Mars 2020 Rover Mission (M2020)

Software Interface Specification

Interface Title: **SuperCam Instrument Experiment Data Record (EDR) and Reduced Data Records (CDR) Data Products for Non-Imaging Components**

Mission: M2020 Date: June 28, 2022

Module ID: D-99966

Module Type (REFerence Only or MISsion-specific info included): MIS

Reference Module ID: N/A Date: N/A

Signatures

GDS Generating Elements:

Instrument Data System (IDS)
Stirling Algermissen

_____
Subsystem Lead Date

Science Receiving Elements:

M2020 Deputy Project Scientist
Ken Williford

_____
Deputy Project Scientist Date

GDS Receiving Elements:

M2020 GDS System Engineer
Maher Hanna

_____
System Engineer Date

M2020 GDS System Engineer
Guy Pyrzak

_____
Subsystem Lead Date

Concurrance:

M2020 SuperCam Instrument Lead
Roger Wiens

_____
Instrument Lead Date

M2020 SuperCam Instrument Scientist
Dorothea Marcia Delapp

_____
Instrument Lead Date

PDS Program Manager
Tim Mclanahan

_____
Manager Date

PDS Geosciences Node Manager
Ray Arvidson

_____
Manager Date

PDS Discipline Node Manager
Lisa Gaddis

_____
Manager Date

Mars 2020 Rover Mission Project

Software Interface Specification (SIS)

SuperCam Experiment Data Record (EDR) and Reduced Data Records (CDR) Data Products for Non-Imaging Components

Version 2.0

Custodians:
Deborah Padgett, Stirling Algermissen

JPL D-99966

June 28, 2022

Jet Propulsion Laboratory
California Institute of Technology

The research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration © 2021. California Institute of Technology. Government sponsorship acknowledged.

CHANGE LOG

DATE	SECTIONS CHANGED	REASON FOR CHANGE	REVISION
2017-01-09	Template created	Initial	Version 0.0
2019-03-08	All sections started	First draft	Version 0.2
2019-08-30	All Sections	Post-ATLO	Version 0.3
2019-09-08	Filename revision		Version 0.4
2019-09-30	Filename time fields	Time formats	Version 0.5
2019-10-10	More filename time fields	Adding use cases	Version 0.6
2019-10-24	Appendix B	Changes to Metadata Keywords	Version 0.7
2020-02-02	Revision for G6.4	New filenames	Version 0.8
2020-03-09	Revision for G6.4.1	New filename scheme	Version 0.9
2020-04-30	Revision G6.5	Metadata,	Version 0.91
2020-05-08	Revision G6.5	Metadata, comments from Dot, new instrument descriptions and figures	Version 0.91
2020-05-14	Revision G6.5	Table 4 PDS, metadata	Version 0.92
2020-06-11	Revision G6.5	Point number in filename; metadata	Version 0.93
2020-06-30	Revision G6.5	Extraneous keywords	Version 0.94
2020-08-14	Revision G6.6	Keywords	Version 0.95
2020-09-10	Revision G6.6	Keywords, Cver	Version 0.97
2020-09-22	Revision for G6.6	Document number	Version 0.98
2020-09-24	PDS Peer Review Revision 1	Export language and s	Version 0.99 rev09-24-20

DATE	SECTIONS CHANGED	REASON FOR CHANGE	REVISION
2020-12-7	PDS Peer Review response 1	Peer review response	V0.99 rev 12-7-2020
2020-12-22	PDS Peer Review Response 2	Peer review response	Version 0.99 rev 12-22-2020
2021-01-05	PDS Peer Review Response 2 + Roger Wiens input	PI response	0.99 rev 1-5-21
2021-01-28	Updated RDR information for PDS Peer Review	RDR	V0.99 rev 1-28-21
2021-03-08	Added reference to User guide and Data Processing documents		V0.99 rev 3-8-21
2021-06-27	PDS Peer Review Calibrated Data response	Peer review response	V0.99 new 6-27-21
2021-07-09	PDS Peer review final liens	PDS	V1.0
2021-07-27	Appendix	Replaced Appendix with PDOG table file reference	V1.0 rev072721
2021-09-15	Updated CDR and DDR sections	PDS	V1.0 rev091521
2021-12-30	PDS Release 2 updates	PDS	V1.1
2022-07-22	PDS Release 3 updates, addition of Raman Derived Data product, and updated for VIS reflectance	PDS	V2.0

OPEN ISSUE ITEMS

REVISION	OPEN ISSUE	CLOSED
v0.99	Appendix C, mapping of ODL label keywords to PDS4 label attributes, is incomplete	
v0.99	Sections 5.5, Derived RDR General Processing, and 5.6, Derived RDR Product Types, will be completed before SuperCam RDR PDS Peer Review Part 2 in September 2021	

Table of Contents

INTERFACE TITLE: SUPERCAM INSTRUMENT EXPERIMENT DATA RECORD (EDR) AND REDUCED DATA RECORDS (CDR) DATA PRODUCTS FOR NON-IMAGING COMPONENTS 1

MISSION: M2020 DATE: DECEMBER 30, 2021 1

SOFTWARE INTERFACE SPECIFICATION (SIS) 2

SUPERCAM 2

EXPERIMENT DATA RECORD (EDR) AND 2

REDUCED DATA RECORDS (CDR) 2

DATA PRODUCTS FOR NON-IMAGING COMPONENTS 2

VERSION 2.0 2

CHANGE LOG 3

OPEN ISSUE ITEMS 5

1 INTRODUCTION 15

1.1 PURPOSE AND SCOPE 15
1.2 CONTENTS 16
1.3 CONSTRAINTS AND APPLICABLE DOCUMENTS 16
1.4 RELATIONSHIPS WITH OTHER INTERFACES 17

2 INSTRUMENT OVERVIEW 18

2.1 INVESTIGATION TECHNIQUES 24
2.1.1 LIBS ACTIVE SPECTRAL DATA COLLECTION: 24
2.1.2 RAMAN ACTIVE SPECTRAL DATA COLLECTION: 25
2.1.3 TIME-RESOLVED LUMINESCENCE DATA COLLECTION: 25
2.1.4 VISIR SPECTRAL DATA COLLECTION: 26
2.1.5 REMOTE MICROSCOPIC IMAGER (RMI) DATA COLLECTION: 26
2.1.6 MICROPHONE DATA COLLECTION: 26
2.2 MODES OF OPERATION: 28
2.3 DATA FORMATS 30

2.4 CRUISE AND SURFACE OPERATIONS 32
2.4.1 DATA PRODUCTS FOR CRUISE 32
2.4.2 TYPICAL SUPERCAM RAMAN SPECTROSCOPY SURFACE USAGE 35
2.4.3 TYPICAL SUPERCAM VISIR SPECTROSCOPY SURFACE USAGE 36

3 GENERAL DATA PRODUCT OVERVIEW 38

3.1 DATA PROCESSING LEVELS 38
3.2 PRODUCT LABEL AND HEADER DESCRIPTIONS 39
3.2.1 OVERVIEW OF LABELS 39
3.2.2 ODL LABELS 39
3.2.3 PDS4 LABEL 42
3.2.4 SUPERCAM FITS FILES 42
3.3 BINARY DATA STORAGE CONVENTIONS 42
3.3.1 BIT AND BYTE ORDERING 43

4 EDR PRODUCT SPECIFICATION 43

4.1 EDR GENERAL PROCESSING 44
4.2 EDR PRODUCT TYPES 45
4.2.1 SCIENCE (SPECTROSCOPY/MICROPHONE/AUTOFOCUS) EDRS 52
4.2.2 ENGINEERING EDRS 53
4.3 EDR PRODUCT FORMAT 55
4.3.1 CRUISE 55
4.4 EDR PRODUCT STRUCTURE 56
4.4.1 FITS 56
4.4.2 SUPERCAM EDR FITS FILES (G6.4+) 56
4.4.3 CRUISE EDR STRUCTURE 59
4.5 EDR PRODUCT VALIDATION 114

5 RDR PRODUCT SPECIFICATION 115

5.1 CALIBRATED RDR GENERAL PROCESSING 115
5.1.1 LIBS PROCESSING (CDRS) 115
5.1.2 RAMAN PROCESSING (CDRS) 116
5.1.3 TRLS PROCESSING (CDRS) 116
5.1.4 VIS PROCESSING (CDRS) 117
5.1.5 IRSPEC PROCESSING (CDRS) 117
5.1.6 MICROPHONE PROCESSING 118
5.1.7 RMI PROCESSING 118
5.2 CALIBRATED RDR PRODUCT TYPES 118
5.3 CALIBRATED RDR PRODUCT FORMAT 119

5.4 CALIBRATED RDR PRODUCT STRUCTURE 119
5.4.1 HDU COMMON TO ALL DATA PRODUCTS 121
5.4.2 TECHNIQUE-DEPENDENT HDSS 122
5.5 DERIVED RDR GENERAL PROCESSING 124
5.5.1 LIBS DDR 124
LIBS ELEMENTAL ABUNDANCES 124
5.5.2 RAMAN DDR 125
RAMAN FEATURES: 125
5.5.3 TRLS DDR 126
5.5.4 VIS DDR 126
5.5.5 IRS DDR 126
5.5.6 RMI MOSAIC DDR 126
5.5.7 MASTER TARGET DDR 127
5.6 DERIVED RDR PRODUCT TYPES 127
5.6.1 LIBS DDR PROCESSING 127

6 STANDARDS USED IN GENERATING PRODUCTS 129

6.1 FILE NAMING STANDARDS 129
6.1.1 EDR FILENAME 129
6.1.2 SUPERCAM NON-IMAGING EDR FILENAME SPECIFICS 130
TABLE 6-2 DESCRIBES THE CONTENTS OF THE SUPERCAM NON-IMAGING EDR FIELDS. 130
6.1.3 SUPERCAM CDR FILENAME SPECIFICS 144
6.1.4 SUPERCAM DDR FILENAME SPECIFICS 146
6.2 PDS STANDARDS 147
6.3 TIME STANDARDS 148
6.4 COORDINATE FRAME STANDARDS 148
6.4.1 ROVER NAVIGATION (ROVER NAV) FRAME 148
6.4.2 ROVER MECHANICAL (ROVER MECH) FRAME 151
6.4.3 LOCAL LEVEL FRAME 151
6.4.4 SITE FRAME 151
6.4.5 RSM FRAME 153
6.4.6 ARM FRAMES 153

7 APPLICABLE SOFTWARE 154

7.1 UTILITY PROGRAMS 154
7.2 APPLICABLE PDS SOFTWARE TOOLS 155
7.3 SOFTWARE DISTRIBUTION AND UPDATE PROCEDURES 155

APPENDIX A – CALIBRATION TARGETS 156

APPENDIX B – DATA PRODUCT NAMES (SAME AS APID NAMES) **157**

APPENDIX C – PRODUCT LABEL KEYWORD DEFINITIONS, VALUES, SOURCES **158**

APPENDIX D – BODY UNIT AND MAST UNIT SOH 160

LIST OF FIGURES

Figure 1a, b, c SuperCam Overview Images 20
Figure 2 Schematic diagram of SuperCam Components 21
Figure 3 SuperCam Body Unit (BU) 22
Figure 4 SuperCam Mast Unit (MU) 24
Figure 5 SuperCam Operations Scenario. 27
Figure 6 SuperCam component synergy. 28
Figure 7 SuperCam typical observing sequence. 33
Figure 8 SuperCam Raman spectroscopy in scan mode 36
Figure 9 SuperCam VISIR spectroscopy in scan mode 37
Figure 10 M2020 instrument data flow to IDS. 44
Figure 11 M2020 EDR Filename Template 131
Figure 12 Rover Navigation (RNAV) Coordinate Frame 150
Figure 13 Yaw, Pitch, and Roll Definitions 151
Figure 14 Site and Rover Frame Examples. 152

LIST OF TABLES

Table 1-1 Interface Relationships 17
Table 3-1 PDS4 Processing levels for Instrument Experiment Data Sets 38
Table 4-1 SuperCam Product ID (PID) 45
Table 4-2 SuperCam ScamScidata Science Products 46
Table 4-3 Mars 2020 SuperCamData Products to be Archived 48
Table 4-4 ScamSciData File Description 56
Table 4-5 Format of SuperCam FITS EDRs 58
Table 5-1 Mars 2020 SuperCam CDR product ID 118
Table 6-1 Product Type for SuperCam Technique 129
Table 6-2 Contents of the SuperCam Non-imaging EDR filename fields 130
Table 6-3 M2020 EDR Filename Convention 132
Table 6-4 SuperCam CDR File Naming Scheme 145
Table 6-5 SuperCam Microphone CDR File Naming Scheme 145
Table 6-6 SuperCam RMI CDR File Name Scheme 147
Table 6-7 Coordinate Frames used for M2020 Surface Operations 148
Table 7-1 Key Software Toolsets 154

ACRONYMS AND ABBREVIATIONS

AIDS	Assembly and Inspection Data Sheet
AEGIS	Autonomous Exploration for Gathering Increased Science
AFT	Allowable Flight Temperature
AOTF	Acousto-Optic Tunable Filter
APID	Application Product ID (telemetry type)
ASCII	American Standard Code for Information Interchange
ATLO	Assembly, Test, Launch and Operations
ATM	Atmospheric Sciences Node at PDS
AZ	Azimuth
BCE	Bench Checkout Equipment
BU	Body Unit
CAD	Computer Aided Drafting
CBE	Current Best Estimate
CCD	Charge Coupled Device
CDR	Calibrated Data Record
CDS	Circuit Data Sheet
C&DH	Command and Data Handling
CDRL	Contract Data Requirements List
CESR	Centre d'Etude Spatiale des Rayonnements
CG	Center of Gravity
CIS	Cartography and Imaging Sciences Node at PDS
CMOS	Metal Oxide Semiconductor
CNES	Centre National d'Etudes Spatiales
CODMAC	Committee on Data Management and Computation
CPU	Central Processor Unit
CWL	Continuous Wave Laser
DEA	Digital Electronics Assembly
Demux	Demultiplexer
Downlink	Data transmitted from spacecraft to ground
DP	Data Product (identical to DPO)
DPO	Data Product Object
DPU	Digital Power Unit
DSN	Deep Space Network
Ebox	Electronics Box (of the Mast Unit)
ECA	Environmental Control Area
EDL	Entry, Descent and Landing
EDR	Experiment Data Record

EDU	Engineering Development Unit
EIDP	End Item Data Package
EL	Elevation
EM	Engineering Model
EMI	Electromagnetic Interference
EMC	Electromagnetic Compatibility
EQM	Engineering Qualification Model
ESD	Electro-Static Discharge
ERD	Environmental Requirements Document
FA	Flight Acceptance
FDD	Functional Design Description
FDR	Fundamental Data Record
FEI	File Exchange Interface
FITS	Flexible Image Transport System
FM	Flight Model
FOC	Fiber Optic Cable
FOV	Field of View
FPA	Focal Plane Array
FPGA	Field-Programmable Gate Array
FRD	Functional Requirements Document
FS	Flight System
FSRD	Flight System Requirements Document
FSW	Flight Software
GDS	Ground Data System
GEO	Geological Sciences Node at PDS
GSE	Ground Support Equipment
HDR	High Dynamic Range
HDU	Header Data Unit
HEPA	High-Efficiency Particulate Air
HRCR	Hardware Requirements Certification Review
HRS	Heat Recovery System
HSS	High Speed Serial
ICCD	Intensified CCD (gain test)
ICD	Interface Control Document
i-cmd	Instrument command (command from RCE to SCBU)
IDS	Instrument Data System
IEEE	Institute of Electrical and Electronics Engineers (engineering standard)
IOM	Inter-Office Memorandum
IRD	Interface Requirements Document
IRAP	Institut de Recherche en Astrophysique et Planétologie
IRS	InfraRed Spectrometer
JMS	Java Message Server
JPL	Jet Propulsion Laboratory
LANL	Los Alamos National Laboratory
LIBS	Laser-Induced Breakdown Spectroscopy
LVDS	Low Voltage Differential Signal
MEL	Mass Equipment List
MER	Mars Exploration Rovers
MIC	SuperCam Microphone
MIPL	Multimission Image Processing Laboratory
MIUL	Materials Identification and Usage List
M2020	Mars 2020

