brightness temperature in the WVM radiometer Channel 1, averaged over the previous 15 seconds	FLOAT4	kelvins	%.3f			
wvm_if.wvmdata	ch_2	Measured brightness temperature in the WVM radiometer Channel 2, averaged over the previous 15 seconds	FLOAT4	kelvins	%.3f			
wvm_if.wvmdata	ch_3	Measured brightness temperature in the WVM radiometer Channel 3, averaged over the previous 15 seconds	FLOAT4	kelvins	%.3f			
wvm_if.wvmdata	ch_4	Measured brightness temperature in the WVM radiometer Channel 4, averaged over the previous 15 seconds	FLOAT4	kelvins	%.3f			
wvm_if.wvmdata	ch_5	Measured brightness temperature in the WVM radiometer Channel 5, averaged over the previous 15 seconds	FLOAT4	kelvins	%.3f			
wvm_if.wvmdata	ch_6	Measured brightness temperature in the WVM radiometer Channel 6, averaged over the previous 15 seconds	FLOAT4	kelvins	%.3f			
wvm_if.wvmdata	status	Status of the WVM	STRING	N/A	%s			
wvm_if.wvmdata	time	Time of wvmdata update, measured by the WVM CPU in seconds since 1970	FLOAT8	seconds	%lf			
wvm_if.wvmdata	water_vapor	Integrated precipitable water vapor to the zenith, running average of previous 60 seconds	FLOAT4	micrometers	%.3f			
wvm_if.wvmerror	errid	Integer identifying the occurrence of a fault so that the corresponding error flag can be cleared when the fault has been corrected	INT4	N/A	%ld			
wvm_if.wvmerror	level	Error level for the given error: warning, fatal, or cleared	STRING	N/A	%s			
wvm_if.wvmerror	text	Descriptive text string describing the error reported by the Water Vapor Monitor	STRING	N/A	%s			
wvm_if.wvmerror	time	Time that the error was detected by Water Vapor Monitor in seconds since 1970	FLOAT8	seconds	%lf			

APPENDIX C. DIGITAL VIDEO DISTRIBUTION SUBSYSTEM

The Digital Video Distribution Subsystem (DVDS) is responsible for acquiring observatory video data, distributing it in real-time to workstations and user-provided computers, and recording the data. DVDS is composed of the TA Imager Processing Subsystem (TAIPS), and Video Processing and Recording Subsystem (VPARS).

Section C-1 describes the functions and interfaces provided by TAIPS.

Section C-2 describes the functions and interfaces provided by VPARS.

C-1. TA Imager Processing Subsystem (TAIPS)

1.0 Background and Scope

The TA Imager Processing Subsystem (TAIPS) is a subset of the MCCS Digital Video Distribution Subsystem (DVDS). Its general function is to interface with the SOFIA TA Tracker to collect the Wide Field Imager (WFI), Fine Field Imager (FFI), and Focal Plane Imager (FPI) images. It streams these images as raw binary data to various workstations and hosts connected at the PI patch panel. The delivery method of the data is UDP packets in multicast mode via the Video LAN. The raw data may then be further processed and/or displayed by the receiving systems. The TAIPS is a VME based system that runs under the VxWorks Operating System. It consists of three dual processor Single Board Computers (SBCs) in a rugged chassis. Figure C-1 shows TAIPS external interfaces. Note that numbers on lines connecting to LAN boxes on the diagram are in Mbits per second.

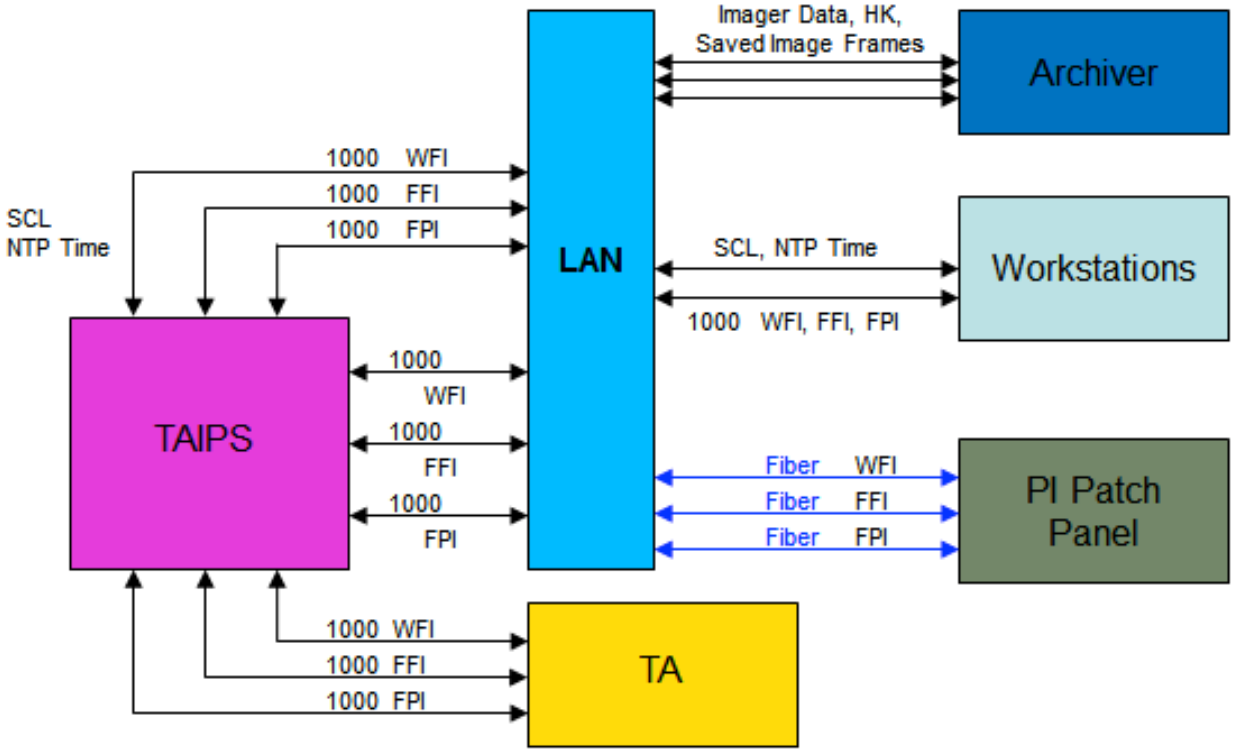

Figure C-1. TAIPS Interfaces

TAIPS provides three products for users.

1) Real-time WFI, FFI and FPI images broadcast over the Video LAN. See Section 2.1 for details.

2) Logging of individual images or sequences of images from WFI, FFI and FPI when commanded via SCL. The logged image files are available in-flight as they are acquired. See Section 2.2.
3) An archive of all images acquired in-flight. This data is independent of the logging of individual or sequences of frames mentioned in (2) above. The image archive may be accessed on-board SOFIA from the MCCS Archiver. It is also available post-flight in the data archives maintained by the science project. See Section 2.3 and Appendix F for further information.

2.0 General Functions

- The TAIPS interfaces to the TA Tracker to receive imager data from the TA Tracker's imagers (WFI, FFI and FPI) via TCP/IP communication per the TA_MCCS_F.
- The target data rate is 18 MB/s maximum per imager (corresponding to 9Hz for 1024 x 1024 resolution setting). Actual rates will increase from the 2 MB/s rate as performance improvements are made.
- All data communication is via network connections as depicted in Figure C-1.
- The TAIPS timestamps each housekeeping message that it receives from the TA Tracker.
- The TAIPS provides all TA Tracker housekeeping data specified in section 30.6 of Appendix C of the TA_MCCS_F ICD as MCCS housekeeping data except the "pixel_n" item (which is the image itself).
- The TAIPS provides the status/state of the TCP/IP connection between the TA and the TAIPS as MCCS housekeeping data.

2.1 Streaming Image Data

- The TAIPS streams the received raw data of the WFI, FFI and FPI to UDP multicast addresses on the Video LAN in the exact format that it receives it. The structure of the raw data packets received from the TA imagers is described in detail in the TA_MCCS_F ICD, Appendix B and section 30.6 of Appendix C.
- Each raw data packet is followed by the 4-byte synchronization value 0xfe6b2840. All values, including the synchronization value are in network (big-endian) byte order.
- Each imager has its own multicast IP address and port number. The TAIPS uses the UDP packet size, multicast address, and port number specified in the table below for the specified imager streams.

	WFI	FFI	FPI
Packet size (in bytes)	8192	8192	8192
Multicast address	230.5.6.7	230.5.6.9	230.5.6.8
Port Number	8234	8236	8235

2.2 Image Logging

- The TAIPS saves the next single image frame or series of frames to an NFS disk upon receiving a “ta_wfi.log”, “ta_ffi.log” or “ta_fpi.log” SCL command. The SCL command specifies the format of the saved file as either “FITS” or “Binary” (binary is a dump of the exact raw data to a file).
- The TAIPS only timestamps in a saved file if it is in FITS format. The binary format only contains the TA timestamps, and is not time-stamped by the TAIPS.
- When saving a series of frames, a new file is created for each image requested.
- For image capture files, the user specifies the filename in the SCL log command. When a series of images is requested, an incrementing number is appended to the filename.
- When logging an image with the same name as a previously saved file, the file will be overwritten by the new image file.

The following sample FITS header shows keywords provided by MCCS in TAIPS image files logged in FITS format. Note that keywords not common to the FITS standard are defined in the TA_MCCS_F ICD.

```
SIMPLE  =                    T / standard conforming (constant T)
BITPIX  =                   16 / bits per pixel (constant 16 for current imagers)
NAXIS   =                    2 / number of axes
NAXIS1  =                 1024 / image width
NAXIS2  =                 1024 / image height
ORIGIN  = 'SOFIA'              / Stratospheric Observatory for Infrared Astronomy
DATE    = '2009-07-22T08:02:49.635Z' / When user requested image
HISTORY   Created by the SOFIA Mission Control System software
DATASRC = 'ASTRO'              / data source ('TEST' for test data, 'ASTRO' for real)
CREATOR = 'sessiond'           / MCCS session daemon
SCLUSER = ‘<user_name>’  / SCL login name of user requesting image
SCLROLE = ‘<user_role>’  / SCL login role of user requesting image
SCLCOMNT= ‘<comment>’    / SCL log command comment string (or ‘’ if none)
FILENAME= 'ta_operator_ws_1.wfi_image.1.fits' / file name (excludes path)
OBSERVAT= 'SOFIA'              / Stratospheric Observatory for Infrared Astronomy
TELESCOP= 'SOFIA'              / 2.5 meter telescope
IMAGTYP = 'CCD_IMAGE'          / image type ('CCD_IMAGE', 'DARKFIELD' or 'FLATFIELD')
INSTRUME= 'FPI'                / 'FPI', 'FFI' or 'WFI'
DATE-OBS= '2009-07-22'         / date of exposure start
UTCSTART= '08:02:50.125Z' / UTC of exposure start
UTCEND  = '08:02:51.325Z' / UTC of exposure finish
EXPTIME =               1.200 / seconds
BINNING = '1x1'                / pixel grouping (='1x1', '2x2' or '4x4')
```

```
PIXBITS =             14 / ADC bits (=8, 14, or 16)
DATAMAX =          16383 / detector full well maximum (=2^PIXBITS - 1)
DATAMIN =              0 / smallest value from detector
BZERO   =   32768.000000 / zero point to adjust pixel values
IMAGE_SN=          30218 / unique image identifier (WFI: 10000..19999, FFI:
                           20000...29999,FPI: 30000...39999)
IMAGUSE = 'NORMAL'       / how image is used ('NORMAL', 'MONITOR',
                           'DARK_ACQUISITION', 'TEST')
FILTPOS = 'CLOSED'       / filter position
FOCLEN  = 21528.3        / focal length of camera optics in mm (100 to 100000)
KWDICT  = ‘DCS SI 01 A’  / conforms with DCS keyword dictionary version
FPIFOCUS=         126.42 / focus position in mm (-300 to 300, or -1000 if
                           unknown) - only valid for FPI
FPIRETCL=             -1 / reticle brightness (0.0 to 1.0, or -1 if undefined) -
                           only valid for FPI
TELRA   = '05:35:28.0'   / RA of center of image (always J2000)
TELDEC  = '-05:26:20'    / declination of center of image (always J2000)
TELROF  =         128.21 / ROF of this image (clockwise from up to north)
END
```

2.3 Image Archival

- The TAIPS archives all raw imager data continuously as it is received for in-flight and post flight review purposes.
- Multiple archive files will be generated, each with a maximum size of 50 Mbytes. (This size is set in the TAIPS configuration XML file.)
- The TAIPS timestamps the archived data.
- Archive data format is defined in Appendix F.

3.0 Workstations, Science Instrument (SI) and Other Interfaces

- Workstations and SI applications may receive streaming raw imager data by joining the multicast groups specified in the table in section 2.1 for the WFI, FFI, and FPI via their socket communication software.
- Workstations and SI applications are responsible for receiving, processing and displaying imager data from the TAIPS as desired since TAIPS does not provide these functions.
- The physical interface for SI computers to the TAIPS for raw imager data is the PI Patch Panel and the Video LAN.

C-2 Video Processing and Recording Subsystem (VPARS)

1.0 Background and Scope

The Video Processing and Recording Subsystem (VPARS) is a subset of the MCCS Digital Video Distribution Subsystem (DVDS). Its general function is to distribute and record video from sources around the aircraft, including cameras and workstation display activity. Each source is encoded and streamed via the network to users and systems on board. Each stream is captured and recorded in a digital video file format. Users will view live streaming video via players on their computers and video displays around the aircraft.

VPARS provides three products for users.

1) Real-time digital video broadcast over the Video LAN.
2) An archive of all video acquired in-flight (each channel can be selected for recording separately). The digital video files may be accessed on-board SOFIA from the MCCS Archiver, and post-flight in the data archives maintained by the science project.
3) External video sources supplied to the VPARS will be accepted and managed in the same manner as the standard DVDS sources. Bandwidth for two such channels has been set aside specifically for SI use.

The following interface information is TBD:

- file naming convention
- number of channels per file
- file size
- data rates
- compression ratio
- how time is handled

2.0 General Functions

- VPARS interfaces with the workstation displays to capture and encode what is seen on each display.
- VPARS includes situational awareness cameras covering mission areas, TA, TA cavity and aft cavity.
- All video is distributed via network connections.
- Two channels of video have been designated for SI use.

2.1 Streaming Video

- Viewers will be able to access streaming video via the MCCS network. Users with Windows (and some Mac) computers can use the integrated player for viewing. Others will use separate player software (such as VLC). Channel details will be provided, including IP address, port, and protocol.
- Viewers' computers are responsible for receiving, and decoding streaming video as desired since VPARS cannot provide these functions. Encoded video will be provided on the network. Only standard encoding formats will be used (H.264 being the most common).

2.2 Video Archiving

- VPARS saves selected video streams to the archive in standard video file formats. Selection of which streams are recorded is made through the DVDS server's web based GUI software. Initial configuration will be to record all video streams.
- Files may be segmented depending on recording configuration in 3-1440 minute segments.
- Recorded video will be available from the Archiver subsystem after that video segment is written.

2.3 External Video Interface to VPARS

- DVDS-VPARS will accept encoded video via the network in standard video formats. MPEG-4 Part-10 H.264 is recommended, but MPEG-2, MPEG-4 are accepted. Other standards may be accepted but require collaboration efforts between external developer and DVDS subsystem support staff.
- Video should be streamed via multicast from the external source to DVDS-VPARS.
- A static IP address, port, and protocol will be assigned to external sources. The use of SAP (Session Announcement Protocol) is required as a means of connecting the external video with the DVDS-VPARS. The SAP multicast address and port will be provided as needed, defaulted to (IP - 224.2.127.254; Port - 9875).
- The external video source generator should consider data rate when determining the video source configuration. Always use the minimum required settings to accomplish the needs of the video in order to reduce the network traffic, impact to viewers and recorded video file size.

APPENDIX D. MISSION AUDIO DISTRIBUTION SUBSYSTEM

The Mission Audio Distribution Subsystem (MADS) is the primary system that enables multiple channel audio communication among personnel onboard the aircraft. The flight deck uses MADS to talk to the mission and science crews in the main cabin, as well as with the ground crews for maintenance and ground operations; and the mission and science crews use MADS to communicate with each other and with the passengers, such as EPO members and other guests, during the flight.

The MADS allows for multiple, separate conferences that enable non-blocking audio communication between discrete groups to allow them to operate without distractions and/or interference. The MADS interfaces to the existing flight crew, cabin, service inter-phone subsystems, and the public address (PA) subsystem. The MADS also provides intercom stations located throughout the aircraft. The key functional components of MADS include the following:

- MADS Headset: The MADS headset is an aviation noise cancelling talk-listen stereo headset, which connects to an ACP (see below). The headset provides a volume control for each ear speaker, has a certified noise reduction rating of 23 dB, and provides an advanced noise-cancelling microphone.

- Audio Connection Panel (ACP): This small box, to which the operator can connect up to two headsets, performs the analog-to-digital and digital-to-analog conversion of the audio signal moving to and from the headsets.

- MADS Control Panel (MCP): The MCP provides a user-selectable button-based interface for operation of the system, including selection of audio channels and modes (e.g., conference on one channel while monitoring many others), and adjustment and selection of various optional functions and settings. The overall MADS system contains multiple MCPs, each operating independently and without interfacing to the actual audio signals.

- *MADS Audio Recorder: This is a commercial, rack-mounted, multiple-input, digital voice recorder, which can simultaneously record up to 24 audio channels; channels can be de-selected, if/as desired. It features digital signal processing to improve voice intelligibility, and stamps the audio recordings with standard observatory date and time information as provided by the NTP server.*

MADS is required to record, playback, and archive time-stamped audio channels. The recordings are compressed into MP3-format files and sent over a network connection in real-time to be stored on the Archiver. For additional details on the MADS capabilities and operation see the *MADS System Description Document (APP-DF-HWCI-SE51-2020).*

Operationally the MADS is configured and initialized during pre-flight and then records audio data throughout the flight. The standard configuration is to use twelve audio channels: ten conference channels and one channel each for the all-call function and for audio alerts and

alarms. Each seat or station on the aircraft is equipped with at least one headset that provides access to the channels available from the controlling MCP, which defines who can talk and/or listen on which channels. Each MCP also provides a "local" channel that enables semi-private conversations with the other users in a pre-defined functional group (e.g., SI Team). The interface with the aircraft PA provides MADS the capability to accept the public address audio signal, and broadcast that to all headsets at 6 dB above all other headset audio signals, regardless of channel volume level settings. During the flight the operators can view the health and status of the MADS via housekeeping provided by the PIS. *The MADS also generates and broadcasts audio alerts and alarms as triggered by the PIS. At the end of the flight the resulting audio files are off-loaded from the Archiver and transferred into the Science Project archive for access by users.*

The following interface information is TBD:

- *file naming convention*
- *number of channels per file*
- *file size*
- *data rates*
- *compression ratio*
- *how time is handled*

APPENDIX E. ERRATA

Table E-1 lists known errata in this version of the ICD. These will be addressed in future revisions. Information on other planned improvements to the ICD can be made available upon request.

Table E-1: Errata Items.

Date Discovered	Errata Description	ICD Impact
	[Place holder row]	

APPENDIX F MCCS FILE FORMATS

APPENDIX F.1 ARCHIVE FILE

1. Overview

This appendix specifies the format of MCCS archive files, also known as “ark” files. This is the format of MCCS housekeeping archive files, TA image archive files, and any other SOFIA files ending with the file extension “.ark”.