MMRTG	Multi-Mission Radioisotope Thermoelectric Generator
MPCS	Mission Data Processing and Control Subsystem
MSL	Mars Science Laboratory
MSPSP	Mission System Prelaunch Safety Package
MU	Mast Unit
NAIF	Navigation and Ancillary Information Facility (for NASA)
OBox	Optical Box (of the Mast Unit)
OCS	Operations Cloud Store
ODL	Object Description Language
PDR	Preliminary Design Review
PDS	Planetary Data System (NASA's Planetary Science Archive)
PEL	Power Equipment List
PFR	Problem / Failure Report
PID	Product ID (EDR type)
PRT	Platinum Resistance Thermometer
PSRD	Project System Requirements Document
RAMP	Rover Avionics Mounting Panel
PDMS	Project Data Management System
PI	Principal Investigator
PP	Planetary Protection
PRT	Platinum Resistance Thermometer
RCE	Rover Compute Element (Rover computer)
RAMP	Rover Avionics Mounting Panel
RDR	Reduced Data Record
RMCA	Rover Motor Control Assembly
RMI	Remote Micro-Imager
RMC	Rover Motion Counter
ROI	Region of Interest
RPAM	Rover Power and Analog Module
RSM	Remote Sensing Mast
RTV	Room Temperature Vulcanization (silicone)
RWEB	Remote Warm Electronics Box
S-cmd	Spacecraft Command
SCBU	SuperCam Body Unit
SCCT	SuperCam Calibration Target
SCD	Source Control Drawing
SCMU	SuperCam Mast Unit
SFDU	Standard Formatted Data Unit
SINDA	Systems Improved Numerical Differencing Analyzer
SIS	Software Interface Specification
SOAS	Science Operations Analysis Software
SOH	State of Health
SPICE	Observation Geometry System for Space Science Missions (used by NASA NAIF)
STT	System Thermal Test
TBC	To Be Confirmed
TBD	To Be Determined
TBR	To Be Reviewed
TBS	To Be Specified
TBU	To Be Updated
TDS	Telemetry Data System
TEC	Thermoelectric Cooler

TRLS	Time-resolved fluorescence spectroscopy
TRT	Temperature Requirements Table
TU	Testbed Unit
Uplink	Commands transmitted from ground to spacecraft
UV	Ultraviolet
VIO	Violet
VICAR	Video Image Communication And Retrieval
VISIR	Visible and Infrared Spectrometer
VNIR	Visible to Near InfraRed
WEB	Warm Electronics Box
XDR	Extracted Data Record
XML	eXtended Markup Language
YAG	Nd:YAG laser

1 INTRODUCTION

1.1 Purpose and Scope

The purpose of this Data Product Software Interface Specification (SIS) is to provide consumers of SuperCam instrument Experiment Data Record (EDR) data products with a detailed description of the products and how they are generated, including data sources and destinations. Content in this document supports EDR data products generated by the Mars 2020 Instrument Data System (IDS) for the non-imaging components of the SuperCam instrument. These components include:

- Laser-Induced Breakdown Spectrometer (LIBS)
- Raman spectrometer
- Passive spectroscopy - Visible and Infrared spectrometer (VIS IR)
- Time-resolved luminescence spectroscopy (TRLS)
- Microphone
- Instrument state-of-health data

Data products from the SuperCam Remote Micro-Imager (RMI) are described in a separate document, D-99960 Camera Instrument EDR and RDR Data Products, along with all other imaging systems in the Mars 2020 rover science payload.

The users for whom this SIS is intended include IDS, users and developers of Science Operations Analysis Software (SOAS), member scientists of the project's Science Team who will analyze the data, and other scientists in the general planetary science community.

In this document, the EDR data product is the raw, uncalibrated, uncorrected data acquired by SuperCam. It may include decompression if there was data product compression performed onboard the rover by the instrument.

1.2 Contents

This Data Product SIS describes how the EDR data product is acquired by SuperCam and how it is processed, formatted, labeled, and uniquely identified. The document discusses standards used in generating the product and software that may be used to access the product. The EDR data product structure and organization is described in sufficient detail to enable a user to read the product. Finally, examples of composite EDR labels are provided, along with the definitions of the keywords in the label.

1.3 Constraints and Applicable Documents

This SIS is meant to be consistent with the contract negotiated between the M2020 Project and the M2020 Principal Investigators (PI) for the SuperCam instrument, in which experiment data records and documentation are explicitly defined as deliverable products. Because this SIS governs the specification of data products used during mission operations, any proposed changes to this SIS must be impacted by all affected software subsystems observing this SIS in support of operations (e.g., IDS).

Product label keywords may be added to future revisions of this SIS. Therefore, it is recommended that software designed to process EDRs specified by this SIS should be robust to (new) unrecognized keywords. Similarly, entirely new products may be added over time.

This Data Product SIS is responsive to the following M2020 and PDS documents:

Mars 2020 Cruise, Entry, Descent, and Landing (CEDL) and Surface SuperCam Functional Design Description (FDD), “Baseline Release, Rev A”, Ivair Gontijo, JPL D-95868
Mars 2020 Flight-Ground Interface Control Document (FGICD), “Volume 1, Downlink, Rev A, Version 1.0”, Biren Shah, JPL D-95521, October 3, 2017

Additionally, this SIS isconsistent with the following Planetary Data System documents. These documents are subject to periodic revision. The most recent versions may be found at https://pds.nasa.gov/datastandards/documents/. The PDS products described in this SIS have been designed based on the versions current at the time, which are those listed below..

Planetary Data System Standards Reference, version 1.15.0.0, October 2, 2020.
Planetary Data System Data Provider’s Handbook, version 1.15.0.0, October 2,2020.
PDS4 Common Data Dictionary, Abridged, version 1.15.0.0, October 2, 2020.
PDS4 Information Model Specification, version 1.15.0.0, December 23, 2020.

Finally, this SIS makes reference to the following documents for technical background information:

Sensing and Perception Research for Space Telerobotics at JPL, D. Gennery et al., *Proceedings of the IEEE Intern. Conf. on Robotics and Automation,* March 31 - April 3, 1987.

Anderson, R.C., et al., Mars Science Laboratory Participating Scientists Program Proposal Information Package, December 14, 2010.

Mars Exploration Rover (MER) Project ICER User's Guide, Aaron Kiely, MER 420-8-0538, JPL D-22103, January 5, 2004.

Malvar, H.S., Li-Wei He, and R. Cutler, "High-quality linear interpolation for demosaicing of Bayer-patterned color images", Proceedings, IEEE Intl. Conf. on Acoustics, Speech, and Signal Processing (ICASSP), I.2004.

Maki, J.N., et al., The Mars 2020 Engineering Cameras and Microphone on the Perseverance Rover: A Next-Generation Imaging System for Mars Exploration, Manuscript Draft for Space Science Reviews, 2020.

DeLapp, D., Pilleri, P. et al., The SuperCam EDR, RDR PDS User Guide, 2021

Wiens, R.C., et al., Pre-flight calibration and initial data processing for the ChemCam laser-induced breakdown spectroscopy instrument on the Mars Science Laboratory rover, Spectrochimica Acta Part B: Atomic Spectroscopy, Volume 82, 2013, doi:10.1016/j.sab.2013.02.003.

Wiens, R.C., Maurice, S., Robinson, S., Nelson, A., Cais, P. et al. The SuperCam Instrument Suite on the NASA Mars 2020 Rover: Body Unit and Combined System Tests. Space Science Reviews, Springer Verlag, 2021, 217 (1), pp.4. (10.1007/s11214-020-00777-5). (insu-03088041)

Maurice, S., Wiens, R.C., Bernardi, P. et al. The SuperCam Instrument Suite on the Mars 2020 Rover: Science Objectives and Mast-Unit Description. Space Sci Rev 217, 47 (2021). https://doi.org/10.1007/s11214-021-00807-w

1.4 Relationships with Other Interfaces

Changes to this EDR data product SIS document will propogate downstream to dependent products, software, and/or documents going forward.

Table 1-1 Interface Relationships

Name	Type P = product S = software D = document	Owner
IDS database schema	P	IDS (JPL)
M2020EDRGEN	S	IDS (JPL)
M2020 SuperCam Non-imaging Instrument EDRs	P	IDS (JPL)
RSVP	S	RSVP Dev Team (JPL)
ASTTRO	S	ASTTRO Dev Team (JPL)
RGIS	S	RGIS Dev Team (JPL)

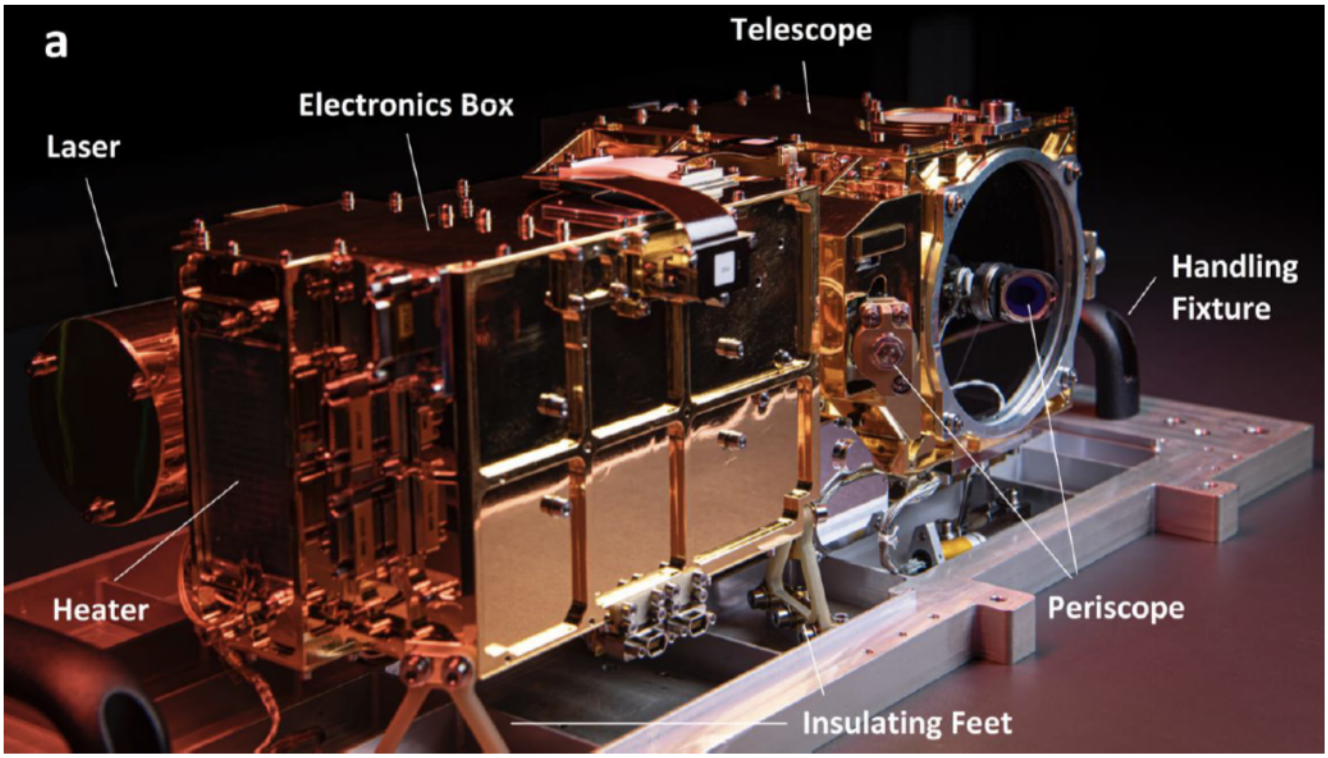
2 INSTRUMENT OVERVIEW

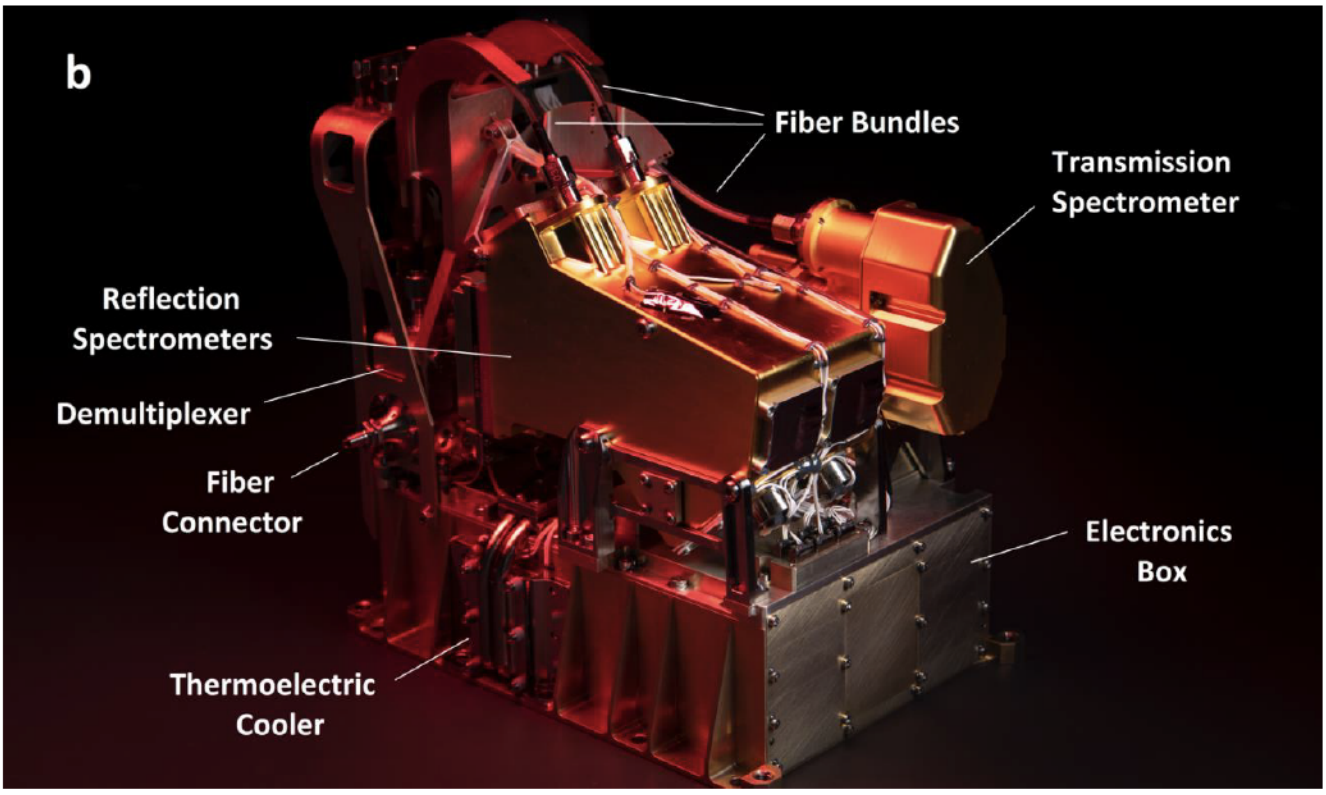
SuperCam is an integrated multi-sensor remote-sensing instrument carried aboard the Mars 2020 rover. It implements four spectroscopic techniques to remotely measure properties of materials on Mars. These include:

- Laser-induced breakdown spectroscopy (LIBS)
- Raman spectroscopy (Raman)
- Time-resolved fluorescence spectroscopy (TRLS)
- Passive spectroscopy - Visible and Infrared spectrometer (VISIR or VIS)

SuperCam also includes a telescopic color context camera, the Remote Micro-Imager (RMI), which is principally used to document the sites of remote spectroscopy measurements. Finally, the instrument also includes a microphone, which is used primarily to measure the pressure wave produced by the generation of plasma at a LIBS target. As a science instrument, it produces a single science data product at each observing point (target or raster point) per observation mode (LIBS, Raman, TRLS, VISIR, microphone). SuperCam builds on the experience of ChemCam on MSL. This former instrument had very much the same overall architecture and operation scheme but only included LIBS and RMI.