2. File Naming Conventions

File name is <hostname>.<subsystem>.<timestamp>.ark, where:

- <hostname> name of host computer where the subsystem is running that created the archive file.
- <subsystem> name of subsystem that created the archive file. This is the same as name in the <subsystem>_data.xml file.
- <timestamp> when the archive file is first created, is has the format of YYMMDDhhmmss.

Examples:

```
das_proxy.das.140613134022.ark
ffi_node_b.ta_ffi.140612231815.ark
```

The naming convention was established for archiving data from all MCCS data sources. Therefore MCCS names its files consistently with the data naming defined for SCL commands, and requires that this consistency be maintained in order for the software to work. This naming convention extends to the archive file names which map one-to-one with the XML data definition file names.

For example, the MCCS Data Acquisition Subsystem (DAS) is an MCCS data source with a base data tree name of "das". Its XML data definition file is named das_data.xml, and its archive file name will be hostname.das.timestamp.ark (all archive files will have the host machine where they were generated prepended on to the file name. This is necessary since a given data source may run on multiple machines. Note that a given machine will not have more than one data source with the same base name running on it). This works the same for any subsystem that the MCCS expects data from, including MCCS internal sources (e.g. Coordinate Transforms). You just need to know the base name for that source/subsystem and all the associated file names are built based on that. Most of these names are defined in the Housekeeping Data List at the front of Appendix B in this ICD. Note that for user (session) sources, the base name for the source is dynamically determined, so the name of the archive file is not known initially. The base name of a user source is determined when the user logs in to the system. The naming convention for user sources is "session_user_role_id", (e.g., "session_mcs_co_1"), where "user" and "role" correspond to the user and role specified in

the login command, while id is a running count of the number of user sessions created on a given machine. All session archive files have the same XML data definition, which is defined in session_data.xml.

3. Life Cycle of the Archive file

Before a flight, archive storage areas are cleared of enough previous data to assure space for data from the upcoming flight. Subsequently, MCCS data sources write data into archive files on the MCCS Archiver subsystem. When a file is closed due to either reaching a configurable max size, or by a user command “stop_archive”, or when the subsystem shuts down, the ending time is added to the end the archive file.

Following the flight, ground support staff copy the flight’s data products to ground-based disks, and send the data (via network or shipped on physical media) to Ames where it is archived for later access by staff and users.

4. Data Structures and Definitions

A data source has an XML definition of the data structure that the associated MCCS data handling process (proxy) uses to create its part of the data tree and to populate and distribute the data for each data update. Each data source maps to one subsystem and has only one XML data definition file. That XML definition is a pre-defined tree structure containing four types of nodes that are of primary interest to the archive: DataNodes, ArrayNodes, Values, and AlertValues. Other informational tags may be specified in the XML definition (such as "help" tags, FieldValues, and AlertItems), but these tags do not represent nodes in the MCCS data tree. Here is a basic breakdown of the rules for how the four nodes mentioned above may be organized in the MCCS tree structure:

- DataNodes – Any of the four node types may be a child of a DataNode. The top-level node for a data source will always be a DataNode (which means the top-level node in the XML definition will always be a DataNode). DataNodes will not be nested within DataNodes more than three deep (i.e., a given DataNode may only have children, grandchildren, and great-grandchildren DataNodes). DataNodes do not represent an actual data value.

- ArrayNodes – DataNodes, Values, and AlertValues may all be children of an ArrayNode.

ArrayNodes cannot be descended from ArrayNodes at any level (i.e., an ArrayNode cannot be nested within an ArrayNode, whether it is as a child, grandchild, etc.). Also, an ArrayNode may only contain one DataNode as an immediate child. ArrayNodes do not represent an actual data value.

- Values – Of the four node types, only Values may be children of Values. However, a given Value node may only have one Value node as a child, and that child will not have any children of its own. Value nodes represent actual data values in the data tree. Values may also have FieldValues as children, which may or may not be of any interest to the archive. FieldValues represent enumerations or bit mappings that are associated with the

Value node's data value (e.g., a value of 5 may represent the string "CMD_EXECUTED").

- AlertValues – Of the four node types, only Values may be children of AlertValues. However, a given AlertValue node may only have one Value node as a child, and that child will not have any children of its own. Similar to Value nodes, AlertValues represent actual data values in the data tree. However, these data values are different in that they represent alert/alarm messages, which have their own unique format (as defined in the MCCS_SI_04 section on Alerts and Alarms). AlertValues may also have AlertItems as children, which are probably not of any interest to the archive. AlertItems are used by the MCCS to define persistent alerts/alarms.

Data is updated as data groups (dataGroup), which are defined as the set of data items that are coincidentally/simultaneously received (and time-stamped) from a source. Data groups are marked in the XML definition by setting the dataGroup attribute to "true" for a given node. A given data source can have one or more data groups. If the dataGroup attribute is not specified for a given node, it is defaulted to a value of "false". Since a dataGroup can be defined ("true") at any level within the tree, the data group may be an entire tree structure of its own or only a single node, depending on how the data set is received from the source. Note that data groups may not be nested within data groups; once dataGroup="true" has been defined for a given node in the tree, none of its descendents (children, grandchildren, etc.) may have dataGroup="true" defined for them. If dataGroup="true" is specified for a given node, it will always contain a child Value node that is defined to be the MCCS timestamp ("mcstime") for the data group. This "mcstime" node indicates when the MCCS received the data group. It will have an attribute defined in it called "attribute", which will be set to the value "yes". This is meant to indicate that the Value node provides extra information about the parent node, and that its contents apply to all children of the parent. Please also note that the "mcstime" node will never have any children of its own.

The data group attribute may only be associated with the four primary node types listed above (DataNodes, ArrayNodes, Values, and AlertValues). Data groups are related to the four primary node types as follows:

- DataNodes – DataNodes may or may not be specified as a data group. When a DataNode is designated as a data group, then its children, and possibly its grandchildren, will contain the data items for the data group. In most cases, however, all the data items in the data group will be children of the DataNode.
- ArrayNodes – ArrayNodes are always specified as a data group. Its children, grandchildren, and possibly its great-grandchildren will contain the data items for the

data group. In most cases, however, all the data items in the data group will either be children or grandchildren of the ArrayNode.

- Values – Values may or may not be specified as a data group. When a Value is designated as a data group, the Value node itself as well as its children nodes will contain the data items for the data group.
- AlertValues – AlertValues may or may not be specified as a data group. When an AlertValue is designated as a data group, the AlertValue node itself as well as its children nodes will contain the data items for the data group.
- Please note that the actual data items in a data group will always either be Value or AlertValue nodes. The data definition is not repeated in the archive files themselves, so the "unpacking" software will need to utilize the same definition files (XML) used by the MCCS data distribution and archiving function to understand the structure of the data written to the MCCS archive. The primary data definition parameters (XML tags/attributes) needed to derive the structure are listed below.
    - DataNode, ArrayNode, Value, and AlertValue tags
    - The dataGroup attribute
    - Value name
    - Value type and size

Any archive retrieval software should be able to use these tags and attributes and their sequence and nesting given in the XML file to determine the complete structure for the given data source. Here are two examples of XML structure for data definitions that correspond to a portion of the data from the FORCAST science instrument and TA TASCU data sources.

The sample below (from forcast_data.xml) is for the FORCAST science instrument in a section of the definition where every value is its own dataGroup because the MCCS receives each data item individually.

```
<DataNode name="forcast" label="FORCAST Science Instrument"
      description="Initial XML Instrument Mode Description File"
      rcs_id="$Id: forcast_data.xml,v 1.3 2009/08/11 12:13:31 corteze Exp $"
    .
    .
    .
   <Value name="x_sep_si" rep="FLOAT8" initial="50.0" units="microns"
          label="SI X pixel separation" description="SI x-axis pixel separation"
          dataGroup="true" format="%lf">
      <Value name="mcstime" rep="FLOAT8" units="seconds" label="MCS Timestamp"
            format="%lf" attribute="yes"
            description="Time the data arrived at MCS (seconds since 1970)"/>
    </Value>
    <Value name="y_sep_si" rep="FLOAT8" units="microns" initial="50.0"
          label="SI Y pixel separation" description="SI y-axis pixel separation"
```

```
          dataGroup="true" format="%lf">
       <Value name="mcstime" rep="FLOAT8" units="seconds" label="MCS Timestamp"
             format="%lf" attribute="yes"
             description="Time the data arrived at MCS (seconds since 1970)"/>
    </Value>
    .
    .
    .
</DataNode>
```

For the TA's TASCU subsystem (ta_tsc_data.xml) the entire block of data Values is one dataGroup sent to the MCCS in one tsc_cmd_ack message from the TASCU.

```
<DataNode
       name="ta_tsc"
       label="TA TASCU data"
       description="Data produced by the TASCU"
       distributor="true"
       dataGroup="false"
       rcs_id ="$Id: ta_tsc_data.xml,v 1.17 2001/11/19 17:32:48 sschlappe Exp $">
   <DataNode
          name = "tsc_cmd_ack"
          description = "generic acknowledgement to a TASCU command"
          msgID = "4001"
          msgVersion = "1"
          dataGroup = "true"
          developmentStatus = "B3"
          visibility = "PUBLIC">
       <help>
       <helptext>
              The TSC_CMD_ACK is sent from the TSC to the MCS indicating that a
              command was received, accepted, or contained a sequence count error.
       </helptext>
       </help>
       <Value
              name="mcstime"
              description="time MCS received data"
              units="seconds" rep="FLOAT8"
              attribute="yes"
              format="%lf" label="MCS Timestamp">
       </Value>
       <Value
              name="packet"
              description="data packet as it was received from the TA"
              units="" rep="BINARY"
              lolim="" hilim=""
              accuracy="" precision=""
              format="%s" label="PACKET"
              >
       </Value>
       .
       .
       .
   </DataNode>
</DataNode>
```

These two examples show some variants in "style" with the use of both <value ... /> and <value> ... </value> for the value tags. These and other variations in language usage are enforced by their validation against DTDs and style sheets.