SuperCam's spectrometers can be pointed at targets in the rover's environment, thanks to key components being mounted on the rover's Remote Sensing Mast. The SuperCam Mast Unit (SCMU) is the remote sensing head that interrogates the Martian surface by firing laser pulses and collecting light back through a small telescope on the rover mast. The light collected is directed into an optical fiber and sent to the body unit for spectral analysis. The SCMU also contains an infrared spectrometer (IRS) for passive spectroscopic analysis at long wavelengths, without the use of its lasers. The microphone and Remote Micro-Imager (RMI) are also located in the SCMU, which is housed on the M2020 rover sensing mast (RSM). The SuperCam Body Unit (SCBU) is housed inside the rover body, in direct contact with the thermal RAMP (Rover Avionics Mounting Panel). This unit performs most of the light spectral analysis in its 3 spectrometers: UV, violet (VIO), and visible to near-infrared (VNIR). In passive mode (no laser) the spectra of interest come from the IRS, the VIO and the VNIR, a combination referred to as VisIR. The SCBU interfaces with the rover for electrical power and communication. It also interfaces with, and provides power to, the Mast Unit. Mounted close to the Multi-Mission Radioisotope Thermoelectric Generator (MMRTG) on the rover, the SuperCam Calibration Target (SCCT) is about 1.56 m away from the RSM. It contains representative samples of minerals and crystals and will be used to refine/cross-calibrate/validate the calibration of the instrument established before launch, on Mars (see Appendix A). It also carries a Titanium target for LIBS calibration, black and white targets for IR spectroscopy calibration and geometrical patterns (USAF1951 target and others) for optical performance assessment and calibration of the RMI camera.


a
Telescope
Electronics Box
Laser
Handling Fixture
Heater
Periscope
Insulating Feet


b
Fiber Bundles
Transmission Spectrometer
Reflection Spectrometers
Demultiplexer
Fiber Connector
Electronics Box
Thermoelectric Cooler

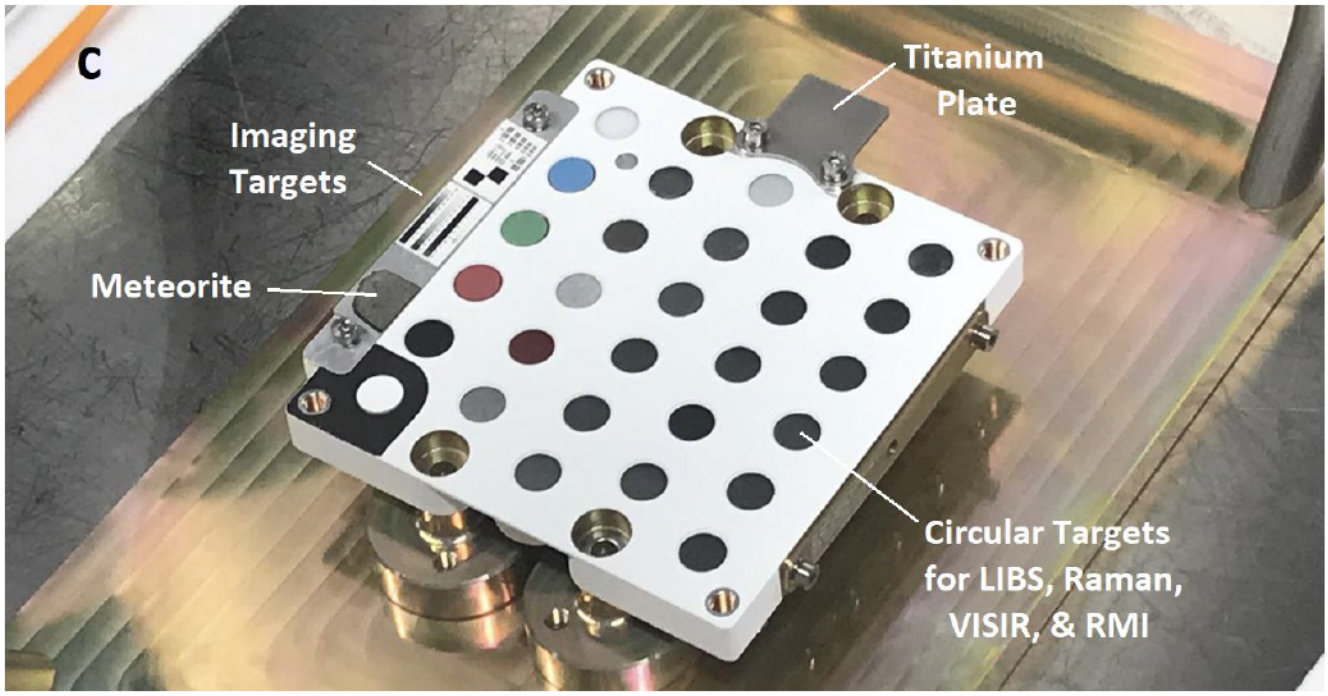


Figure 1a, b, c SuperCam Overview Images

The SuperCam instrument is shown in Figure 1, consisting of the Mast Unit (MU, top), the Body Unit (BU, middle), and the SuperCam Calibration Target (SCCT, bottom). As part of the MU (top), the laser can be seen to the left, protruding from behind the electronics box. The telescope is at the far end, at the center of which the periscope mirror for the green laser beam is mounted. The corresponding periscope mirror can be seen just past the electronics box, facing away from the camera. At the near end of the electronics box, a heating pad is just to the left of the connectors. The Mast Unit is mounted on insulating feet, and is shown here resting on a handling fixture. The BU (middle) shows a transmission spectrometer resting behind two identical reflection spectrometers, all mounted on top of the electronics box. Three optical fiber bundles can be seen with their protective shields near the upper left; these transfer light to the spectrometers from the demultiplexer. The only part of the demultiplexer that is visible is the AVIM fiber connector, protruding at the left center. This is where the light from the Mast Unit enters the Body Unit. One of three sets of thermo-electric coolers are seen in the lower center, identified by two visible heat pipes that run under the spectrometers to cool their detectors. On the SCCT (bottom), twenty-nine circular targets and several other calibration targets are mounted. The protruding titanium plate at the upper right is used for wavelength calibration via LIBS spectra. Optical targets line the left side.

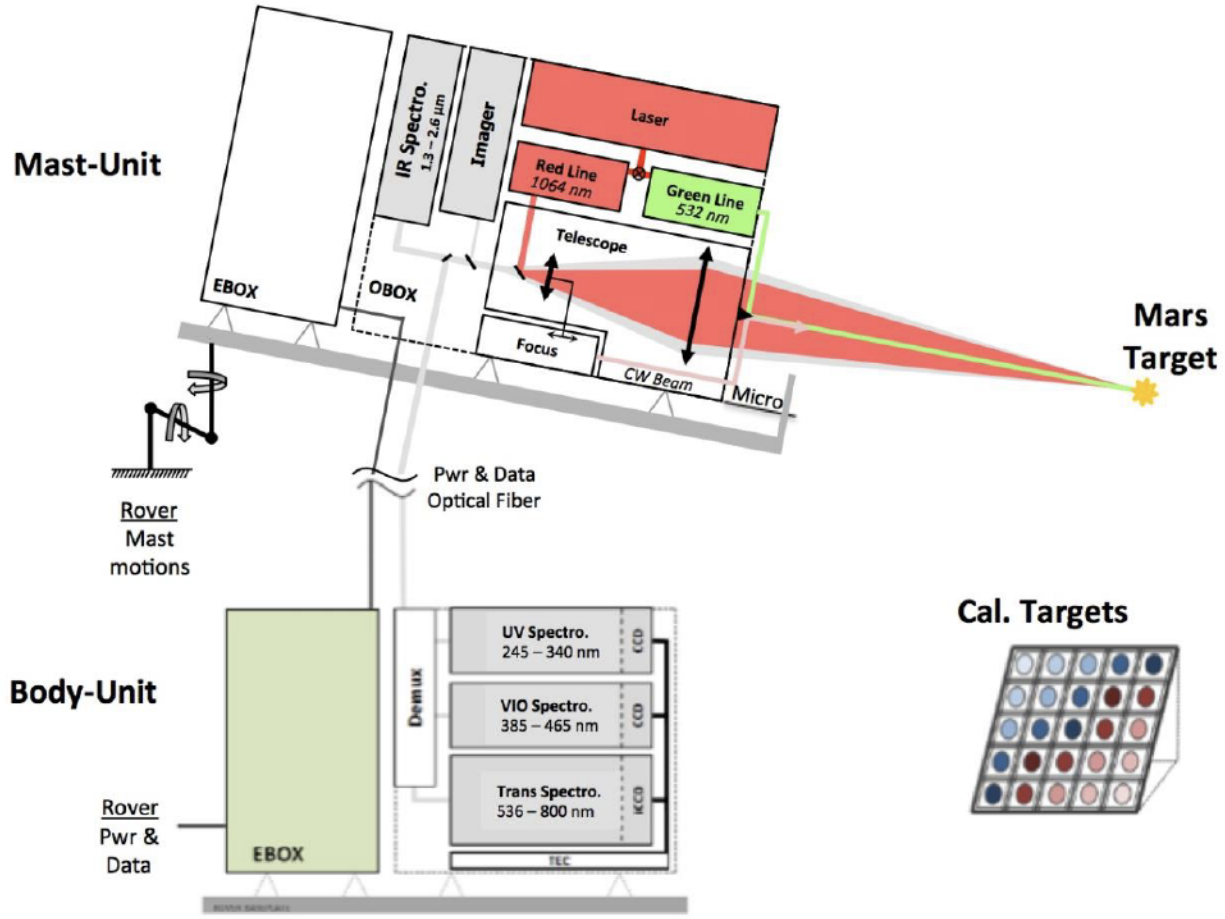


Figure 2 Schematic diagram of SuperCam Components

The Mast Unit (MU) consists of the main laser which provides two wavelengths, the telescope, a continuous-wave (CW) laser for focusing, and a microphone. The optical box (OBOX) is completed by the infrared (IR) spectrometer and the Remote Micro-Imager (RMI). An electronics box (EBOX) controls and powers the various subsystems in the MU. Acquisition of the target is provided by the rover mast motions. Electrical cable and an optical fiber connect the Body Unit (BU) to the MU. The fiber carries light in the 245-853 nm range to the demultiplexer (demux) in the BU, which distributes the light into three spectrometers covering ultraviolet (UV), violet (VIO), and visible to near infrared spectral ranges. The latter is characterized by a transmission spectrometer. The electronics box (EBOX) in the BU operates the instrument, provides power to the BU spectrometers and the MU, and communicates with the rover. A set of calibration targets (Cal Targets) are mounted on the back of the rover to facilitate calibration while on Mars.

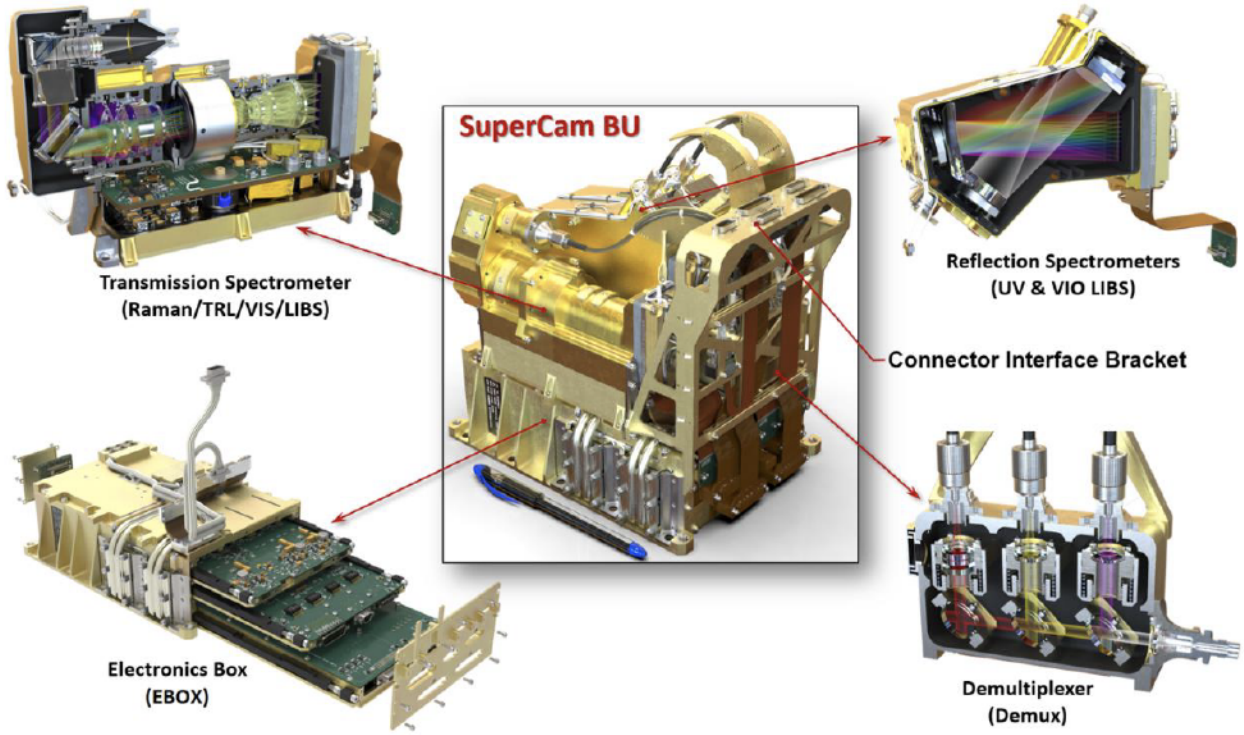
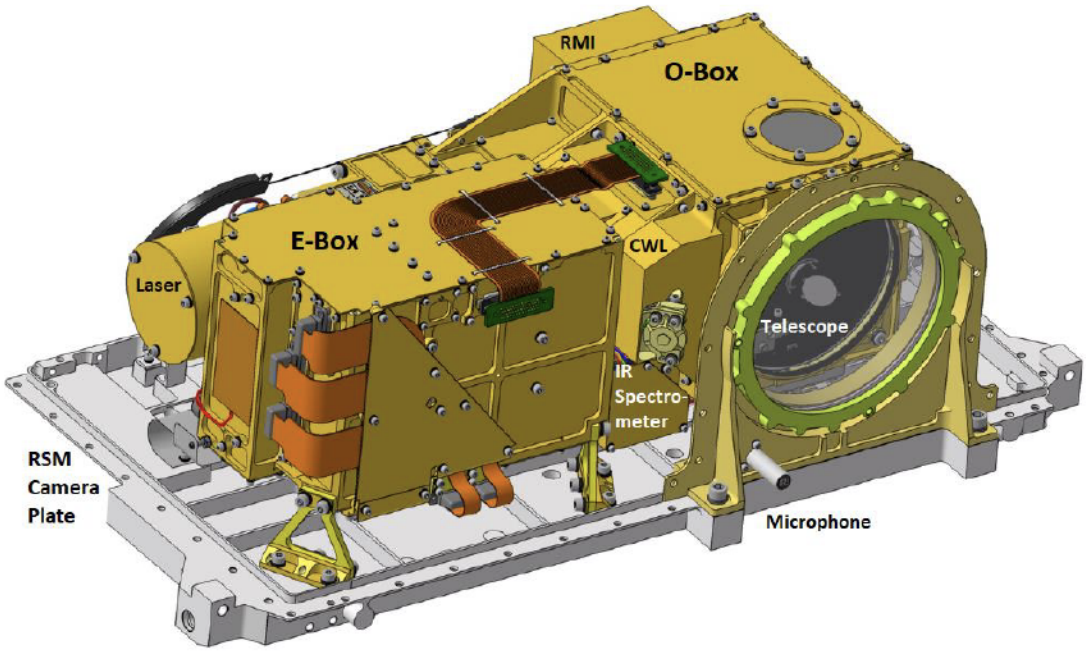
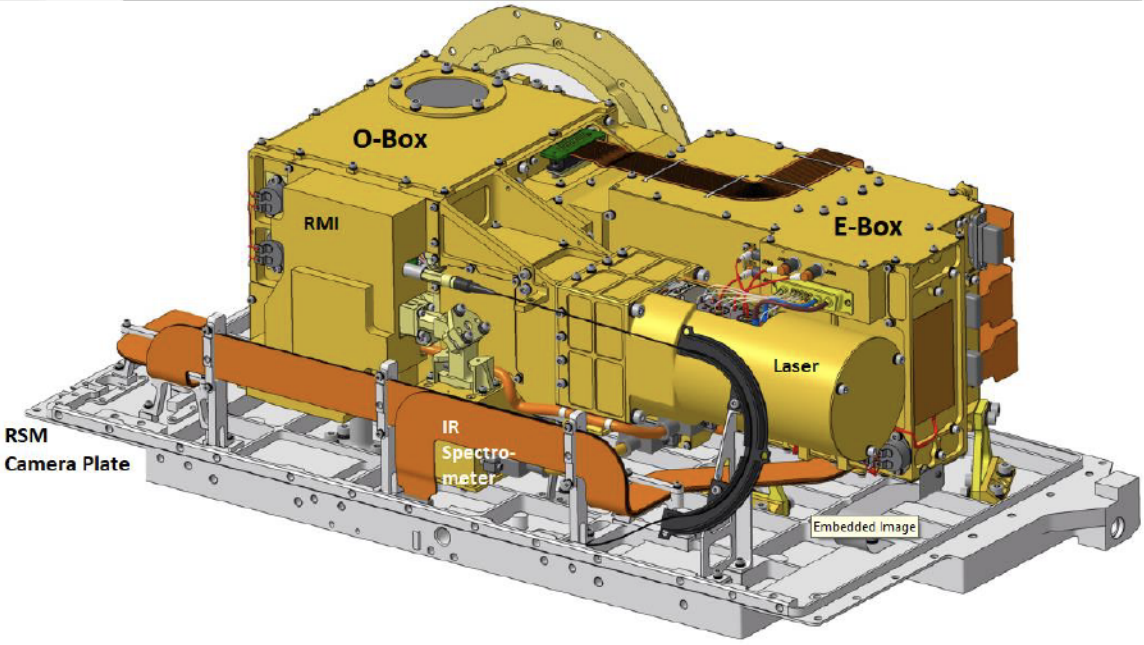


Figure 3 SuperCam Body Unit (BU)

SuperCam Body Unit is shown in Figure 3. The base of the Body Unit contains DPU (Digital Power Unit) electronics and power supplies and is thermally sunk to the RAMP (Rover Avionics Mounting Plate). On top of, and thermally isolated from, the base (and the RAMP) are the instrument's Raman and UVIS Spectrometers.


RMI
O-Box
E-Box
CWL
Laser
Telescope
IR
Spectro-
meter
RSM
Camera
Plate
Microphone


O-Box
RMI
E-Box
Laser
IR
Spectro-
meter
RSM
Camera Plate
Embedded Image

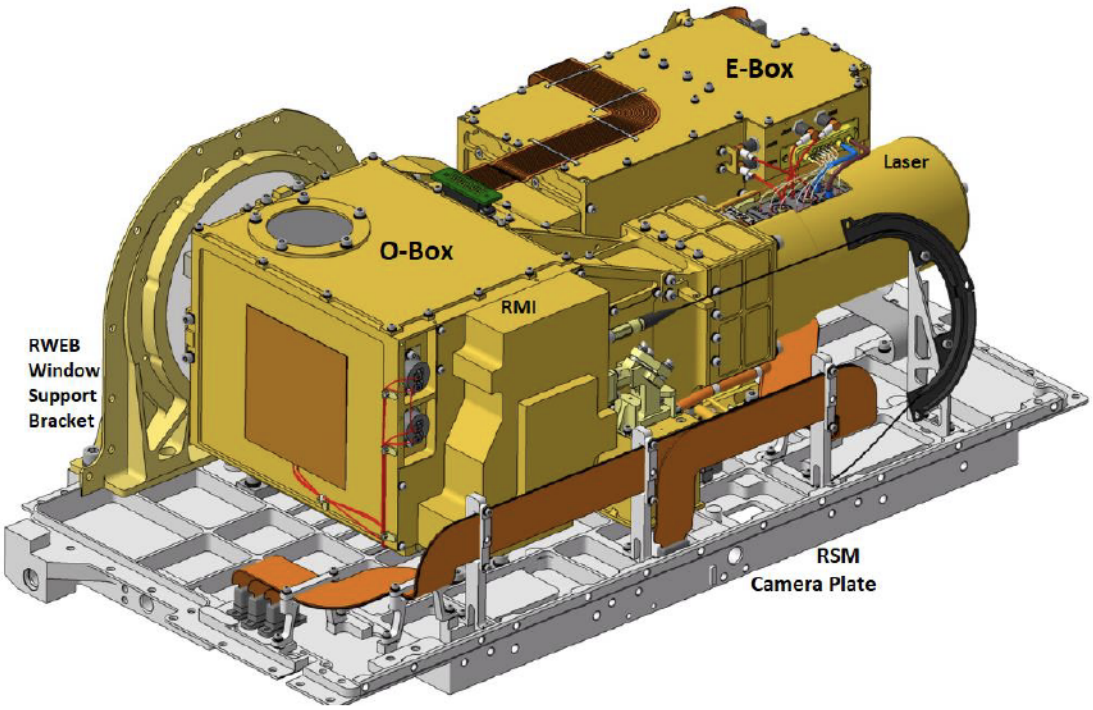


Figure 4 SuperCam Mast Unit (MU)

The SuperCam Mast Unit is shown in Figure 4, shown from three different angles. The Optical Box (O-Box) contains SuperCam's telescope (green ring and adjacent box, top image), Remote Micro Imager (RMI) (back of telescope optics box), and the Continuous Wave Laser (CWL) used for focusing. The tubular horizontal housing towards the rear of the Mast Unit contains the LIBs/Raman laser and is rigidly attached to the O-Box. Attached underneath the O-Box is the IR spectrometer. The Electronics Box (E-Box) (adjacent to the laser) contains the Mast Unit's controller electronics. The O-Box and E-Box are mounted to the RSM (Remote Sensing Mast) camera plate on flexures. The microphone is mounted on the RWEB (Remote Warm Electronics Box) window support bracket. The details for the mechanical accommodation are contained in the SCMU Source Control Drawing (SCD).

2.1 Investigation Techniques

SuperCam will use a suite of spectroscopic, optical and acoustic techniques to investigate the Martian environment. An overview of these techniques is provided here and after that they will be grouped into instrument "operation modes". The surface of Mars is covered by dust. Using the LIBS laser we can remove or ablate the dust. The LIBS, Raman, and VISIR analyses can be done in any order, but the Raman and VISIR normally benefit from the partial dust removal provided by LIBS, so usually they will be done after the LIBS. There is also the possibility to do a small number (5) shots to remove dust without doing the full standard 30 shot series. Figure 5 illustrates the general operational flow of SuperCam, while Figure 6 discusses the relative footprints and ranges of the cameras and spectrometers.

2.1.1 LIBS Active Spectral Data Collection:

This technique uses the mast unit LIBS laser beam to generate plasma clouds from target rocks, analyzing the resultant emissions with the body unit spectrometers to provide insight into the chemical makeup of the target rock. The LIBS laser is infrared ("red" laser). Single point observations as well as rasters made up of a collection of points will be achievable in this observation mode. The standard LIBS analysis will be 30 laser enabled shots. The wavelength range is ~240-855nm.

We will also use the spectrometers to collect background light from the environment without the laser, using the same exposure duration. These "LIBS darks" are subtracted from the active LIBS spectra as part of the data processing. Darks will be taken just before, just after, or before and after the active collection. Typically, 30 darks will be averaged for better statistics; this can be done onboard to save data volume. Dark (i.e. no laser) spectra are subtracted from the laser-active spectra to remove instrument noise or background.

2.1.2 Raman Active Spectral Data Collection:

SuperCam Raman spectroscopy uses 532 nm green Raman laser beam to induce inelastic Raman scattering in a material where shifts in the scattered light are related to material properties, such as molecular bonds that can be used to determine mineralogical composition.The Raman signal is analyzed with the body unit transmission spectrometer. A short 100 ns exposure duration is used to minimize interference from luminescence. Single point observations as well as rasters will be achievable in this observation mode. The spectral exposure delay will need to be advanced for distant targets to allow the light time to return to the instrument. (Round-trip light time for each 10 m increment is ~60 ns). The minimal number of shots is 100, which may be collected as 10 coadditions 10 times.

Raman darks will also be taken either just before or just after the active exposures. Dark (i.e. no laser) spectra are subtracted from the laser-active spectra to remove instrument noise or background.

2.1.3 Time-Resolved Luminescence Data Collection:

This is a sensitive discriminator of short-lived organic matter (typically < 200 ns) from longer-lived inorganic luminescence centers (typically <1 ms). The SuperCam time-resolved luminescence experiment is designed to operate by collecting spectra at variable delay times, where delay refers to the time between firing the green laser and triggering the intensifier. Identification of organic material is accomplished by scanning the minimum 50ns intensifier integration window from 0 to 200 ns delay. In contrast, long-lived inorganic luminescence is detected with a long intensifier window, generally from between 1 us to 0.5 ms, the expected max delay is <20 ms and width of <0.5 ms from the laser trigger. It is also possible to collect a single luminescence spectrum, either at a time delay that collects the Raman signal as well as luminescence or shortly after the Raman signal has completed to avoid confusion between the two signals. Single point observations as well as rasters will be achievable in this observation mode. No-laser "dark" spectra will also be taken either just before or just after the active exposures. Dark (i.e. no laser) spectra are subtracted from the laser-active spectra to remove instrument noise or background.

2.1.4 VISIR Spectral Data Collection:

Using the infrared spectrometer on the Mast Unit, the VIO reflection spectrometer and the transmission spectrometer on the body unit, reflected sunlight from the targets is collected and analyzed. The objective of VISIR is the identification of minerals. The transmission spectrometer is set to low gain and an exposure time in the tens to hundreds of microseconds. The Acousto-Optic Tunable Filter (AOTF) provided by Gooch and Housego (Iliminster, England) is the heart of the spectrometer and performs the wavelength selection. The IR spectrometer operates by scanning of an AOTF across the wavelengths from 1.3-2.6 microns. The exposure times are estimated based on the time of day and the type of target, either through an autoexposure algorithm or by manually selecting a given exposure time. The exposure time can vary as a function of wavelength. Dark exposures are collected as a separate observation using non-illuminated rows in the VIO spectrometer and low gain and the minimum gate time for the transmission spectrometer regions. Dark measurements are interleaved between each wavelength for the IR spectrometer. Dark (no laser) spectra are subtracted from the laser-active spectra to remove instrument noise or background. Referring to the circular diagram in Fig. 6, compositions can be determined from mineral characterization techniques. Specifically, from the VIS spectral range (400-850 nm), the oxidation state of iron affects the reflectance curve in this spectral range.

NOTE: An AOTF is an optical device acting as a monochromator to extract a monochromatic signal from a polychromatic incident light. It relies on the diffraction of incident light by non-standing acoustic waves in a tellurium dioxide (TeO2) birefringent crystal generated by an electro-acoustic transducer. In the frame of the Bragg's diffraction formalism, the acoustic waves stream behaves as a thick grating for which the phase matching condition leads to the emergence of a unique diffracted monochromatic light beam deviated from the non-diffracted polychromatic beam. The central wavelength of this beam is bijectively determined by the acoustic waves frequency (itself determined by the radio frequency signal supplying the transducer), and its spectral width is related to the crystal's geometry. Moreover, due to TeO2 birefringence, a second couple of diffracted and non-diffracted beams is symmetrically generated to the optical axis. Both couples of beams have the same properties excepted they are linearly cross-polarized. IRS' optical architecture uses a beam stop to block non diffracted beams and focuses both diffracted beams onto two photodiodes, one being nominal, the other being redundant.

2.1.5 Remote Microscopic Imager (RMI) Data Collection:

RMI uses the mast unit camera to take high-resolution color images of Mars. In this mode the square FOV of the instrument telescope is used to collect light from the environment and is captured in the mast unit 2048×2048 pixel CMOS device. The RMI data are beyond the scope of this document and are instead discussed in the M2020_EDR_Camera_SIS (D-99960). RMI spatial scale is calculated from pixel scale and distance to the target.

2.1.6 Microphone Data Collection:

Using the mast-unit-mounted microphone (MIC), SuperCam will record the LIBS plasma generation acoustic signatures, to determine quantity of material ablated from Martian rocks. It will also be possible to make Martian and rover sound recordings for maximum recording duration of 167 seconds at 25 kHz and 41 seconds at 100 kHz.

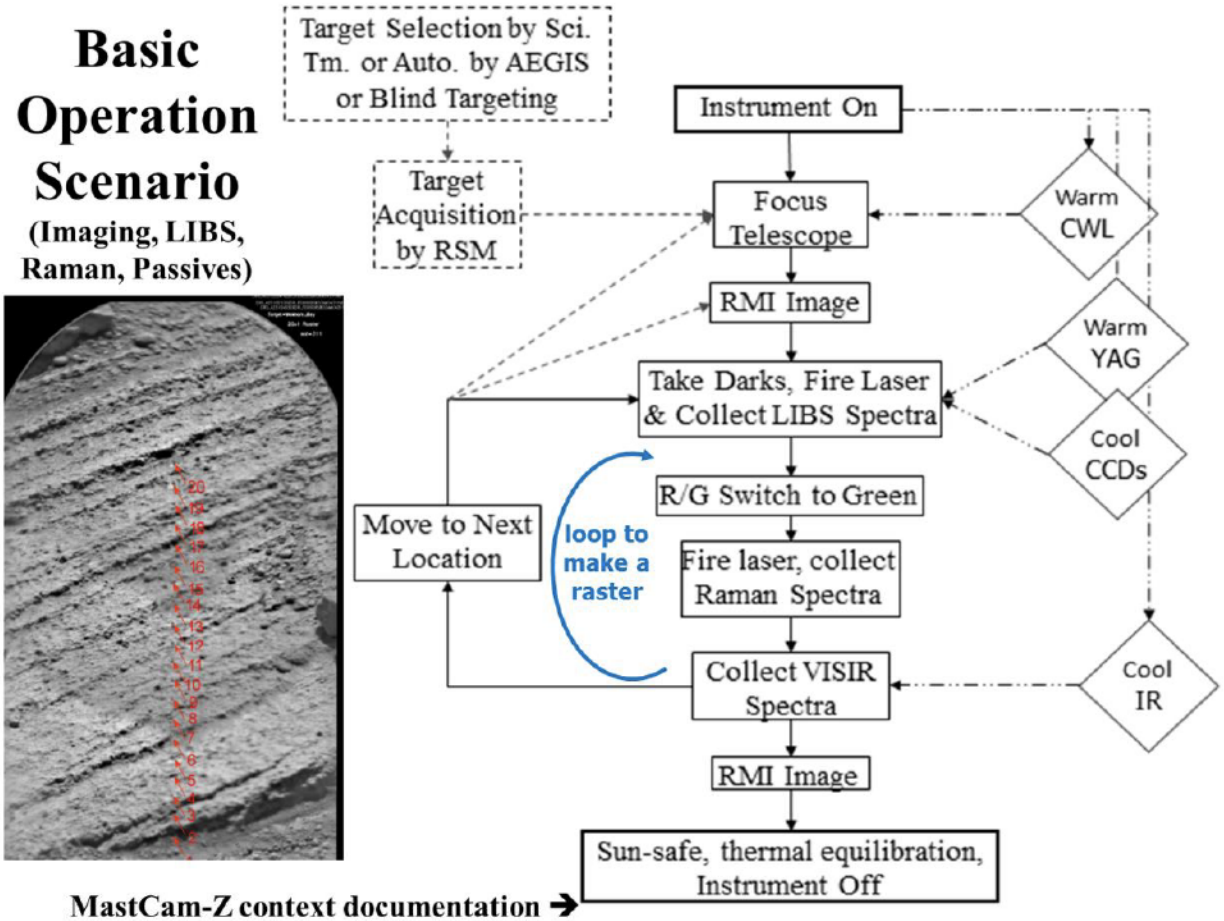


Figure 5 SuperCam Operations Scenario. "R/G" is "red/green" where "red" = LIBS infrared laser beam; "green" = green Raman laser beam. SuperCam switches laser beams when going from LIBS to Raman spectroscopy.

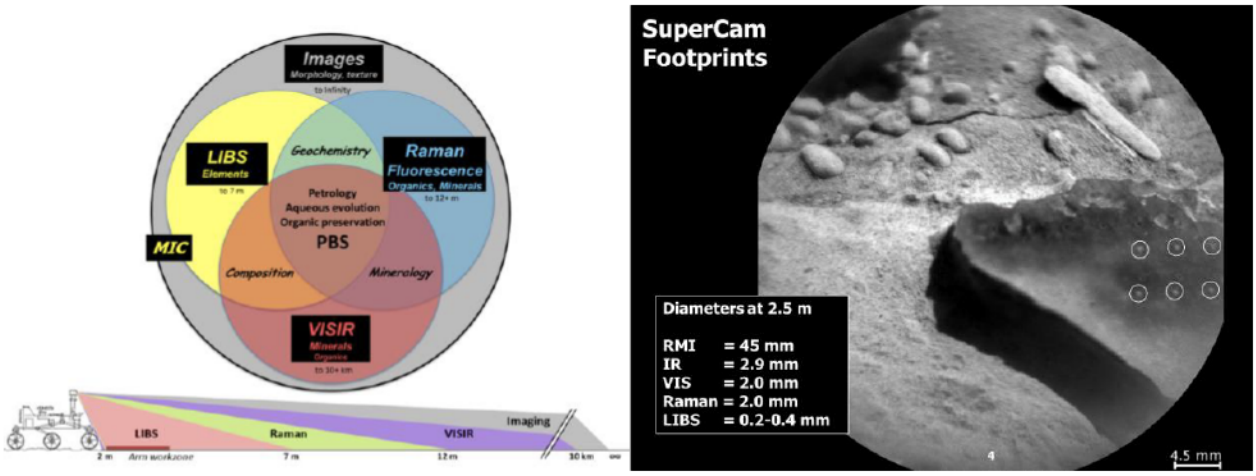


Figure 6 SuperCam component synergy, all working within the arm work zone and with typical measurement footprints. "MIC = microphone". The left upper Venn diagram shows the overlap between science topics explored by the SuperCam spectrometers, camera, and microphone. The left lower diagram shows the target distances from the rover which can be observed by LIBS, Raman, and VISIR spectroscopy. Distances are all inclusive (i.e. from 2 m to 12 m, in the case of Raman). The right diagram shows the relative footprint ("beam") size for RMI and the spectrometers for a target at 2.5 m. RMI range is 1 m to infinity.

2.2 Modes of Operation:

SuperCam has a rich set of operation modes, which can be grouped into three classes, the "Primitive Modes", "Observation Modes" and "Diagnostic or infrequently-used Modes". The 21 modes of operation are listed below, with short explanations. This is followed by a more detailed explanation of the "power-up mode" or "power-up sequence".