5. Archive records

The binary records in each archive file have a variable length with a fixed structure (sequence of elements), with the record element lengths or terminations given in the table below. All fields in header, ender, and data records are in big-endian byte order.

Archive Header	Record #1	. . .	Record #n	Archive Ender

An Archive Header is defined as:

Archive component	Component element	Contents	Length	Terminator
Archive Header	Xml file size	Size of the entire XML file (bytes)	4 bytes	no term
	Xml file	The XML file that defines the data in this archive file subsystem_data.xml	variable	no term

Each data record is defined as:

Archive component	Component element	Contents	Length	Terminator
Data Record Header	Sync Word	Sync Word Hex byte pattern 0x1FDFA7C9	4 bytes	no term
	Record size(bytes)	Size of the entire record (header w/sync word + data) (bytes)	4 bytes	no term
	Timestamp	When the record was written to the file(floating-point value denoting seconds since 1970)	8 bytes	no term
	Address	ASCII SCL syntax data group address, e.g., coord.pos	variable	NULL ('\0')
Data Group Serialized	Value #1	Value of first node in the data group	length of item #1 (**)	no term
	Value #2	Value of second node in the data group	length of item #2 (**)	no term
	. . .	. . .	. . .	. . .
	Value #N	Value of last node in the data group	length of item #N (**)	no term

An Archive Ender is defined as:

Archive component	Component element	Contents	Length	Terminator
Archive Ender	Sync Word	Sync Word Hex byte pattern 0x1FDFA7C9	4 bytes	no term
	Record size(bytes)	Size of the entire record (header w/sync word + data) (bytes)	4 bytes	no term
	Timestamp	When the file was closed(floating point value denoting seconds since 1970)	8 bytes	no term

(**) Variable length data types (e.g. STRING) will include their length in bytes at the beginning of the Value element, adding 4 bytes to the element length.

An archive file is composed of 3 components, an archive header, any number of data records, and an archive ender.

An archive header consists of a 4 byte integer indicating the size the subsystem_data.xml following by the ASCII string that is the file contents. The xml file is included with each archive file so the data definition is always with the data that has been archived.

The data record part consists of 1 to many individual records. Each data record has two components, a data record header and the data component.

The first element of the header is a 4 byte long Sync Word element to allow the retrieval software to remain in synch and distinguish the breaks between records. After that is a Record Size element, which is the size of the entire record in bytes, including all the Header component elements. The Record Size element itself has a fixed length of 4 bytes. Next is the timestamp, which denotes when the data group was written to the archive file. The data component will also have the MCCS timestamp for the data group as one (usually the first) of the data elements. Finally, there is the address or data name, which completely specifies the path through the data tree to that data group, and uses the same syntax provided to the user by the SOFIA Command Language (SCL) to fetch that same data (get/subscribe) from the MCCS.

The data component is the list of new data values for that data group serialized in the exact same way as is done to distribute the data group around the MCCS. The retrieval software will need to use the XML definition for the data group to know how to de-serialize and unpack the individual data elements based on their ordering, type, and size. Elements of a data group will be written to the archive in the exact same order as they are specified in the XML definition file. The type and size of a given data element can be obtained from the "rep" attribute in its XML definition. If a data element is of a variable length type (STRING, BINARY), its first four bytes will be the data element length in bytes, followed by the value of the element. Note that if the size of a variable length type is 0, then there will be no value for that data element. The length of the fixed length types (UINT2, INT4, FLOAT8, etc.) can be determined from the value of the "rep" attribute, so the length of the data element is not included in the archive file. The end of the data record is not explicitly marked or terminated, but can be identified from the data record size in the record header and confirmed by the sync word for the next record which immediately follows the data record.

Coincident with its distribution of a data update to the rest of the MCCS, any given MCCS data source handler, proxy in most cases, will do a single write of one record to the archive file for that data source with the record format appropriate for that data group. It is important to understand that even though the data in an individual record "obeys" the sequence of data items given in the XML data group definition, the records themselves can come (and should be assumed will come) in any order. A record is added to the archive when it arrives at the MCCS from a given data source, the sequence of which will depend on the data rates of all the data groups for that source or on how the given MCCS proxy retrieves that data relative to the other data from the source.

The archive ender component is technically an empty data record ending with a timestamp and no data. This is the time when the archive file is closed.

APPENDIX F.2 ARCHIVE INDEX FILE

1. Overview

The MCCS Archiver maintains an index of all files stored on its removable drive. This file is used by SOFIA staff to assure that all expected data products were acquired and to confirm the integrity of the file contents as it moves from the aircraft into permanent archives at the Science Support Center (SSC).

2. File Naming Convention

File name is archiver_index.csv and it resides in the highest-level directory of the removable drive.

3. File Content

Comment lines (those with '#' as the first non-whitespace character) and blank lines (empty lines or those with only whitespace characters) may occur anywhere in the file and should be ignored by programs reading this file since they have no pre-defined content. All other lines are comma-separated-value lines which are stored alphabetically by the FilenameWithPath field.

The format of each comma-separated-value line is:

<FilenameWithPath>,<SHA-1 checksum>,<Date>,<Time>,<FileSize>,<FileState>

Where:

<FilenameWithPath>	*relative path and filename of file (path starts at highest level directory of Archiver removable drive)*
<SHA-1 checksum>	*40 hexadecimal character SHA-1 checksum of the file*
<Date>	*date (in YYYY-MM-DD format) of last modification of file*
<Time>	*time (in HH:MM:SS.sss format) of last modification of file*
<FileSize>	*number of bytes in file*
<FileState>	*state of this file:*
OPEN	*lsof system call determined that the file is open for writing*
PENDING	*lsof system call determined the file is closed but it has not been closed for one update cycle*
PENDINGCKSUM	*file is closed and has been closed for one cycle and time, date and size are unchanged*
FINAL	*file is not open and the runonce option is set (this indicates that the information represents the final post-flight state)*
DELETED	*file was on the system for one run but not on subsequent runs*

For example:

```
#       This is a manifest of the archive data sorted alphabetically by FilenameWithPath
#
#       Date: 2013-11-01
#       Time: 15:39:54.558
#       Process Time elapsed: 9.501172327 seconds
#               Number of Directories:        36
#               Number of Files:              3548
#                       Number Open:          3
#                       Number Pending:       14
#                               Wait Cycle:   0
#                               Time Changed: 5
#                               Size Changed: 9
#                       Number Final:         3531
#                       Number Deleted:       1

ark/archiver_proxy.ars.131031164219.ark,87294825657a962ae610040a4a8f2235,2013-10-31,17:17:11.746,538078,FINAL
ark/cdds_proxy.cdds.131031164223.ark,8dd99b1786ff8fb21537f26da7110d6a,2013-10-31,16:42:27.601,66966,FINAL
ark/das_proxy.das.131031164220.ark,81cd4f7ae393bd6a171391b9dc3f06f0,2013-10-31,17:17:13.031,20185088,FINAL
ark/fpi_node_b.ta_fpi.131031201253.ark,9f78576bf7727442fd31612a4e66c143,2013-10-31,22:29:13.122,73639,FINAL
ark/ntp_proxy.ntp.131031164217.ark,0106512dd161099ff9dd22098108a17c,2013-10-31,16:47:09.023,10057,FINAL
ark/ta_proxy.ta.131031164217.ark,efca7ea15e9e892dc7c8ca25f4b1df4e,2013-10-31,17:17:05.504,32309,FINAL
ark/ta_proxy.ta_mcp.131031164244.ark,34772a36d31d8f8b66e8e6168154711d,2013-10-31,17:14:00.040,1861824,FINAL
ark/ta_proxy.ta_trc.131031223123.ark,3458ac0cb0099f166490daf3325703cd,2013-10-31,23:59:28.589,50000026,FINAL
ark/ta_proxy.ta_tsc.131031223119.ark,7bff27c4ea4fce2aba4a199e584d38f4,2013-11-01,13:17:22.908,42186179,FINAL
ark/wfi_node_b.ta_wfi.131031164243.ark,aa4eb58e2dfd6e002cc8e38d25ed1f09,2013-10-31,17:17:03.348,30329,FINAL
ark/xform_proxy.coord.131031164247.ark,fa0ed9ba921443999969fa63087e6ba7,2013-10-31,17:17:12.938,365517,FINAL
ark/xform_proxy.dither.131031164243.ark,d879aa75261f144d69fc1f2ffe28c604,2013-10-31,17:14:00.151,26656,FINAL
ark/xform_proxy.great.131031224336.ark,998733b87d7f7d306bd515211d9adce1,2013-11-01,10:57:07.355,278692,FINAL
ark/xform_proxy.nod.131031164256.ark,51449a7b7ad7046ba3fe5757deb95ca5,2013-10-31,17:14:00.470,40121,FINAL
ark/xform_proxy.sma.131031164245.ark,4c509f05b4e2c5bd8c1261b886aa2b88,2013-10-31,17:13:57.726,86523,FINAL
ark/xform_proxy.ta_pos.131031164253.ark,428456814b37813effa06f1585835687,2013-10-31,17:13:57.492,60922,FINAL
ark/xform_proxy.ta_state.131031164242.ark,4bc6c8bac9e80ee109aae0c0683c0477,2013-10-31,17:17:12.089,2818048,FINAL
das/log/log.txt,2810afd3bd8ca458336a53cc5293eedf,2013-11-01,13:19:16.911,2091313,DELETED
fmi/mission_data/fmi_log/fmi_backend_log.20130826103822,300f60ee0e4f14c255d8ee8234646f9b,2013-08-26,16:41:32.966,842,FINAL
fmi/mission_data/fp/data/F136/31Oct12hr-SOFIA_gridV21.xlsx,8e6cbaea6beb1f6975b2b2bab025f3fe,2013-10-31,18:17:52.421,89619,FINAL
mission_data/SOFIA_Manifest_Flt_136.pdf,e52988f0d3067e4be6ec4e0e530d5346,2013-10-31,15:00:33.499,80277,FINAL
mission_data/cmdlog_20131031T200316Z.txt,f28976956854954b4a51af08a4e3ac15,2013-10-31,22:29:10.617,2677732,FINAL
mission_data/ephemeris/JUNO.TXT,6c7cf048294bed99918984756bdc6ed2,2013-08-28,05:12:51.157,511473,OPEN
mission_data/gui_records/tester_to.20131031.203439.log,0af6c3d6c0be1434cc847b53284859c4,2013-10-31,21:29:22.173,29429388,FINAL
mission_data/imager_data/co-ws.fpi_image.1.fits,e57d0b373347cce3bed4a9f65c600fe3,2013-11-01,00:58:53.906,2100032,FINAL
mission_data/losrewind_error.png,3151576bab90da07489557c0a16a8158,2013-10-01,11:02:27.095,22632,FINAL
si/GREAT/cycle1_14/009338_002_GREAT_CPBE0.fits,ebd90e6e341c6d44be7746167d13f03b,2013-11-01,07:23:50.556,23040,FINAL
si/GREAT/cycle1_14/009338_002_GREAT_SAFFTS0.fits,eb62b29a11f0d8c8a71f35d661375b40,2013-11-01,07:23:52.699,51840,FINAL
si/GREAT/cycle1_14/otf_chopped_cross.log,e2632e0586d06670d30f46677f7ffe0e,2013-11-01,14:03:14.021,88,FINAL
```