SuperCam "Primitive" Modes

1. **Power Off:** during cruise for example
2. **Power up:** the sequence of events that take the instrument from its off mode to its standby mode, including instrument flight software booting and the issue of TIME_SYNC.
3. **Idle:** power ON, lasers warmed up, CCDs cooled down, but all instruments are in standby. Laser is not fired, data is not collected
4. **Laser only:** only fire the high power laser, for dust cleaning purposes. No spectra are collected.SOH and laser data are returned.
5. **Spectra only:** passive VisIR reflectance spectra are collected. No laser is fired.
6. **Imaging only:** focus the telescope and take pictures with the RMI. No laser is fired. No spectra are collected.

7. **"Active Spectroscopy" modes:** modes when the laser is fired and either Raman or LIBS spectra are collected.
8. **Microphone-only mode:** sound recording of the wind or rover.

SuperCam Observation Modes:

1. **Raster survey:** 1x5, 1x10 or 1x20 one-dimensional rasters, 3x3 or other 2-D rasters. Might perform typical SuperCam operational sequence or a subset.
2. **Depth profile:** use LIBS to probe the first 10-500 um below the surface using a high number of laser shots
3. **Blind surveys:** use AEGIS or Simple Planner or a simpler sequence (as on MSL-ChemCam) for untargeted days and soil surveys. On ChemCam these are done by pointing the RSM 90 degrees to starboard in azimuth and -42 degrees in elevation, focusing, and shooting.The instrument performs an autofocus and takes data in the form of 1x5, 1x10, or 1x20 rasters. This can be useful for traverse soil surveys in a long-term analysis.
4. **Fine scale mode:** targeted (LIBS+Raman+VisIR+RMI) from target pictures downlinked and analyzed the previous day. Targeted pointing within 7 m for full chemistry and mineralogy investigation. A RMI image is acquired and sent to the ground. The science team locates targets of interests (veins, nodules, laminae, etc.) within the image frame of reference. The next day, a new image is acquired and is processed by the onboard computer (AEGIS software developed on MSL) to redirect SuperCam to the precise location. For example, if a vein is of interest, the team sends to the AEGIS onboard they want to do a detailed analysis of a vein, a new RMI is generated the day after, and AEGIS will locate a vein on the new RMI and relocate SuperCam towards the vein to analyse it. Several targets can be designated on a single image. This allows fine pointing (closed-loop) and increases the odds to hit the desired target with the first try. The average offset between raster points is ½ mradian but finer offset will depend on performace of the RSM.
5. **Long raster (Scan Mode):** This mode is derived from the raster survey above. It consists of a ~10° exploration in azimuth for IRS. At the starting point, the procedure performs a single autofocus which is good for the whole scan, taking advantage of > ±10% focus tolerance for IR observations. The spectra are acquired like for a 1D raster, but to save time and allow a long raster, only IRS spectra are taken and they are grouped into a single data buffer in the BU instead of transmitting the data point by point to the rover Data volume is minimal since it is IRS-only (no VIS acquired). Long rasters can be repeated after re-pointing in elevation to cover an area rather than a single line, or the exploration in azimuth can continue with a new autofocus.

SuperCam Diagnostic Modes

1. **SCCT observations**. Observation of SuperCam Calibration Targets. These are used to calibrate the spectrometers for wavelength, check the laser output, check various spectrometer functions, and to provide checks to the elemental calibration model. IR cal and RGB (Red Green Blue = color) RMI calibration are also performed during these observations.
2. **Passive sky observations.** These are done to observe the atmospheric O_2 and H_2O compositions using the transmission and IR spectrometers. A series of active and dark measurements are made at two different elevation angles in the sky. The science contact is Tim McConnochie, who has made equivalent observations with ChemCam.
3. **RMI z-stacks.** A “stack” of images taken at a series of slightly different focal positions is used either to obtain better focus on a target that is rough or has a large range in z-direction across the field of view of the image. A focus-merge compression can be performed in the SCBU to render the entire field of view in focus.
4. **Image mosaics** to increase the RMI FOV. Image mosaics are routinely done to cover the field of view as a part of a large (e.g., 1x10 or 1x20) raster on SuperCam. Mosaics can also be taken by themselves, especially at long distances.
5. **CWL alignment test.** The alignment of the CWL can be tested by turning on the CWL and taking an RMI image.
6. **Two-dimensional spectral image.** The BU CCDs have a mode whereby a 2D image can be read out to check the overall image quality and detect bad pixels. These 2D images require a larger amount of memory (6 MB; 2 MB per CCD). .
7. **ICCD (Intensified CCD) gain test.** The ICCD can be tested by taking spectral images at various gain levels.
8. **LIBS z-stacks.** On occasion, the accuracy of the autofocus may be tested relative to the best LIBS focal position. For this a z-stack of LIBS data is taken, similar to the z-stacks taken with the RMI.
9. **LIBS time scans with the transmission spectrometer.** The LIBS emission peaks each have characteristic lifetimes in the hundreds to thousands of nanoseconds. Molecular emission peaks, such as from CaF, occur over many tens of microseconds. A series of LIBS shots may be taken at different exposure window durations and at different delay times. This may be done to characterize the plasma environment on Mars and compare it to that in the Mars chambers on Earth. The procedure is somewhat similar to that of time-resolved luminescence only it is done with LIBS plasmas. This cannot be done over the UV and VIO spectral ranges, which do not have ICCDs.

2.3 Data Formats

Spectrometer data is collected by a CCD with pixels arranged in a 2148 columns by 515 row array. There is one CCD for each of the three spectrometers in the Body Unit: the UV spectrometer, the VIO spectrometer and the transmission spectrometer. Spectral lines are projected onto the CCD array vertically across multiple rows within a column. To minimize the amount of data generated by each CCD, the CCDs sum together multiple rows in each column to obtain a spectral intensity value for that column. For both the UV and VIO spectrometers there is one set of spectral lines projected onto the CCD. Summing down the individual columns of the CCD array results in 2148 values per spectrometer (2048 active channels and 100 masked serial channels). However, to cover a larger range of wavelengths the design of the transmission spectrometer causes three sets of spectral lines to be projected onto the CCD separated vertically with blank rows in between. Each of these sets of spectral lines is separately summed together in the CCD's serial register. The initial un-calibrated wavelength for each pixel for each technique is documented in the files: SC_LIBS_DEFAULT_WAVELENGTH.CSV, SC_TRLS_DEFAULT_WAVELENGTH.CSV, SC_VIS_DEFAULT_WAVELENGTH.CSV and SC_RAMAN_DEFAULT_WAVELENGTH.CSV included with the SuperCam bundle.

The data have five different formats:

LIBS Data Format, using the three BU spectrometers generates five sets of spectral data: one each from the UV and VIO spectrometers and three from the transmission spectrometer. The amount of data that generated is:

2148 x 5 = 10740 channels x 16 bits/collect.

For LIBS, each set of five spectra consists of 21.4 kB of data. A single observation point typically consists of 30 shots, or 644 kB, with an equivalent amount for the corresponding dark spectrum. Unless a depth profile is desired, the median, mean, standard deviation is used for the dark, reducing its data volume to 64 kB. The LIBS EDR FITS file will contain a table of all 30 Active spectra and a table of the median, mean, standard deviation of the dark spectra. Thus, the measured spectra plus dark would become 708 kB per observation point.

Raman-Luminescence Data Format, using the transmission spectrometer generates three sets of spectral data. The amount of data that is then generated is:

2148 x 3 = 6444 channels x 16 bits/collect.

Each set of Raman spectra consists of 12.9 kB. As Raman spectra do not ablate the surface, there is no additional information in the individual spectra, so each Raman observation can consist of the statistics given above (mean, median and standard deviation), for 38 kB. A dark spectrum in the transmission spectral range usually consists of a number of individual spectra, so the statistics are again downlinked for the same volume (38 kB). Together they result in 76 kB for a Raman observation.

VIS IR Data Format, using VIO and the transmission spectrometers plus the IR spectrometer. No UV is acquired for surface targets. For the VIS part (VIOlet and transmission spectrometer), the data volume is:

8592 channels x 16 bits/collect
if both VIO and transmission spectrometer are used
6444 channels x 16 bits/collect if only the transmission spectrometer is used.

For the IR the data volume is:

256 channels x 24 bits/channel

RMI Data Format (using the CMOS imager)

RMI images will take up 8 MB (2048 x 2048 pixels, 16 bits) uncompressed. However, excellent compression schemes exist, both lossless and lossy, and on ChemCam we compress images by 5 - 20 times with essentially no loss of details.

For all non-image data (anything returned by SCAM_SEND_DATA s-cmd), compression must be lossless or no compression. A lossless compression, when no information is lost during compression/decompression, is obtained with ICER compression and a bits_per_pixel parameter set at 10 for single images, at 12 for multi-frame HDR images combining 3 or 4 frames, at 13 for multi-frame HDR images combining 5 to 7 images. All other ICER options and JPEG compressions are lossy, i.e. some information are lost during compression/decompression. Last, the images can be returned with no compression at all. Based on ChemCam's operations, SCAM has no visibility whether compression is used or not when data is received by the science team. This requirement is to make sure that data is NOT compressed by a lossy algorithm. Images on the other hand might be lossy-compressed. Occasionally, SuperCam will want to download an un-compressed image. Note that compression algorithms are described in the SuperCam sections of the M2020 Camera SIS.

Microphone Data Format: The microphone is capable of recording an audio signal for for maximum recording duration is of 167 seconds at 25 kHz and 41 seconds at 100 kHz, producing a maximum science data size of 8 MB uncompressed.

FSW **shall** use lossless compression for microphone acoustic data.

State-of-health Data Format for monitoring the instrument. Consists of various housekeeping parameters and temperatures for monitoring the health of the MU and BU. See Appendix D for list of values recorded in the SOH data product.

2.4 Cruise and Surface Operations

2.4.1 Data products for cruise

A Data Product consists of a file containing a set of instrument telemetry, optionally concatenated with RCE (Rover Compute Element; rover computer) supplied "ancillary" data, and an associated set of RCE "metadata". Data Products are stored in RCE nonvolatile

FLASH memory. Data Product metadata is used to select Data Products for prioritized downlink, transmission route selection, reprioritization, retransmission, and deletion. A key piece of information is the Data Product Priority, which is used to select data products for downlink. The initial Product Priority value is provided by Mission Operations (including each Instrument Operations Team) as an argument to the generating Spacecraft Command and is assigned by the RCE Flight Software at Product Creation time. After initial assignment, the Product Priority of any individual data product can be changed at any time by Mission Operations.

The RCE Flight Software provides the standard services of accepting specified sets of telemetry from instruments, then constructing, storing and managing such telemetry as Data Products.

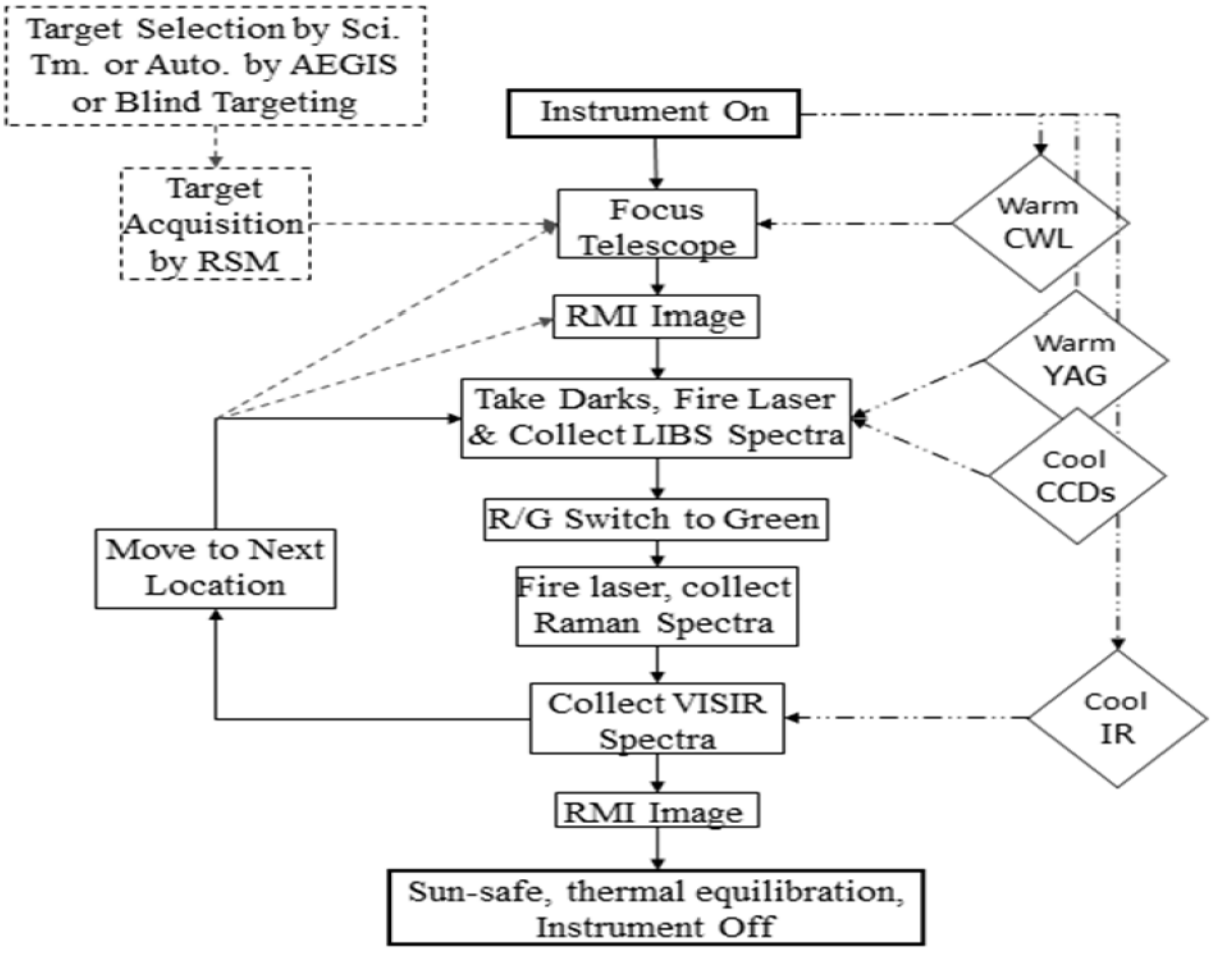


Figure 7 SuperCam typical observing sequence. CWL = Continuous Wave Laser, R/G = Red/Green (laser), YAG = Nd:YAG laser

SuperCam operational steps are as follows:

1. Instrument Power On: The instrument is powered on and boots into bank1. The instrument can be optionally rebooted into bank1 or bank2 as specified in an RCE

parameter. Boot bank 1 should be used for nominal observations. Communication between the RCE and SuperCam is established and their clocks are synchronized. State of health (SOH) data is also collected. Note that only the body unit is powered-on at this point, as the SCMU is powered-on later, by the SCBU. The mast unit must be powered on before cooling of the IR detector or warming of the lasers.

2. CCD cooldown: - Thermoelectric Cooler (TEC) power on for the IR spectrometer (if used) and/or the BU spectrometers. These must cool the detectors before data collection. The laser warmup and the TEC can start simultaneously upon power on, but more likely, the SCBU TECs are turned on first, as they need more time to cool. The SCMU can stay off while the initial SCBU TEC cooling occurs. The lasers and detectors take a few minutes to reach the desired operational temperatures. The lasers and detectors take a few minutes to reach the desired operational temperatures.
3. Laser warmup for the Continuous Wave Laser (CWL) and LIBS laser temperatures are checked and if they are below their minimum operating temperature, their warm up heater routines are started.
4. Focusing: A narrow-beam diode laser is turned on, illuminating the target with modulated monochromatic light. The focus stage is scanned around the approximate position given by the mission operations team. A photodiode receives the signal returned by the target, which is demodulated to increase the sensitivity. Then the position giving the maximum signal is computed and the focus stage is automatically brought back to this position using the autofocus algorithm. There is a primary and backup photodiode, and the one used will be identified in the FITS file.
5. RMI Imaging: The imager includes auto-exposure algorithms as part of its High Dynamic Range (HDR) software. While spectral observations and imaging will be used frequently together, there will be numerous times when RMI will be used separately, such as to provide close-up images to support robotic arm operations, or to provide images of very distant objects.
6. LIBS Data Acquisition: The laser is fired in a series of shots (typically 30) while all three BU spectrometers record spectra. The laser can operate at 3-10 Hz, with a nominal repetition rate of 3Hz. The red/green switch must be in the LIBS (red) position before this analysis. Non-laser “dark” spectra are taken either before or after (or both) the active LIBS. If autofocus was acquired using the RMI image, LIBS must be commanded to the offset focus position for optimal focusing. WARNING: Offset can ONLY be used when an autofocus has been completed. Otherwise the command will be rejected by the BU.
7. Raman Data Acquisition: The laser is fired in a rapid series of shots (up to 500) while the transmission spectrometer records spectra. These spectra may be taken more rapidly, e.g., up to 10 Hz. Non-laser “dark” spectra are taken either before or after (or both) the active Raman. The red/green switch must be in the Raman (green) position before this analysis. This analysis is less sensitive to focus than LIBS or RMI, so no focus offset is needed.
8. Passive VISIR Data Acquisition: Passive reflectance spectra are taken with the transmission spectrometer (also VIO) and the IR spectrometer. The transmission spectrometer uses a low gain setting and an exposure time of tens to hundreds of microseconds. The IR spectrometer scans through its spectral range. VISIR analysis is less sensitive to focus than LIBS or RMI so no focus offset is needed. The LIBS, Raman,

and VISIR analyses can be done in any order, but the Raman and VISIR normally benefit from the partial dust removal provided by LIBS, so usually they will be done after the LIBS.

9. Building a raster: Typically, spectra will be recorded for a number of observation points on each target. These will be done as either linescans (1x5, 1x10, 1x20 points) or grids (3x3, 4x4, etc). For each point we anticipate recording LIBS, Raman, and VISIR spectra. In most cases a separate RMI image is not needed for each point, as several raster points fit within the field of view of the RMI. For example, for a 1x5 linescan with reasonably closely spaced points, only before (point 1) and after (point 5) images are needed. However, for a 1x10 raster, usually 3 or 4 images are used to photo-document all the points. Templates for the various rasters will be set up with the proper spacing of RMI images. An RMI image is always used after the last point of a raster. The most frequent type of “analysis” is a raster. In the case of a raster the RSM moves to the next designated analysis point in the raster and the LIBS, Raman, and VISIR analyses are repeated. RMI images are taken at intermediate points in the raster if the spacing of the raster points requires it to provide continuous image coverage of the raster. Additional focuses are included every several positions.
10. Post-analysis RMI Image: Typically, an image of the laser spot will be taken subsequent to LIBS analysis, as mentioned above. The post LIBS image must be commanded to the offset position from LIBS to obtain an in focus image. WARNING: Offset focus position can ONLY be used when an autofocus has been completed. Otherwise the command will be rejected by the BU.
11. Transfer data, equilibrate laser and CCDs temperature, and power off. (The RMI can hold images, so we need to decide if they are transferred at the end of analysis or during the analysis). Otherwise it is only the controlled cooling of the lasers, controlled warming of the CCDs, and the RCE powers off the instrument.

2.4.2 Typical SuperCam Raman Spectroscopy Surface usage

Here is the flowchart showing the steps performed during Raman Spectroscopy in scan mode:

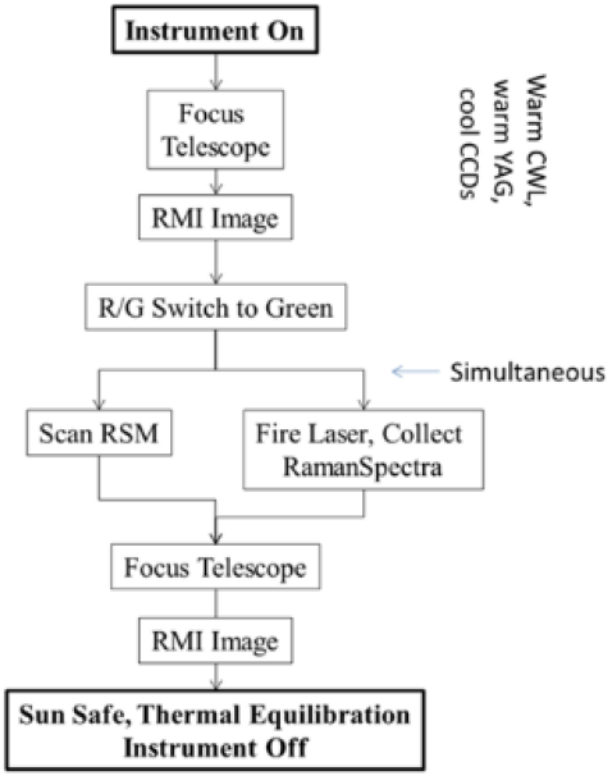


Figure 8 SuperCam Raman spectroscopy in scan mode

2.4.3 Typical SuperCam VisIR Spectroscopy Surface usage

Here is the flowchart showing the steps performed during VisIR Spectroscopy in scan mode:

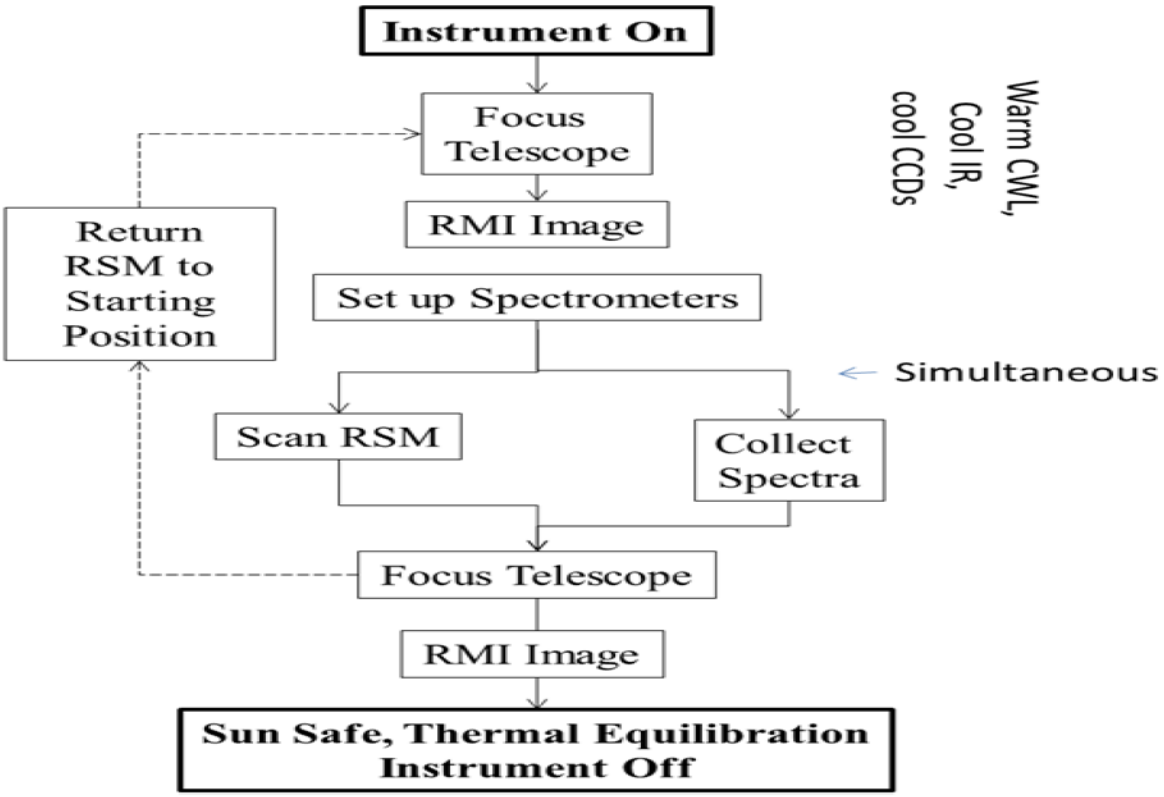


Figure 9 SuperCam VISIR spectroscopy in scan mode

3 GENERAL DATA PRODUCT OVERVIEW

3.1 Data Processing Levels

This documentation uses the "Committee on Data Management and Computation" (CODMAC) data level numbering system. The M2020 instrument EDRs referred to in this document are considered "Level 2" or "Edited Data". The EDRs are to be reconstructed from "Level 1" or "Raw Data", which are the telemetry packets within the project specific Standard Formatted Data Unit (SFDU) record. They are to be assembled into complete data products (typically spectra), but will not be radiometrically or geometrically corrected.

The PDS4 archive standard does not use CODMAC data processing level numbers. Instead, it recognizes five data processing level terms: telemetry, raw, partially processed, calibrated, and derived. Table 3-1 shows the definitions of these processing levels.

Table 3-1 PDS4 Processing levels for Instrument Experiment Data Sets

Processing Level for PDS4 Archive	Operations Data Product Name	Description
Telemetry	n/a	An encoded byte stream used to transfer data from one or more instruments to temporary storage where the raw instrument data will be extracted. PDS does not archive telemetry data.
Raw	EDR (Experiment Data Record, heritage term based on MSL mission)	Original data from an instrument. If compression, reformatting, packetization, or other translation has been applied to facilitate data transmission or storage, those processes will be reversed so that the archived data are in a PDS approved archive format. For Mars 2020, these are the original products.However for M2020, the product ID field in the EDR filenames have a variety of 3 character codes beginning with "E" to distinguish between many types of Surface EDRs
Partially Processed	EDR FDR (Fundamental Data Record) RDR (Reduced Data Record)	Data that have been processed beyond the raw stage, but which have not yet reached calibrated status. For Mars 2020 imaging, these are the decompanded and debayered EDRs, the FDRs, and a few RDRs. These do not apply to M2020 non-imaging data
Calibrated	RDR	Data converted to physical units, which makes values independent of the instrument.
Derived	RDR	Results that have been distilled from one or more calibrated data products (for example, maps, gravity or magnetic fields, or ring particle size distributions). Supplementary data, such as calibration tables or tables of viewing geometry, used to interpret observational data should also be classified as "derived" data if not easily matched to one of the other three categories.

3.2 Product Label and Header Descriptions

3.2.1 Overview of Labels

Mars 2020 SuperCam non-imaging products consist of two files: the instrument data file in FITS/ODL format, and a detached PDS4 label file in XML format. The spectrometer labels differ from the SuperCam RMI image labels in that there is no VICAR (Video Image Communication And Retrieval = JPL planetary imaging format) label. The instrument data file is in a dual FITS/ODL format. It consists of two major parts: the ODL label and the binary FITS data area. Thus, each file actually has two labels: the ODL label, which is used during operations, and the detached PDS4 label, which is used for archiving the data at PDS. Both labels contain the same semantic content and can be used interchangeably. The ODL label is used by many science teams. Both are included for maximum compatibility, so that operations tools can still be used on archive data. Additional information on the software tools which make these files can be found in Section 7.

The primary label from the archive perspective is the detached PDS4 label. This is a separate file with the same base name as the image file, with a ".XML" extension. The detached label references the EDR/RDR filename via an XML attribute. This label is fully compliant with all PDS4 archive standards.

3.2.2 ODL Labels

As implied in the previous section, the EDRs described in this document have an attached ODL label and a detached PDS4 label. Each institution is responsible for converting PDS-formatted products to be compatible with their own software systems (such as VICAR, IDL, ISIS, etc.).

The attached ODL label starts with the keyword assignment:

ODL_VERSION_ID = ODL3

An ODL label is object-oriented and describes the objects in the data file. The ODL label contains keywords for product identification, along with the data object definition containing descriptive information needed to interpret or process the data in the file.

ODL label statements have the form of "keyword = value". Each label statement is terminated with a carriage return character (ASCII 13) and a line feed character (ASCII 10) sequence to allow the label to be read by many operating systems. Pointer statements with the following format are used to indicate the location of data objects in the file:

^object = location

where the carat character (^, also called a pointer) is followed by the name of the specific data object. The location is the 1-based starting record number for the data object within the

file. Alternatively, it could be the 1-based byte location within the file if it includes a <bytes> unit tag. Pointers are used to define the locations of the binary instrument data itself (^IMAGE for image data), the VICAR header in the case of images (^IMAGE_HEADER), and the non-imaging binary data (SuperCam only, e.g. ^SCAM_SOH_DPO).

3.2.2.1 PDS Data Dictionaries

The PDS4 label includes many XML classes and attributes defined in the PDS Common Data Dictionary, the PDS Mars 2020 Mission Dictionary, and other PDS data dictionaries. The dictionaries are represented by XML schema, which are listed at the top of every PDS4 label. Current and older versions of all PDS dictionaries are available at https://pds.nasa.gov/datastandards/dictionaries/.

3.2.2.2 Keyword Length Limits

All ODL keywords are limited to 31 characters in length. Therefore, software that reads M2020 ODL labels must be able to ingest keywords up to 31 characters in length.

For image RDR-producing institutions wishing to accommodate the VICAR mapping of ODL keywords that use a <*unit*> tag after the value, such keywords must be limited to 25 characters in length to accommodate the "__UNIT" suffix. Otherwise, those keywords will not be transcoded from the ODL label into a VICAR label.

3.2.2.3 Data Type Restrictions

In order to accommodate VICAR dual-labeled files, 16-bit data must be stored as signed data. Unsigned 16-bit data are not supported. 12-bit unsigned data from the cameras are stored in a 16-bit signed value. 8-bit data are unsigned.

3.2.2.4 Interpretation of N/A, UNK, and NULL

During the completion of data product ODL labels, one or more values may not be available for some set of required data elements. In this case the literals "N/A", "UNK", and "NULL" are used, each of which is appropriate under different circumstances. As a note, if any one of these three symbolic literals are used in place of a keyword value that is normally followed by a Unit Tag(s) (e.g., "<*value*>"), the Unit Tag(s) is removed from the label.

- "N/A" ("Not Applicable") indicates that the values within the domain of this data element are not applicable in this instance. For example, a label describing NAIF SPK kernels would contain the statement:

  INSTRUMENT_ID = "N/A"

  because this data set is not associated with a particular instrument.

"N/A" may be used as needed for data elements of any type (e.g., text, date, numeric, etc.).

- "UNK" ("Unknown") indicates that the value for the data element is not known and never will be. For example, in a data set comprising a series of images, each taken with a different filter, one of the labels might contain the statement:

  FILTER_NAME = "UNK"

  if the observing log recording the filter name was lost or destroyed and the name of the filter is not otherwise recoverable.

  "UNK" may be used as needed for data elements of any type.

- "NULL" is used to flag values that are *temporarily* unknown. It indicates that the data preparer recognizes that a specific value should be applied, but that the true value was not readily available. "NULL" is a placeholder. For example, the statement:

  DATA_SET_RELEASE_DATE = "NULL"

  might be used in a data set catalog file during the development and review process to indicate that the release date has not yet been determined.

  "NULL" may be used as needed for data elements of any type.

  Note that all "NULL" indicators should be replaced by their actual values prior to final archiving of the associated data.

Unlike earlier missions, some effort has been expended to reduce the number of UNK, N/A, and NULL values appearing in the label, since they can cause difficulties with the Velocity technology used to create PDS4 labels. Therefore, while these values are possible, they should be rare.

3.2.2.5 ODL Label Constructs "Class", "Object" and "Group"

For the EDRs and RDRs described in this document, the ODL label includes the following constructs:

- *Class* - The Class construct resides in a ODL label as a grouping of keywords that are thematically tied together. Classes are usually preceded by a label comment, although it is not required. ODL label comments are character strings bounded by "/* */" characters.

  In the M2020 non-imaging ODL label a Class of keywords will be preceded by a comment string as follows:

  ```
  /*  comment string  */
  keyword     =   keyword value
  keyword     =   keyword value
  ```

- *Object* - The Object construct is a set of standard keywords used for a particular data product. In the M2020 non-imaging ODL label an Object's set of keywords is specified as follows:

```
OBJECT          =  Object identifier
    keyword     =  keyword value
    keyword     =  keyword value
END  OBJECT     =  Object identifier
```

- *Group* - The Group construct is a grouping of keywords that are not components of a larger Object. Group keywords may reside in more than one Group within the label. In the M2020 non-imaging ODL label, a Group's set of keywords is specified as follows:

```
GROUP           =  Group identifier
    keyword     =  keyword value
    keyword     =  keyword value
END_GROUP       =  Group identifier
```

3.2.3 PDS4 Label

The PDS4 label is a separate file with the same base name and an extension of ".xml". It is in XML format whose content is controlled by the PDS Information Model and PDS core, discipline, and mission data dictionaries. The PDS4 label contains the same semantic information as the VICAR label, although the format is quite different. For FITS files (ending in ".fits"), the label removes the fits extension and replaces it with ".xml". Appendix B contains tables mapping between ODL keywords and their corresponding PDS4 construct.

3.2.4 SuperCam FITS Files

SuperCam stores EDR binary data in FITS format. The metadata is stored in HDU (Header Data Unit) structures within the FITS file. It consists of instrument State of Health (SOH), housekeeping, autofocus, and similar information, in binary format as received in the telemetry stream. A copy of the ODL label is also stored in an HDU.

These FITS files are included in the PDS archive.

3.3 Binary Data Storage Conventions

EDR data products for M2020 SuperCam spectral data are stored as binary data. For the non-image EDRs, the data formats include rescaled 8-bit integers stored in an unsigned byte, as well as 10-bit to 13-bit integers stored in signed 16-bit integers. The spectrum EDR cases vary between unsigned 16-bit and unsigned 32-bit integers. The PDS labels are stored as ASCII text.