APPENDIX F.3 EPHEMERIS FILE

The ephemeris file contains time ordered rows of Equatorial Reference Frame (ERF) coordinates, distance, and other observing information in a comma-separated-value (CSV) format that can be output by JPL Horizons (see http://ssd.jpl.nasa.gov/horizons.cgi#top).

The MCCS has a fixed amount of memory allocated for use by ephemeris files and will provide an error response to the user of the coord.position command when this memory has been used up. The size of the ephemeris file is dependent on its duration and step size; but a limit of 14,400 ephemeris rows per ephemeris file governs the maximum size of any one file. MCCS allows up to 10 ephemeris files to be actively defined and allows 10 positions to be associated with each one. Refer to the "ephemeris" keyword of the "coord.position" command to define ephemeris objects. See the "coord.aoi_create" command to assign ephemeris to AOIs. To inspect whether AOIs or the TA fine drive are associated with ephemeris, subscribe for and inspect the "coord.non_sidereal.AOI{1-8}.ephemeris" and "cood.non_sidereal.target.ephemeris" housekeeping data.

Note that if using JPL Horizons, care must be taken to produce geocentric ephemeris files. JPL Horizons can create topocentric ephemeris files that are correct for a given longitude, latitude, and altitude; however, MCCS assumes that the ephemeris files are geocentric and will perform the calculations necessary to compute topocentric coordinates for the current conditions. If using JPL Horizons to generate the ephemeris files, the "Center-site name" field prior to the start-of-ephemeris ($$SOE) field will state "GEOCENTRIC".

NOTE: the MCCS enforces the presence of "GEOCENTRIC" prior to the ephemeris table.

The following table identifies the non-sidereal ephemeris input that is supported:

Item Name	Meaning	Notes/Format
Site Name	Site name needs to be GEOCENTRIC because the MCCS converts the geocentric ephemeris to the topocentric coordinates using SOFIA's present longitude, latitude and altitude.	Center-site name: GEOCENTRIC
******…****	Asterisks	MCCS expects at least 6 asterisks before the ephemeris header
Ephemeris Table Header	Identifies the column names present in the ephemeris table in CSV format; e.g. Date__(UT)__HR:MN:SC.fff, , , R.A._(ICRF/J2000.0), DEC_(ICRF/J2000.0), APmag, S-brt, Ang-diam, delta, TDB-UT,	Precedes the Start of Ephemeris **NOTE**: the fields for APmag, S-brt are variable with the following possibilities: **APmag, S-brt** (Non-comet with known dimensions)

		APmag (Non-comet with unknown dimensions) **T-mag, N-mag** (comets; total & nuclear magnitudes) This line is required. It defines which variable field format is being used in the file. **NOTE**: the TDB-UT field was formerly CT-UT; either is supported. The only other required lines are the $$SOE and $$EOE lines, and the comma-separated-value (CSV) ephemeris lines between them. All other lines (before or after the $$SOE and $$EOE lines) are ignored.
******…****	Asterisks	MCCS expects at least 6 asterisks after the ephemeris header
Start	Start of Ephemeris tag – separates the ephemeris comma-separated-value (CSV) table within the file from anything else in the file. JPL Horizon's allows CSV as an option. SOFIA requires the comma for ease of processing because some fields values are empty.	$$SOE
Date	UTC (of user specified precision which can be HH:MN, HH:MN:SS or HH:MN:SC.SSS). JPL HORIZONS may support additional precision in the future.	"2013-Mar 25 00:00" or "2013-May-12 00:00:00.000" In table that follows $$SOE. Step size is chosen by the observer to produce the required accuracy. First column in file; could be preceded by a blank. See JPL HORIZONS observer-table settings to define time digit precision.
SOLAR PRESENCE (OBSERVING SITE)	This field is unused by MCCS and should be empty, but it is retained for compatibility with JPL HORIZONS, where this field is: '*' Daylight (refracted solar upper-limb on or above apparent horizon) 'C' Civil twilight/dawn 'N' Nautical twilight/dawn 'A' Astronomical twilight/dawn ' ' Night OR geocentric ephemeris	In table that follows $$SOE SOLAR PRESENCE follows the date field; but note that it should be empty for a geocentric ephemeris for SOFIA.
LUNAR PRESENCE	This field is unused by MCCS and should be empty, but it is retained for compatibility with JPL HORIZONS, where this field is: LUNAR PRESENCE WITH TARGET RISE/TRANSIT/SET MARKER (OBSERVING SITE) The solar-presence symbol is immediately followed by another marker symbol:	In table that follows $$SOE LUNAR PRESENCE follows the SOLAR PRESENCE field; but note that it should be empty for a geocentric ephemeris for SOFIA.

	'm' Refracted upper-limb of Moon on or above apparent horizon ' ' Refracted upper-limb of Moon below apparent horizon OR geocentric 'r' Rise (target body on or above cut-off RTS elevation) 't' Transit (target body at or past local maximum RTS elevation) 's' Set (target body on or below cut-off RTS elevation)	
R.A.	J2000.0 astrometric right ascension of target center. Adjusted for light-time. Units: HMS (HH MM SS.ff or HH MM SS.ffff). JPL HORIZONS may support additional precision in the future.	10 20 24.32 In table that follows $$SOE R.A. follows the LUNAR PRESENCE field. See JPL HORIZONS observer-table settings to define extra precision and angle format in HMS.
DEC	J2000.0 astrometric declination of target center. Adjusted for light-time. Units: DMS (DD MM SS.f or DD MM SS.fff) JPL HORIZONS may support additional precision in the future.	+04 46 22.5 In table that follows $$SOE The DEC field follows the R.A. field. See JPL HORIZONS observer-table settings to define extra precision and angle format.
APmag (or T-mag)	Approximate apparent visual magnitude. Note that if insufficient information is available, APmag field can be returned as text symbols "n.a." instead of numerics, meaning "not available". Additionally, for comets, this field changes to T-mag meaning “total magnitude”.	12.51 In table that follows $$SOE The APmag field follows the _DEC field. NOTE: no current use for this field by MCCS is currently planned; this field/property is present in MCCS_SI_04 for compatibility with TASim ephemeris. NOTE: This field can change meaning, see the description on the Ephemeris Table Header above.
SBrt	Surface brightness. Units: VISUAL MAGNITUDES PER SQUARE ARCSECOND. Additionally, for comets, this field changes to N-mag meaning “nuclear magnitude”.	7.66 In table that follows $$SOE Follows APmag The SBrt field follows the APMag field. There is no current SOFIA use planned for SBrt, but it is usually provided by JPL HORIZONS when APmag is selected for output. NOTE: This field can change meaning and/or be absent, see the description on the Ephemeris Table Header above. NOTE: no current use for this field by MCCS is currently

		planned; this field/property is present in MCCS_SI_04 for compatibility with TASim ephemeris.
Ang-diam	The equatorial angular width of the target body full disk, if it were fully visible to the observer. Units: ARCSECONDS Note that if insufficient information is available, Ang-diam field can be returned as text symbols "n.a." instead of numerics, meaning "not available".	0.122253 In table that follows $$SOE The Ang-diam field follows the SBrt field. NOTE: no current use for this field by MCCS is currently planned; this field/property is present in MCCS_SI_04 for compatibility with TASim ephemeris.
Delta	Range of target center with respect to the observer at the instant light seen by the observer at print-time would have left the target center (print-time minus down-leg light-time); Units: AU	3.41166034885096 In table that follows $$SOE The Delta field follows the Ang-diam field.
Delta-T	Delta-T will be present to inform the difference between Coordinate Time (CT – historical) and Universal Time or between Barycentric Dynamical Time (TDB)	CT-UT ----or---- TDB-UT **NOTE**: JPL Horizons documentation and label usage of Coordinate Time (CT) has been changed to Barycentric Dynamical Time (TDB). There is no change in definition, it has always been the same time-scale, this is only a nomenclature change now that the IAU definition of TDB has been fixed. MCCS supports both CT-UT or TDB-UT headings.
Extra fields	Additional fields might be present	For instance delta-dot might be present if the JPL HORIZONS user does not click the option to suppress range-rate for range/range-rate output on the observer table settings.
End	End of Ephemeris tag --- separates the ephemeris portion of file	$$EOE