3.3.1 Bit and Byte Ordering

The ordering of bits and bytes is only significant for instrument (image and spectra) and binary header data; all other labeling information is in ASCII.

For non-byte instrument data, which includes 8-bit unsigned shorts, 16-bit signed shorts, 32-bit signed ints, and 32- and 64-bit IEEE floating-point numbers, the data may be stored in either Most Significant Byte first ("big-endian", as used by e.g. Sun computers and Java), or Least Significant Byte first ("little-endian", as used by e.g. Linux and Windows computers). In an EDR/RDR product, the instrument data can have only one ordering, but it is dependent on the host platform where the data was processed. Binary header data, applicable to SuperCam, can have a different ordering than the instrument data. This follows the PDS/ODL file format conventions.

4 EDR Product Specification

M2020 instrument EDRs described in this document will be generated by JPL's Multimission Instrument Processing Laboratory (MIPL) under the IDS subsystem of the M2020 Ground Data System (GDS) element. Other RDRs described in this document will be generated by the SuperCam Science Team.

The EDR consists of unprocessed experiment data stored in the FITS binary format . The EDR structures defined in this document vary depending on instrument, with all containing attached metadata labels. A SuperCam PDS User Guide is included as part of the SuperCam PDS archive as an introduction to the data produced by SuperCam. This document is a quick reference on how to find the SuperCam data and identify the technique from the file name as well as how access the data.

For M2020, with the exception noted in Section 1.3, the spectometer and microphone EDRs are the fundamental non-imaging instrument data archive product. They will be generated as "raw" uncalibrated data within an automated pipeline process managed by IDS under GDS at JPL. The size of the EDRs varies with select instrument modes. The size of an image EDR data product is approximately 8 MB, while the size of a spectrum EDR is approximately 700 KB.

EDRs are versioned on M2020. When updated telemetery is received a new version of the EDR will be generated. Updated telemetry is possible when missing packets are retransmitted and/or telemetry is reflowed through the Ground Data System (GDS).

4.1 EDR General Processing

The EDR processing begins with the reconstruction of packetized telemetry data resident on the Telemetry Data System (TDS) by the Mission Data Processing and Control Subsystem (MPCS) into a binary ".dat" data product and associated ".emd" Earth metadata file. The data product and metadata are written by the M2020 Mission data Processing Control System (MPCS) to the Operations Cloud Store (OCS). Messages are generated on a Java Message Server (JMS) bus, where they are ingested by MIPL's EDR generation software "m2020edrgen" and processed using NASA navigational code (SPICE kernels provided by NAIF). The EDR will be generated after the notification describing the OCS location of the respective the binary data product and associated Earth metadata file has been received by the IDS pipeline system. The data flow from instrument to IDS is illustrated in Figure . The data flow within the IDS pipeline to EDR generation is elaborated subsequently in this section.

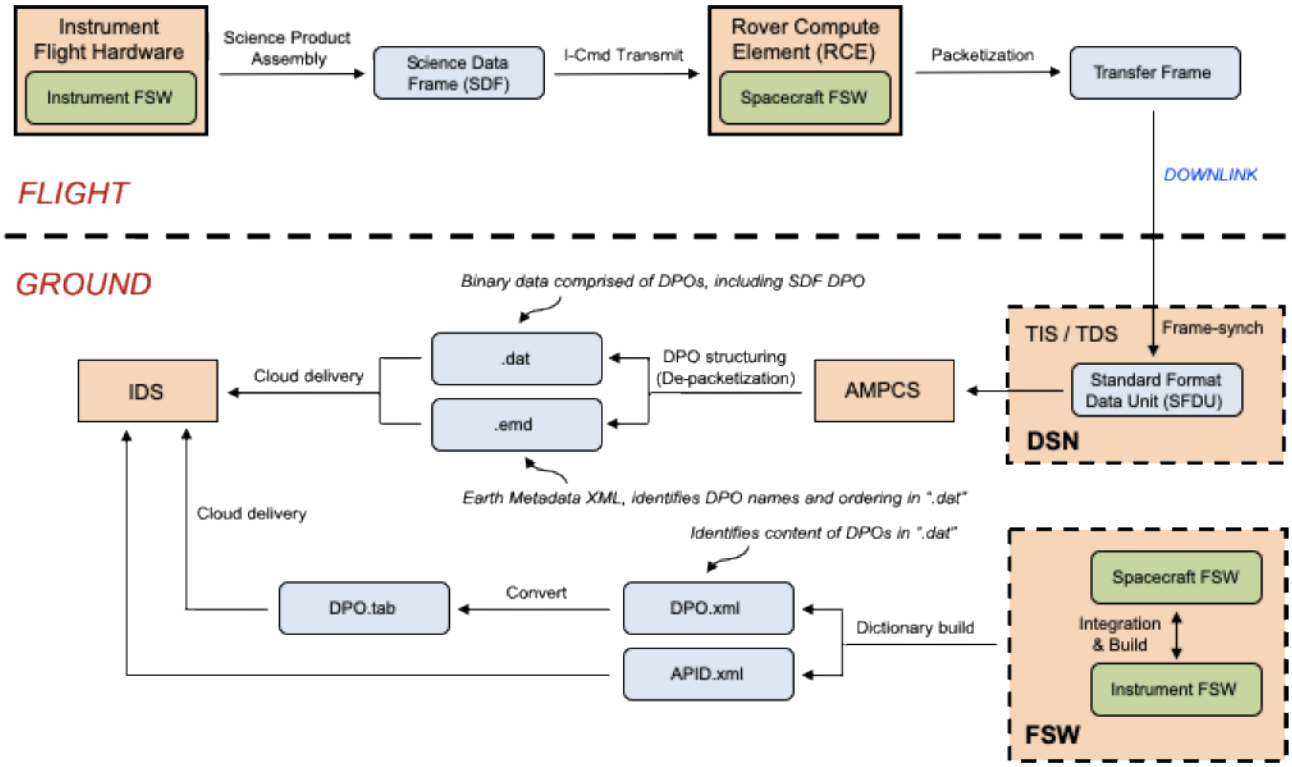


Figure 10 M2020 instrument data flow to IDS. Acronyms include: Instrument Data System (IDS) at JPL where the EDRgen software resides; Deep Space Network (DSN), the system of radio telescopes which transmit commands to and receive data from deep space missions; Flight Software (FSW), "firmware" code on board the spacecraft and instruments which controls operation of the hardware; Data Product Object (DPO), a data set produced by a single instrument command as transmitted from the rover; Instrument Command (I-cmd), command to the instrument FSW, as distinguished from a Spacecraft Command (S-cmd); Application Process Identifier (APID), a unique integer which identifies the type of telemetry being transmitted from a spacecraft. The diagram depicts data delivery from the spacecraft during flight (FLIGHT to DSN to IDS) and ithe incorporation of the FSW translation tables which enable conversion of flight instrument DPOs to scientific data files (EDR).

In all EDR cases, missing packets will be identified and reported for retransmission to the ground as “partial datasets”. Prior to retransmission, the missing EDR data will be filled with zeros. The EDR data will be reprocessed only after all “partial datasets” are retransmitted and received on the ground. In these cases, the EDR version will be incremented so as not to overwrite any previous EDR versions.

4.2 EDR Product Types

Descriptions for the various EDR product types are provided in this section. They are broken down into two groupings: a) Archived Science Products and b) Engineering Products. Refer to Table 4-2 for a mapping between the source M2020 instrument and the EDR product type.

Table 4-1 SuperCam Product ID (PID)

DPO	FITS File Tag	EDR Product ID	Notes	Archived?
ScamFault	SFAU	ESF		
ScamBuMemDump	SBMD	EBMD		
ScamMuMemDump	SMMD	EMMD		
ScamDebugDump	SDDM	EDD		
ScamRmiImage	RMI0	EDR		Y
ScamRmiImageIcer	RMI0	EDR		Y
ScamRmiImageJpg	RMI0	EDR		Y
ScamRmiZstackImage	RMI0	ECZ		Y
ScamRmiZstackImageIcer	RMI0	ECZ		Y
ScamRmiZstackImageJpg	RMI0	ECZ		Y
ScamSoh	SOH	ESH		
ScamUtilTest	SOH or other	ESU	FITS tag may be from SOH table entries below	
ScamBuSohPowerOn	SBP1	EBP1		
ScamBuSohPowerOff	SBP0	EBP0		
ScamMuSohPowerOn	SMP1	EMP1		
ScamMuSohPowerOff	SMP0	EMP0		
ScamBuSohThermalOn	SBT1	EBT1		
ScamBuSohThermalOff	SBT0	EBT0		
ScamMuSohThermalOn	SMT1	EMT1		
ScamMuSohThermalOff	SMT0	EMT0		
ScamSohSunSafe	SSSF	ESS		
ScamPoint	-	ESP	Not parsed by FITSGEN	

ScamScidata	Varies	Varies	See Table 4-2	Y

Table 4-2 SuperCam ScamScidata Science Products

– "non-nominal" data are only obtained in contingency situations to diagnose anomalies. This data type is not routinely expected.

Instrument Mode	Actives	Darks	Type	FITS Filename Tag	EDR Product ID
LIBS	All	All		LRD0	EL0
	All	Stats		LRD1	EL1
	Stats	Stats		LRD2	EL2
	All	None		LRD3	EL3
	Stats	None		LRD4	EL4
	None	All		LRD5	EL5
	None	Stats		LRD6	EL6
2D	-	-		LR2D	EL9
Non-nominal	-	-		LRDS	ELS
RAMAN	All	All		RRD0	ER0
	All	Stats		RRD1	ER1
	Stats	Stats		RRD2	ER2
	All	None		RRD3	ER3
	Stats	None		RRD4	ER4
	None	All		RRD5	ER5
	None	Stats		RRD6	ER6
2D	-	-		RR2D	ER9
Non-nominal	-	-		RRDS	ERS
TRLS	All	All		TRD0	ET0
	All	Stats		TRD1	ET1
	Stats	Stats		TRD2	ET2
	All	None		TRD3	ET3
	Stats	None		TRD4	ET4
	None	All		TRD5	ET5
	None	Stats		TRD6	ET6
2D	-	-		TR2D	ET9
Non-nominal	-	-		TRDS	ETS
VIS	-	All		PRD0	EP0
	-	Stats		PRD1	EP1
	-	IRSpec		PRD2	EP2
		Double VIS		PRD3	EP3
		IRScan		PRD4	EP4
Passive 2D	-	-		PR2D	EP9
Non-nominal	-	-		PRDS	EPS
Microphone	-	-	Libs-sync	ARD0	EA0
	-	-	Libs-cont	ARD1	EA1
	-	-	Standalone	ARD2	EA2
Autofocus	-	-	Rmi-AF	AFD0	EF0
	-	-	CWL-AF	AFD1	EF1
Non-nominal SOH-only	-	-	-	SOH0	ESH

Non-nominal RMI-cmds-only	-	-	-	RMIC	EIC

Table 4-3 Mars 2020 SuperCamData Products to be Archived

Product Name	PDS Processing Level (raw, pp*, calibrated, derived)	PDS4 bundle:collection, assuming prefix urn:nasa:pds:	Product Description	PDS4 Data Type (e.g. Array_2D_Image, Array_3D_Image, Table_Character, Table_Delimited, Table_Binary)	Estimated Single Product Size (Data + Label)	Product Aggregation (e.g. how many per sol)	Data Producer	Provider to PDS	PDS Curator (GEO, CIS, ATM)
SuperCam									
SuperCam LIBS EDR	raw	mars2020_supercam: data_raw_spectra	Raw LIBS spectra	Table_Binary	250kB – 10MB	About 11 per sol	IDS	SuperCam	GEO
SuperCam VIS EDR	raw	mars2020_supercam: data_raw_spectra	Raw VISIR spectra	Table_Binary	100kB – 1MB	About 16 per sol	IDS	SuperCam	GEO
SuperCam IR EDR	raw	mars2020_supercam: data_raw_spectra	Raw IR spectra	Table_Binary	150-200 kB	About 16 per sol	IDS	SuperCam	GEO
SuperCam Raman EDR	raw	mars2020_supercam: data_raw_spectra	Raw Raman spectra	Table_Binary	75kB – 1MB	About 8 per sol	IDS	SuperCam	GEO
SuperCam TRLS EDR	raw	mars2020_supercam: data_raw_spectra	Raw Time-resolved luminescence spectra	Table_Binary	75kB – 1MB	About 2 per sol	IDS	SuperCam	GEO
SuperCam RMI EDR	raw	mars2020_imgops: data_rmi_imgops mars2020_supercam: data_raw_rmi (secondary members**)	RMI raw images	Array_2D_Image	~8MB	About 10 per sol	IDS	SuperCam	GEO
SuperCam MIC EDR	raw	mars2020_supercam: data_raw_audio	Raw acoustic recording	Table_Binary	1-8MB	About 7 per sol	IDS	SuperCam	GEO
SuperCam LIBS RDR	calibrated	mars2020_supercam: calibration_supercam	LIBS Spectral Library	Table_Binary		n/a	SuperCam	SuperCam	GEO

Product Name	PDS Processing Level (raw, pp*, calibrated, derived)	PDS4 bundle:collection, assuming prefix urn:nasa:pds:	Product Description	PDS4 Data Type (e.g. Array_2D_Image, Array_3D_Image, Table_Character, Table_Delimited, Table_Binary)	Estimated Single Product Size (Data + Label)	Product Aggregation (e.g. how many per sol)	Data Producer	Provider to PDS	PDS Curator (GEO, CIS, ATM)
SuperCam LIBS RDR	calibrated	mars2020_supercam: data_calibrated_spectra	LIBS spectra calibrated with wavelength calibration	Table_Binary	1-12 MB	About 11 per sol	SuperCam	SuperCam	GEO
SuperCam VIS RDR	calibrated	mars2020_supercam: data_calibrated_spectra	VIS spectra calibrated	Table_Binary	<1 MB	About 16 per sol	SuperCam	SuperCam	GEO
SuperCam IR RDR	calibrated	mars2020_supercam: data_calibrated_spectra	IR spectra calibrated	Table_Binary	200-600 kB	About 16 per sol	SuperCam	SuperCam	GEO
SuperCam Raman RDR	calibrated	mars2020_supercam: data_calibrated_spectra	Raman spectra calibrated with wavelength calibration	Table_Binary	<1 MB	About 8 per sol	SuperCam	SuperCam	GEO
SuperCam TRLS RDR	calibrated	mars2020_supercam: data_calibrated_spectra	TRLS spectra calibrated with wavelength calibration	Table_Binary	<1 MB	About 2 per sol	SuperCam	SuperCam	GEO
SuperCam RMI RDR	calibrated	mars2020_supercam: data_radcal_rmi	RMI Partially radiometrically corrected images	Array_2D_Image	~ 16 MB	About 10 per sol	SuperCam	SuperCam	GEO
SuperCam MIC RDR	calibrated	mars2020_supercam: data_calibrated_audio	MIC time series calibrated in Pascal	Table_Binary	1-8MB	About 2 per sol	SuperCam	SuperCam	GEO
SuperCam MIC RDR	calibrated	mars2020_supercam: data_calibrated_audio	MP4 audio recordings supplementary to data_calibrated_audio products (WAV files)	Encoded_Audio	8MB	About 2 per sol	SuperCam	SuperCam	GEO
SuperCam LIBS RDR	derived	mars2020_supercam: data_derived_spectra	LIBS element abundances	Table_Delimited	10MB (final)	1 product per mission, accumulating with each delivery	SuperCam	SuperCam	GEO

Product Name	PDS Processing Level (raw, pp*, calibrated, derived)	PDS4 bundle:collection, assuming prefix urn:nasa:pds:	Product Description	PDS4 Data Type (e.g. Array_2D_Image, Array_3D_Image, Table_Character, Table_Delimited, Table_Binary)	Estimated Single Product Size (Data + Label)	Product Aggregation (e.g. how many per sol)	Data Producer	Provider to PDS	PDS Curator (GEO, CIS, ATM)
SuperCam VIS RDR	derived	mars2020_supercam: data_derived_spectra	Ratio of VIS data to the SCCT_White cal target taken under similar conditions	Table_Delimited	5MB (final)	1 product per mission, accumulating with each delivery	SuperCam	SuperCam	GEO
SuperCam IR RDR	derived	mars2020_supercam: data_derived_spectra	Ratio of IR data to the SCCT_White cal target taken under similar conditions	Table_Delimited	5MB (final)	1 product per mission, accumulating with each delivery	SuperCam	SuperCam	GEO
SuperCam Raman RDR	derived	mars2020_supercam: data_derived_spectra	Table of features with columns for likely mineral identifications and possible mineral identifications	Table_Delimited	5MB (final)	1 product per mission, accumulating with each delivery	SuperCam	SuperCam	GEO
SuperCam TRLS RDR	derived	mars2020_supercam: data_derived_spectra	Table of features with columns for likely mineral identifications and possible mineral identifications	Table_Delimited	5MB (final)	1 product per mission, accumulating with each delivery	SuperCam	SuperCam	GEO
SuperCam RMI Mosaic RDR	derived	mars2020_supercam: data_special_rmi	RMI mosaics from individual RMI RDR products, position of observations noted	Array_2D_Image	80 MB	About 2 per sol	SuperCam	SuperCam	GEO
SuperCam RMI Mastcam Mosaic RDR	derived	mars2020_supercam: data_special_rmi	Mastcam images with the outlines of the RMI image overlaid	Array_2D_Image	ZCAM size	About 2 per sol	SuperCam	SuperCam	GEO
SuperCam Master target	derived	mars2020_supercam: data_observation_log	SuperCam master spreadsheet with information about all targets.	Table_Delimited	25MB	1 product per mission, accumulating with each delivery	SuperCam	SuperCam	GEO

Product Name	PDS Processing Level (raw, pp*, calibrated, derived)	PDS4 bundle:collection, assuming prefix urn:nasa:pds:	Product Description	PDS4 Data Type (e.g. Array_2D_Image, Array_3D_Image, Table_Character, Table_Delimited, Table_Binary)	Estimated Single Product Size (Data + Label)	Product Aggregation (e.g. how many per sol)	Data Producer	Provider to PDS	PDS Curator (GEO, CIS, ATM)
SuperCam EDR RDR SIS	document	mars2020_supercam: document	Description of EDR and RDR contents and format	Document	n/a	n/a	IDS / SuperCam	SuperCam	GEO
SuperCam Bundle SIS	document	mars2020_supercam: document	Description of mars2020_supercam bundle organization	Document	n/a	n/a	SuperCam	SuperCam	GEO

Notes

* pp = "partially processed". This processing level is for data that have undergone some processing beyond the raw product, but that are not yet calibrated.