If using JPL Horizons to generate the ephemeris, table settings of "QUANTITIES=**1,9,13,20,30**; time digits=**FRACSEC**; suppress range-rate=**YES**; extra precision=**YES**; CSV format=**YES"**

Example 1: This just contains only the essential lines of an MCCS ephemeris file – header line, $$SOE line, ephemeris lines, and $$EOE line. All other lines in the file are ignored. The "APmag", "S-brt", and "Ang-diam" fields are not used by MCCS and may be left empty as shown in this example.

```
*******************************************************************************************************************
Date__(UT)__HR:MN:SC.fff, , ,R.A._(ICRF/J2000.0), DEC_(ICRF/J2000.0), APmag, S-brt, Ang-diam, delta, TDB-UT,
*******************************************************************************************************************
$$SOE
```

VERIFY THAT THIS IS THE CORRECT VERSION BEFORE USE

```
 2013-Jun-10 00:00:00.000, , ,05 43 05.4557,+23 08 06.535, , , , 6.11944719953273, 67.184680,
 2013-Jun-10 01:00:00.000, , ,05 43 07.2050,+23 08 07.792, , , , 6.11967646111565, 67.184679,
 2013-Jun-10 02:00:00.000, , ,05 43 08.9805,+23 08 09.091, , , , 6.11992712225892, 67.184678,
 2013-Jun-10 03:00:00.000, , ,05 43 10.7860,+23 08 10.431, , , , 6.12019831194085, 67.184677,
 2013-Jun-10 04:00:00.000, , ,05 43 12.6250,+23 08 11.813, , , , 6.12048905054673, 67.184676,
 2013-Jun-10 05:00:00.000, , ,05 43 14.5011,+23 08 13.234, , , , 6.12079825597399, 67.184675,
 2013-Jun-10 06:00:00.000, , ,05 43 16.4173,+23 08 14.694, , , , 6.12112475017587, 67.184674,
 2013-Jun-10 07:00:00.000, , ,05 43 18.3768,+23 08 16.192, , , , 6.12146726609478, 67.184673,
 2013-Jun-10 08:00:00.000, , ,05 43 20.3822,+23 08 17.725, , , , 6.12182445493721, 67.184672,
 2013-Jun-10 09:00:00.000, , ,05 43 22.4360,+23 08 19.293, , , , 6.12219489374340, 67.184671,
 2013-Jun-10 10:00:00.000, , ,05 43 24.5406,+23 08 20.894, , , , 6.12257709320632, 67.184670,
$$EOE
```

Example 2: The following CSV file was created for Europa, a JOVIAN satellite via JPL Horizons with the settings "QUANTITIES=**1,9,13,20, 30**; time digits=**FRACSEC**; suppress range-rate=**YES**; extra precision=**YES**; CSV format=**YES**".
Note that although this example contains many more lines than Example 1 above, all the additional lines and fields in the ephemeris entry are ignored by MCCS, so the two examples should work identically in MCCS.

```
*******************************************************************************
 Revised: Sep 12, 2013           Europa / (Jupiter)                        502

 SATELLITE PHYSICAL PROPERTIES:
  Radius (km)             = 1565   +- 8     Density (g cm^-3)   =  2.99 +- 0.05
  Mass (10^20 kg )        =  479.7 +- 1.5   Geometric Albedo    =  0.6

 SATELLITE ORBITAL DATA:
  Semi-major axis, a (km) = 671.079 (10^3)  Orbital period      = 3.551810 d
  Eccentricity, e         = 0.0101          Rotational period   = Synchronous
  Inclination, i  (deg)   = 0.464
*******************************************************************************




*******************************************************************************
Ephemeris / WWW_USER Tue Nov 26 10:17:15 2013 Pasadena, USA      / Horizons
*******************************************************************************
Target body name: Europa (502)                   {source: JUP309}
Center body name: Earth (399)                    {source: DE-0431LE-0431}
Center-site name: GEOCENTRIC
*******************************************************************************
Start time      : A.D. 2013-Jun-10 00:00:00.0000 UT
Stop  time      : A.D. 2013-Jun-10 10:00:00.0000 UT
Step-size       : 60 minutes
*******************************************************************************
Target pole/equ : IAU_EUROPA                     {East-longitude -}
```

```
Target radii    : 1562.6 x 1560.3 x 1559.5 km     {Equator, meridian, pole}
Center geodetic : 0.00000000,0.00000000,0.0000000 {E-lon(deg),Lat(deg),Alt(km)}
Center cylindric: 0.00000000,0.00000000,0.0000000 {E-lon(deg),Dxy(km),Dz(km)}
Center pole/equ : High-precision EOP model        {East-longitude +}
Center radii    : 6378.1 x 6378.1 x 6356.8 km     {Equator, meridian, pole}
Target primary  : Jupiter
Vis. interferer : MOON (R_eq= 1737.400) km        {source: DE-0431LE-0431}
Rel. light bend : Sun, EARTH                      {source: DE-0431LE-0431}
Rel. lght bnd GM: 1.3271E+11, 3.9860E+05 km^3/s^2
Atmos refraction: NO (AIRLESS)
RA format       : HMS
Time format     : CAL
EOP file        : eop.131125.p140216
EOP coverage    : DATA-BASED 1962-JAN-20 TO 2013-NOV-25. PREDICTS-> 2014-FEB-15
Units conversion: 1 au= 149597870.700 km, c= 299792.458 km/s, 1 day= 86400.0 s
Table cut-offs 1: Elevation (-90.0deg=NO ),Airmass (>38.000=NO), Daylight (NO )
Table cut-offs 2: Solar Elongation (  0.0,180.0=NO ),Local Hour Angle( 0.0=NO )
Table format    : Comma Separated Values (spreadsheet)
*******************************************************************************************************************
 Date__(UT)__HR:MN:SC.fff, , ,R.A._(ICRF/J2000.0), DEC_(ICRF/J2000.0),  APmag, S-brt, Ang-diam,            delta,        TDB-UT,
*******************************************************************************************************************
$$SOE
 2013-Jun-10 00:00:00.000, , ,05 43 05.4557,+23 08 06.535,   6.15,  5.13, 0.704150, 6.11944719953273,    67.184680,
 2013-Jun-10 01:00:00.000, , ,05 43 07.2050,+23 08 07.792,   6.15,  5.13, 0.704124, 6.11967646111565,    67.184679,
 2013-Jun-10 02:00:00.000, , ,05 43 08.9805,+23 08 09.091,   6.15,  5.13, 0.704095, 6.11992712225892,    67.184678,
 2013-Jun-10 03:00:00.000, , ,05 43 10.7860,+23 08 10.431,   6.15,  5.13, 0.704064, 6.12019831194085,    67.184677,
 2013-Jun-10 04:00:00.000, , ,05 43 12.6250,+23 08 11.813,   6.15,  5.13, 0.704030, 6.12048905054673,    67.184676,
 2013-Jun-10 05:00:00.000, , ,05 43 14.5011,+23 08 13.234,   6.15,  5.13, 0.703995, 6.12079825597399,    67.184675,
 2013-Jun-10 06:00:00.000, , ,05 43 16.4173,+23 08 14.694,   6.15,  5.13, 0.703957, 6.12112475017587,    67.184674,
 2013-Jun-10 07:00:00.000, , ,05 43 18.3768,+23 08 16.192,   6.15,  5.13, 0.703918, 6.12146726609478,    67.184673,
 2013-Jun-10 08:00:00.000, , ,05 43 20.3822,+23 08 17.725,   6.15,  5.13, 0.703877, 6.12182445493721,    67.184672,
 2013-Jun-10 09:00:00.000, , ,05 43 22.4360,+23 08 19.293,   6.15,  5.13, 0.703834, 6.12219489374340,    67.184671,
 2013-Jun-10 10:00:00.000, , ,05 43 24.5406,+23 08 20.894,   6.15,  5.13, 0.703790, 6.12257709320632,    67.184670,
$$EOE
*******************************************************************************************************************
Column meaning:

TIME

  Prior to 1962, times are UT1. Dates thereafter are UTC. Any 'b' symbol in
the 1st-column denotes a B.C. date. First-column blank (" ") denotes an A.D.
date. Calendar dates prior to 1582-Oct-15 are in the Julian calendar system.
Later calendar dates are in the Gregorian system.

  Time tags refer to the same instant throughout the universe, regardless of
where the observer is located.

  The dynamical Coordinate Time scale is used internally. It is equivalent to
the current IAU definition of "TDB". Conversion between CT and the selected
non-uniform UT output scale has not been determined for UTC times after the
```

```
next July or January 1st.  The last known leap-second is used over any future
interval.

  NOTE: "n.a." in output means quantity "not available" at the print-time.

 R.A._____(ICRF/J2000.0)_____DEC =
    J2000.0 astrometric right ascension and declination of target center.
Adjusted for light-time. Units: HMS (HH MM SS.ffff) and DMS (DD MM SS.fff)

 APmag S-brt =
    Target's approximate apparent visual magnitude & surface brightness. For
planets and satellites, values are available only for solar phase angles in the
range generally visible from Earth. This is to avoid extrapolation of models
beyond their valid (data-based) limits.  Units: NONE & VISUAL MAGNITUDES PER
SQUARE ARCSECOND

 Ang-diam =
   The equatorial angular width of the target body full disk, if it were
fully visible to the observer.  Units: ARCSECONDS

 delta  =
   Range ("delta") of the target center relative to the observer at the instant
light seen by the observer at print-time would have left the target center
(print-time minus down-leg light-time); the distance traveled by a light ray
emanating from the center of the target and recorded by the observer at
print-time. Units: AU

 TDB-UT =
   Difference between uniform Coordinate Time scale and Earth-rotation
dependent Universal Time. Prior to 1962, the difference is with respect
to UT1 (CT-UT1).  For 1962 and later, the delta is with respect to UTC
(CT-UTC).  Values beyond the next July or January 1st may change if a
leap-second is introduced. Units: SECONDS


 Computations by ...
     Solar System Dynamics Group, Horizons On-Line Ephemeris System
     4800 Oak Grove Drive, Jet Propulsion Laboratory
     Pasadena, CA  91109   USA
     Information: http://ssd.jpl.nasa.gov/
     Connect    : telnet://ssd.jpl.nasa.gov:6775  (via browser)
                  telnet ssd.jpl.nasa.gov 6775    (via command-line)
     Author     : Jon.Giorgini@jpl.nasa.gov

*******************************************************************************************************************************
```

APPENDIX G. ALERT AND ALARM MESSAGES

The available alert and alarm messages are shown, respectively, in Tables G-1 and G-2 below; see Section 4.3 for details on the usage of alerts and alarms. For each is shown the associated subsystem, index/identification number, type (INFO, WARNING, ERROR, or FATAL per Table 19, Alert/Alarm Severity Levels), *audio sound and associated MADS channel (if an audible one),* and associated message.

Table G-1: Alert Messages.

Subsystem	Index (ID=)	Type (I/W/E/F)	*Sound & Channel*	Message and/or Formatted String
All Systems *(Alerts based on housekeeping parameter values)*				
all	***parameter_name***	WARNING		The value ***full_housekeeping_parameter_name*** (***current_value***) is below the warning limit at ***low_warning_limit***
all	***parameter_name***	WARNING		The value ***full_housekeeping_parameter_name*** (***current_value***) is above the warning limit at ***high_warning_limit***
all	***parameter_name***	ERROR		The value ***full_housekeeping_parameter_name*** (***current_value***) is below the error limit at ***low_limit***
all	***parameter_name***	ERROR		The value ***full_housekeeping_parameter_name*** (***current_value***) is above the error limit at ***high_limit***
all	***parameter_name***	ERROR		The value ***full_housekeeping_parameter_name*** is +inf
all	***parameter_name***	ERROR		The value ***full_housekeeping_parameter_name*** is -inf
all	***parameter_name***	ERROR		The value ***full_housekeeping_parameter_name*** is Not A Number
CDDS				
cdds	CDDS0x50_A	INFO		AC power-up.
cdds	CDDS0x51_A	INFO		DC power-up.
cdds	CDDS0x9D_B	INFO		CDCS initializing in flight mode.
cdds	CDDS0x9D_C	INFO		CDCS initializing in shop mode.
cdds	CDDS0x9D_D	INFO		CDCS initializing in dataloader mode.
cdds	CDDS0x9D_E	INFO		CDCS initializing in qual flight mode.
cdds	CDDS0x50_B	WARNING		AC undervoltage, less than 50 ms.
cdds	CDDS0x51_B	WARNING		DC undervoltage, less than 50 ms.
cdds	CDDS0x1E_A	WARNING		Invalid checksum in setpoint packet.
cdds	CDDS0x1E_B	WARNING		Parity error in setpoint packet.
cdds	CDDS0x1E_C	WARNING		One or more setpoints late/missed.
cdds	CDDS0x1E_D	WARNING		Invalid message ID in setpoint packet.
cdds	CDDS0x4F_B	WARNING		URD seal source pressure out of range.
cdds	CDDS0x4F_D	WARNING		URD seal regulated pressure out of range.
cdds	CDDS0x4F_H	WARNING		URD seal inflation time out of range.
cdds	CDDS0x4F_J	WARNING		URD seal inflation pressure out of range.
cdds	CDDS0x0	ERROR		***CDDS Message***
cdds	CDDS0x17	ERROR		Gap created between URD and LFD!
cdds	CDDS0x1F	ERROR		Requested setpoint is out of range.
cdds	CDDS0x47	ERROR		URD has exceeded torque limit with torque:
cdds	CDDS0x48	ERROR		URD has exceeded speed limit with speed:
cdds	CDDS0x49	ERROR		LFD has exceeded torque limit with torque:
cdds	CDDS0x4A	ERROR		LFD has exceeded speed limit with speed:
cdds	CDDS0x4B_A	ERROR		URD has reached end of travel (closed limit)!
cdds	CDDS0x4B_B	ERROR		URD has reached end of travel (open limit)!
cdds	CDDS0x4C_A	ERROR		LFD has reached end of travel (closed limit)!
cdds	CDDS0x4C_B	ERROR		LFD has reached end of travel (open limit)!
cdds	CDDS0x4D_A	ERROR		Self-test processor instructions fault.
cdds	CDDS0x4D_B	ERROR		RAM fault.
cdds	CDDS0x4D_C	ERROR		Non-volatile memory fault.
cdds	CDDS0x4D_D	ERROR		External watchdog fault.
cdds	CDDS0x4D_E	ERROR		ROM fault.
cdds	CDDS0x4D_F	ERROR		Previous system fault.
cdds	CDDS0x4D_G	ERROR		Adjustable parameters fault.

Subsystem	Index (ID=)	Type (I/W/E/F)	*Sound & Channel*	Message and/or Formatted String
cdds	CDDS0x4D_H	ERROR		URD upper end of travel monitor #1 fault.
cdds	CDDS0x4D_I	ERROR		URD upper end of travel monitor #2 fault.
cdds	CDDS0x4D_J	ERROR		URD lower end of travel monitor #1 fault.
cdds	CDDS0x4D_K	ERROR		URD lower end of travel monitor #2 fault.
cdds	CDDS0x4D_L	ERROR		AA/LFD upper end of travel monitor #1 fault.
cdds	CDDS0x4D_M	ERROR		AA/LFD upper end of travel monitor #2 fault.
cdds	CDDS0x4D_N	ERROR		AA/LFD lower end of travel monitor #1 fault.
cdds	CDDS0x4D_O	ERROR		AA/LFD lower end of travel monitor #2 fault.
cdds	CDDS0x4D_P	ERROR		Gap monitor #1 latent failure fault.
cdds	CDDS0x4D_Q	ERROR		Gap monitor #2 latent failure fault.
cdds	CDDS0x4D_R	ERROR		URD upper torque limit overtorque monitor A1 fault.
cdds	CDDS0x4D_S	ERROR		URD lower torque limit overtorque monitor A1 fault.
cdds	CDDS0x4D_T	ERROR		URD upper torque limit overtorque monitor A2 fault.
cdds	CDDS0x4D_U	ERROR		URD lower torque limit overtorque monitor A2 fault.
cdds	CDDS0x4D_V	ERROR		URD upper torque limit overtorque monitor B1 fault.
cdds	CDDS0x4D_W	ERROR		URD lower torque limit overtorque monitor B1 fault.
cdds	CDDS0x4D_X	ERROR		URD upper torque limit overtorque monitor B2 fault.
cdds	CDDS0x4D_Y	ERROR		URD lower torque limit overtorque monitor B2 fault.
cdds	CDDS0x4D_Z	ERROR		AA/LFD upper torque limit overtorque monitor A1 fault.
cdds	CDDS0x4D_AA	ERROR		AA/LFD lower torque limit overtorque monitor A1 fault.
cdds	CDDS0x4D_AB	ERROR		AA/LFD upper torque limit overtorque monitor A2 fault.
cdds	CDDS0x4D_AC	ERROR		AA/LFD lower torque limit overtorque monitor A2 fault.
cdds	CDDS0x4D_AD	ERROR		AA/LFD upper torque limit overtorque monitor B1 fault.
cdds	CDDS0x4D_AE	ERROR		AA/LFD lower torque limit overtorque monitor B1 fault.
cdds	CDDS0x4D_AF	ERROR		AA/LFD upper torque limit overtorque monitor B2 fault.
cdds	CDDS0x4D_AG	ERROR		AA/LFD lower torque limit overtorque monitor B2 fault.
cdds	CDDS0x4D_AH	ERROR		URD positive overspeed monitor #1 fault.
cdds	CDDS0x4D_AI	ERROR		URD negative overspeed monitor #1 fault.
cdds	CDDS0x4D_AJ	ERROR		URD positive overspeed monitor #2 fault.
cdds	CDDS0x4D_AK	ERROR		URD negative overspeed monitor #2 fault.
cdds	CDDS0x4D_AL	ERROR		AA/LFD positive overspeed monitor #1 fault.
cdds	CDDS0x4D_AM	ERROR		AA/LFD negative overspeed monitor #1 fault.
cdds	CDDS0x4D_AN	ERROR		AA/LFD positive overspeed monitor #2 fault.
cdds	CDDS0x4D_AO	ERROR		AA/LFD negative overspeed monitor #2 fault.
cdds	CDDS0x4D_AP	ERROR		Shop mode enabled.
cdds	CDDS0x4D_AQ	ERROR		RS-422 loopback fault.
cdds	CDDS0x4D_AR	ERROR		Torque loopback fault.
cdds	CDDS0x50_C	ERROR		AC undervoltage, between 50 and 200 ms!
cdds	CDDS0x50_D	ERROR		AC undervoltage, more than 200 ms!
cdds	CDDS0x51_C	ERROR		DC undervoltage, between 50 and 200 ms!
cdds	CDDS0x51_D	ERROR		DC undervoltage, more than 200 ms!
cdds	CDDS0x79	ERROR		URD did not open!
cdds	CDDS0x9F	ERROR		Invalid IFD switch transition!
cdds	CDDS0x3C_A	ERROR		LFD forward limit switch has tripped!
cdds	CDDS0x3C_B	ERROR		LFD middle limit switch has tripped!
cdds	CDDS0x3C_C	ERROR		LFD aft limit switch has tripped!
cdds	CDDS0x4E_A	ERROR		Flight executable CRC continuous BIT failed!
cdds	CDDS0x4E_B	ERROR		Shop executable CRC continuous BIT failed!
cdds	CDDS0x4F_A	ERROR		URD seal source pressure out of range.
cdds	CDDS0x4F_C	ERROR		URD seal regulated pressure out of range.
cdds	CDDS0x4F_G	ERROR		URD seal pressure out of range.
cdds	CDDS0x4F_I	ERROR		URD seal inflation pressure out of range.
cdds	CDDS0x78_A	ERROR		URD jammed during initial opening sequence!
cdds	CDDS0x78_B	ERROR		URD jammed during closing sequence!
cdds	CDDS0x96_A	ERROR		URD1 motor controller fault!
cdds	CDDS0x96_B	ERROR		URD2 motor controller fault!
cdds	CDDS0x96_C	ERROR		LFD1 motor controller fault!
cdds	CDDS0x96_D	ERROR		LFD2 motor controller fault!
cdds	CDDS0x97_A	ERROR		URD position is inconsistent!
cdds	CDDS0x97_B	ERROR		LFD position is inconsistent!
cdds	CDDS0x98_A	ERROR		URD uncommanded motion detected!
cdds	CDDS0x98_B	ERROR		LFD uncommanded motion detected!
cdds	CDDS0x99_A	ERROR		URD loss of motion detected!

Subsystem	Index (ID=)	Type (I/W/E/F)	*Sound & Channel*	Message and/or Formatted String
cdds	CDDS0x99_B	ERROR		LFD loss of motion detected!
cdds	CDDS0x9A_A	ERROR		URD motor current mismatch detected!
cdds	CDDS0x9A_B	ERROR		LFD motor current mismatch detected!
cdds	CDDS0x9B_A	ERROR		URD motor speed mismatch detected!
cdds	CDDS0x9B_B	ERROR		LFD motor speed mismatch detected!
cdds	CDDS0x9C_A	ERROR		URD1 brake failure detected!
cdds	CDDS0x9C_B	ERROR		URD2 brake failure detected!
cdds	CDDS0x9C_C	ERROR		LFD1 brake failure detected!
cdds	CDDS0x9C_D	ERROR		LFD2 brake failure detected!
cdds	CDDS0x9D_A	ERROR		Undetermined error during CDCS initialization!
cdds	CDDS0x9E_A	ERROR		IFD console switch indicates indeterminate setting.
cdds	CDDS0x9E_B	ERROR		IFD console switch indicates invalid RESET.
cdds	CDDS0x9E_C	ERROR		IFD console switch indicates invalid CLOSE_PARK.
cdds	CDDS0x9E_D	ERROR		IFD console switch indicates invalid OPEN.
cdds	CDDS0x9E_E	ERROR		IFD console switch indicates invalid TRACK.
DAS				
das	DAS_CONNECT	INFO		MCS connected to DAS
das	DAS_DISCONNECT	ERROR		MCS disconnected from DAS
TA				
ta	TA_CONNECT	INFO		MCS connected to TA
ta	TA_DISCONNECT	ERROR		MCS disconnected from TA
ta	ta_mcp	INFO		Asynchronous message received
ta	ta_sma	INFO		Asynchronous message received
ta	ta_tsc	INFO		Asynchronous message received
ta	ta_trc	INFO		Asynchronous message received
ta	ta_mcp	ERROR		Asynchronous message received
ta	ta_sma	ERROR		Asynchronous message received
ta	ta_tsc	ERROR		Asynchronous message received
ta	ta_trc	ERROR		Asynchronous message received
TA Imagers				
ta_ffi	FFI_CONNECT	INFO		MCS connected to FFI
ta_fpi	FPI_CONNECT	INFO		MCS connected to FPI
ta_wfi	WFI_CONNECT	INFO		MCS connected to WFI
ta_ffi	FFI_DISCONNECT	ERROR		MCS disconnected from FFI
ta_fpi	FPI_DISCONNECT	ERROR		MCS disconnected from FPI
ta_wfi	WFI_DISCONNECT	ERROR		MCS disconnected from WFI
WVM State				
wvm	WVM_IF70	INFO		WVM autonomously changed to state ***state***
wvm	WVM_IF80	INFO		Received an acknowledgement (command_ack= ***ack***) even though a command is not active
wvm	WVM_IF50	INFO		WVM Operational
wvm	WVM_IF30	WARNING		WVM did not respond to command sent to it.
wvm	WVM_IF40	WARNING		WVM did not send housekeeping update when expected
wvm	WVM_ERR***id***	WARNING		***Text From Error Status Array***
wvm	WVM_IF20	WARNING		WVM status = ***not_nominal_state***
wvm	WVM_IF90	WARNING		WVM BIT = ***BIT_String***
wvm	WVM_IF60	WARNING		Could not take WVM to operational state
xforms				
coord	coord00	INFO		Coordinate Transforms Command Handler is starting up
dither	dither00	INFO		Dither Command Handler is starting up
nod	nod00	INFO		Nod Command Handler is starting up
sma	sma00	INFO		Sma Command Handler is starting up
ta_pos	ta_pos00	INFO		TA Position Command Handler is starting up
ta_state	ta_state00	INFO		Ta State Command Handler is starting up
xforms	mode	INFO		Mode changed to ***mode***
ta_state	rewind	WARNING		LOS approaching limit: please send 'ta_pos.rewind'. Thank you!
ta_state	iqc	WARNING		Elevation threshold crossed: please send 'ta_state.iqc'. Thank you!
xforms	midnight	WARNING		It is now ***time*** GMT == ***date time MJD***
xforms	irq	WARNING		Elevation threshold crossed: please send 'ta_state.irq'. Thank you!
xforms	mode	ERROR		Mode changed to ***mode***
xforms	non_data	ERROR		trk_centroid_data notification was not a DataNode!!
xforms	moved	ERROR		ta_trc.trc_centroid_table moved from 0x***node*** to ***node***

Subsystem	Index (ID=)	Type (I/W/E/F)	*Sound & Channel*	Message and/or Formatted String
xforms	centroid	ERROR		Failed updating AOI #***aoi_number***
ta_state	xel_los	FATAL		Cross-elevation and elevation are outside of limits!

Table G-2: Alarm Messages.

Subsystem	Index (ID=)	Type (I/W/E/F)	*Sound & Channel*	Message and/or Formatted String
No Alarms are currently defined in MCCS				

APPENDIX H. RELEASE NOTES

1.1 Introduction

This appendix describes the changes between this version of the ICD and the previous major version (Rev K). There are three sub-sections: significant changes throughout ICD, differences in commands (see Appendix A), and differences in housekeeping (see Appendix B).

1.2 Significant Changes throughout ICD

Several significant changes have been made to this document since the previous revision – the list below summarizes those changes.

- General
  - Updated signature page and revision history.
- Section 2 Applicable Documents
  - Per SOFIA-676 (Remove obsolete reference to cancelled NTP/IRIG doc in MCCS_SI_04) removed APP-DR-ICD-SE03-2024 NTP/IRIG_MCCS ICD document reference.
- Section 3 Interface Description
  - Modified language to be more specific regarding the connection of a computer to the MCCS network.
- Section 3.3.5 SI Data-Definition XML Files
  - Per SOF-2211 / SOFIA-836 updated to correct SIRF figures, references, and example with reference to SOF-2182, which is a software bug that is described rather than fixed.
- Section 3.3.6 Observatory Status and Metric Data
  - Modified language to be more specific for setting the observatory status and metric data housekeeping values.
- Section 3.5 Reference Frames
  - Per SOFIA-762 (Remove obsolete/unimplemented coord commands and related HK) removed references to coord.delta and ta_pos.scan from Table 11.
- Appendix F.3 Ephemeris File
  - Per SOFIA-137/SOF-2519 (Reproducible inconsistent behavior of ephemeris files and Named Positions in rev40/rev39) indicated "GEOCENTRIC" is required
  - Per SOFIA-532 (Update MCCS_SI_04 and ephemeris processing to expect TDB-UT versus CT-UT) indicated both TDB-UT and CT-UT are supported

1.3 Changes to the Commands

The details of the changes to the commands are shown directly in Appendix A using markup: deleted items are shown in ~~red strikeout~~ while new items are shown in blue. A summary of the major changes is as follows:

- Per SOFIA-662 (Typo in MCCS_SI_04 [session.] comment description) corrected the text to match the XML.
- Greyed/italicized the save_as argument of the coord.convert command (this argument is not currently supported).
- Greyed/italicized the images argument of the nod.correct command (this argument is not currently supported).
- Per SOFIA-1099/SOF-4548 (sma.position command always returns with an error) greyed out the sma.position command.
- Greyed/italicized the scan mode type of the ta_pos.tweak_define command (this mode is not currently supported).
- Per SOFIA-732 (Utilize TASCU-aware tracker for tweaks) added text to indicate that the offset mode of the ta_pos.tweak command is deprecated.

1.4 Changes to the Housekeeping

The details of changes to housekeeping are shown directly in Appendix B.4 using markup. Deleted items are shown in ~~red strikeout~~ while new items are shown in blue; in addition, housekeeping items which may be added in the future are highlighted with a light green background. If users have an upcoming need for any such highlighted items, they should advise observatory staff so that their addition may be worked into the observatory schedule. A summary of the major changes is as follows:

- Per SOFIA-675 (Eliminate obsolete HK from MCCS_SI_04) removed coord.archive_nsamples.drift_axis_nsample, coord.archive_nsamples.drift_rate_nsample, ta_pos.responses.scan, and ta_pos.responses.stow.
- Corrected an ICD error in the ta_pos.time_until_rewind_ideal representation from a FLOAT4 to a FLOAT8 to match the XML.
- Per SOFIA-343/SOF-3234 (Remove support for unused HK: ta_state.vpa.rate and ta_state.vpa.stdev) removed ta_state.archive_nsamples.vpa_nsample.