** "Secondary members" indicates that some or all members of the collection are primary members of another collection. Every product is a primary member of exactly one collection, and is physically resident at the location of that collection. A product may be a secondary member of another collection, meaning its Logical Identifier is listed in that collection's inventory, but it is not physically copied there.

4.2.1 Science (Spectroscopy/Microphone/Autofocus) EDRs (Archived)

4.2.1.1 LIBS EDR Product ID (PIDs)

The LIBS EDRs are comprised of one or more SuperCam LIBS spectroscopy intensity (spectra) values, or digital numbers (DNs), stored as an array of 16-bit or 32-bit unsigned integers. Applicable to the SuperCam LIBS instrument.

4.2.1.1.1 EL0

This EDR includes all LIBS actives and darks. The FITSGEN tag is LRD0.

4.2.1.1.2 EL1

This EDR includes all LIBS actives and stats for darks. The FITSGEN tag is LRD1.

4.2.1.1.3 EL2

This EDR includes stats only for LIBS actives and darks. The FITSGEN tag is LDR2.

4.2.1.1.4 EL3

This EDR includes all LIBS actives and no darks. The FITSGEN tag is LRD3.

4.2.1.1.5 EL4

This EDR includes stats for LIBS actives and no darks. The FITSGEN tag is LRD4.

4.2.1.1.6 EL5

This EDR includes no LIBS actives and all LIBS darks. The FITSGEN tag is LRD5.

4.2.1.1.7 EL6

This EDR includes no LIBS actives and LIBS dark stats. The FITSGEN tag is LRD6.

4.2.1.1.8 EL9

This EDR is for 2D LIBS spectra. The FITSGEN tag is LR2D.

4.2.1.1.9 ELS

This EDR is for non-nominal LIBS spectra. The FITSGEN tag is LRDS.

4.2.1.2 RAMAN Spectrum EDRs PIDs

The RAMAN Spectrum EDR is comprised of one or more SuperCam Raman spectroscopy intensity (spectra) values, or digital numbers (DNs), stored as an array of 16-bit or 32-bit unsigned integers. Applicable to the SuperCam RAMAN instrument.

4.2.1.2.1 ER0

This EDR includes all Raman actives and darks. The FITSGEN tag is RRD0.

4.2.1.2.2 ER1

This EDR includes all Raman actives and stats for Raman darks. The FITSGEN tag is RRD1.

4.2.1.2.3 ER2

This EDR includes stats only for Raman actives and darks. The FITSGEN tag is RDR2.

4.2.1.2.4 ER3

This EDR includes all Raman actives and no Raman darks. The FITSGEN tag is RRD3.

4.2.1.2.5 ER4

This EDR includes stats for Raman actives and no Raman darks. The FITSGEN tag is RRD4

4.2.1.2.6 ER5

This EDR includes no Raman actives and all Raman darks. The FITSGEN tag is RRD5.

4.2.1.2.7 ER6

This EDR includes no Raman actives and Raman dark stats. The FITSGEN tag is RRD6.

4.2.1.2.8 ER9

This EDR is for 2D Raman spectra. The FITSGEN tag is RR2D.

4.2.1.2.9 ERS

This EDR is for non-nominal Raman spectra. The FITSGEN tag is RRDS.

4.2.1.3 TRLS Spectrum EDR PIDs

The TRLS spectrum EDR is comprised of one or more SuperCam TRLS spectroscopy intensity (spectra) values, or digital numbers (DNs), stored as an array of 16-bit or 32-bit unsigned integers.

4.2.1.3.1 ET0

This EDR includes all TRLS actives and darks. The FITSGEN tag is TRD0.

4.2.1.3.2 ET1

This EDR includes all TRLS actives and stats for TRLS darks. The FITSGEN tag is TRD1.

4.2.1.3.3 ET2

This EDR includes stats only for Raman actives and darks. The FITSGEN tag is TDR2.

4.2.1.3.4 ET3

This EDR includes all Raman actives and no Raman darks. The FITSGEN tag is TRD3.

4.2.1.3.5 ET4

This EDR includes stats for TRLS actives and no TRLS darks. The FITSGEN tag is TRD4.

4.2.1.3.6 ET5

This EDR includes no TRLS actives and all TRLS darks. The FITSGEN tag is TRD5.

4.2.1.3.7 ET6

This EDR includes no TRLS actives and TRLS dark stats. The FITSGEN tag is TRD6.

4.2.1.3.8 ET9

This EDR is for TRLS 2D spectra. The FITSGEN tag is TR2D.

4.2.1.3.9 ETS

This EDR is for non-nominal TRLS spectra. The FITSGEN tag is TRDS.

4.2.1.4 VIS Spectra EDR PIDs

The VIS spectrum EDR is comprised of one or more SuperCam VIS or IR spectroscopy intensity (spectra) values, or digital numbers (DNs), stored as an array of 16-bit or 32-bit unsigned integers.

4.2.1.4.1 EP0

This EDR includes all VIS darks. The FITSGEN tag is PRD0.

4.2.1.4.2 EP1

This EDR includes VIS dark stats. The FITSGEN tag is PRD1.

4.2.1.4.3 EP2

This EDR includes IRSpec darks and spectra. The FITSGEN tag is PRD2.

4.2.1.4.4 EP9

This EDR is for 2D VIS spectra. The FITSGEN tag is PR2D.

4.2.1.4.5 EPS

This EDR is for off-nominal VIS spectra. The FITSGEN tag is PRD9.

4.2.1.5 Microphone EDR PIDs

The microphone EDRs are comprised of SuperCam microphone recordings as digital numbers (DNs), stored as an array of 16-bit or 32-bit unsigned integers. The microphone data can be configured as PULSED (LIBS-sync), the microphone is turned on, LIBS is performed and the MIC “listens” around each individual laser shot, Continuous (LIBS-cont.) the microphone is turned on and LIBS is performed (a burst of shots) while “listening in” with the microphone, or Standalone (Microphone only, no LIBS).

4.2.1.5.1 EAO

This EDR consists of LIB-sync microphone data. The FITSGEN tag is ARD0.

4.2.1.5.2 EA1

This EDR consists of LIBS-cont microphone data. The FITSGEN tag is ARD1.

4.2.1.5.3 EA2

This EDR consists of Standalone microphone data. The FITSGEN tag is ARD2.

4.2.1.6 Autofocus EDR PIDs

4.2.1.6.1 EF0

This EDR is for RMI autofocus data. The FITSGEN tag is AFD0.

4.2.1.6.2 EF1

This EDR is for CWL autofocus data. The FITSGEN tag is AFD1.

4.2.2 Engineering EDRs

4.2.2.1 "State-of-Health" (SOH) EDR PIDs (Archived)

Supercam SOH EDR FITS file will have 2 binary tables that contain all Mast Unit and Body Unit SOHs, both raw (unsigned integers, 16 or 32 bits depending on the SOH) and when appropriate, Physical Units (mostly floating points). The format of each SOH is specified in a corresponding header. All SuperCam EDR's will also contain the 2 SOH binary tables with SOH associated with the data in the specific EDR.

There are 11 Variants of the SuperCam Soh EDR's described below.

4.2.2.1.1 ESH

This SOH-only EDR is a special case in which non-science I-cmds were issued and stored in the ScamScidata DP (Cruise, non-nominal surface). The FITSGEN tag is SDH0.

4.2.2.1.2 EIC

This RMI-cmds-only EDR is a special case in which a non-imaging DP contains the RMI generic buffer (Cruise, non-nominal surface). The FITSGEN tag is RMIC.

4.2.2.1.3 EBP1

This EDR reports information from ScamBuSohPowerOn. The FITSGEN tag is SBP1.

4.2.2.1.4 EBP0

This EDR reports information from ScamBuSohPowerOff. The FITSGEN tag is SBP0.

4.2.2.1.5 EMP1

This EDR reports information from ScamMuSohPowerOn. The FITSGEN tag is SMP1.

4.2.2.1.6 EMP0

This EDR reports information from ScamMuSohPowerOff. The FITSGEN tag is SMP0.

4.2.2.1.7 EBT1

This EDR reports information from ScamSohBuThermalOn. The FITSGEN tag is SBT1.

4.2.2.1.8 EBT0

This EDR reports information from ScamSohBuThermalOff. The FITSGEN tag is SBT0.

4.2.2.1.9 EMT1

This EDR reports information from ScamSohMuThermalOn. The FITSGEN tag is SMT1.

4.2.2.1.10 EMT0

This EDR reports information from ScamSohMuThermalOff. The FITSGEN tag is SMT0.

4.2.2.1.11 ESS

This EDR reports information from ScamSohSunSafe. The FITSGen tag is SSSF.

4.2.2.2 Miscellaneous Engineering EDR PIDs

4.2.2.2.1 ESF

This EDR reports information from ScamFault. The FITSGEN tag is SFAU.

4.2.2.2.2 EBMD

This EDR reports information from ScamBuMemDump. The FITSGEN tag is SBMD.

4.2.2.2.3 EMMD

This EDR reports information from ScamMuMemDump. The FITSGEN tag is SMMD.

4.2.2.2.4 EDD

This EDR reports information from ScamDebugDump. The FITSGEN tag is SDDM.

4.2.2.2.5 ESU

This EDR reports information from ScamUtilTest. The FITSGEN tag will be one of the SOH types.

4.2.2.2.6 ESP

This EDR reports data from ScamPoint. It cannot yet be parsed by FITSGEN.

4.3 EDR Product Format

All of the differing SuperCam non-imaging EDR science product types are produced by varying parameter values within the same DPO: ScamSciData. Thus, their formats are very similar, although files of differing types will contain data from multiple spectrometers.

The EDR will be formatted according to this SIS, following the general terms of labeling and bit ordering previously discussed in Sections 3.2 and 3.3, respectively. This section details the specifics of a variety of formats across all products.products The various EDR formats are listed in Table 4-1 and discussed subsequently in this section. Note that this section is technical and likely not of interest to most science users.

4.3.1 Cruise

Currently there are 11 SuperCam cruise sequences that produce a total of 16 ScamSciData data products and 2 RMI data products.

SCAMSCIDATA
The format of the ScamSciData products for cruise will be:
4 byte implicit data size
4 byte instrument transfer frame header
4 byte size of science data
56 byte generic data buffer header (described below)
(??)bytes the science data of variable size
(??) bytes the generic data buffer trailer
4 byte instrument transfer frame checksum

To identify the data packet IDS should read the ScamSciData file down to the generic data buffer header (table 1 below). This header has a variable for the XMIT_data_id. For cruise this is a unique number that corresponds to a known data type. The lookup table for marker is described in table 2 below).

Table 4-4 ScamSciData File Description

Field	Num Bytes	Description
Packet header marker	2	0xEBAA – Start of packet header marker.
RCE time sync	4	Last time sync value received from RCE (Rover Compute Element)
Data definition version	2	Version of the instrument command dictionary. Defines specifics of each command parameter, and the structure of individual data products like SOH.
NV xmit buffer count	4	Non-volatile count of the number of times the generic buffer has been transmitted to the RCE. Acts as a unique identifier for the buffer.
BU software version	4	Unique identifier for the software version, is a unix timestamp of when the software was built.
CNDH firmware version	4	Unique identifier for the CNDH firmware version, is a unix timestamp of when the firmware was built.
BU hardware identifier	4	Hardware identifier for a specific BU CNDH board. Used to know which SOH engineering unit conversion factors to apply to the data.
SE firmware version	4	Unique identifier for the CNDH firmware version, is a unix timestamp of when the firmware was built.
SE hardware identifier	4	Hardware identifier for a specific SE board. Used to know which SOH engineering unit conversion factors to apply to the data.
MU version identifier	4	Version of the MU, response from the MU send version command.
MU hardware identifier	4	Hardware identifier for a specific MU board. Used to know which housekeeping engineering unit conversion factors to apply to the data.
Xmit data ID	4	The data_id parameter sent with the XMIT_DATA command
Reserved	4	Not used in the current version. Filled with 0's.
Number of sections	4	Number of command or data sections contained in this buffer. Does not include the header section or the trailer section.
Generic data buffer size	4	The number of bytes of the entire generic buffer. Includes commands, data, header, and trailer. Every byte from the packet header marker through the Fletcher checksum.

4.4 EDR Product Structure

4.4.1 FITS

The main file format for EDR's (aka. XDRs) is FITS (Flexible Image Transport System). FITS is a flexible, portable, file standard that allows to structure data in an efficient way. It is widely used in astronomy and there exist a number of tools that allow to read their content. A FITS file consists of a number of partitions (called HDU for Header & Data Unit), each one divided into a "header" and a "data" section. The header is made of several ASCII records that contain a key (8-digit, alphanumeric), a value (numeric, literal or Boolean) and a comment section. The data section is very flexible: it can contain ASCII TABLES, BINARY TABLE and IMAGES. For a thorough description of the FITS format, see https://fits.gsfc.nasa.gov.

4.4.2 SuperCam EDR FITS Files (G6.4+)

The EDR FITS structure will depend on which kind of data it contains. The following table describes the content of each HDU. HDU 0 to 4 consist only of metadata and is common to all file types. Keeping this metadata allows to keep every EDR independent. This means that the basic data processing for any analysis can be done starting from the EDR FITS file only.

The data section of HDU 1 and 2 will be under the form of an ASCII table. Important information from each of these data will be converted into a header key of that HDU for rapid access. These header objects contain information about the format of the table data, such as the number of records (rows) and number of fields (columns) as well as the data type of the field. The format of the header is keyword, value. Refer to the SuperCam user guide for examples of the HDU headers. The data section of HDU 3 and 4 will contain SOH information from the Mast Unit and Body Unit, respectively.

4.4.2.1 HDU commons to all data products

4.4.2.1.1 HDU 0 (Primary HDU)
Historically, the primary HDU of a FITS file is reserved for the primary header. The data part of the primary HDU can only be of Image format. Because of the specificity of the SuperCAM data files, it is kept empty.
4.4.2.1.2 HDU 1 (Reserved)
4.4.2.1.3 HDU 2 (Timeline)
The data in this HDU contains the timeline of the filling of the generic buffer with commands and command replies. This data is stored under the form of a FITS bintable, that contain the following columns:

- ms_since_last_sync (double) : this specifies the offset in millisecond at which the command was sent (or command reply received) since the last RSYNC (specified in the primary header in the keywork RCE_SYNC (in epoch).
- cmd_name (string): the command name according to the supercam command dictionary.
- opcode (string): the command opcode
- type (string): can be *cmd* (command), *data, spectra*.
- data_length (double): specifies the length of the data (or command arguments, or spectra array) that followed the opcode in the generic buffer.
- args (string): contains a string representation of the python dictionary containing the command arguments to that command. Dictionary is empty in case type = data or type = spectra
- data (string): contains a string representation of the hex values of the command replies. In case the data is reported elsewhere (spectra,replies of MU_SEND_HOUSEKEEPINGS, MU_RECORD_SOUND, COLLECT_SOH, MU_TAKE_IR_SPECTRUM, MU_SEND_LASER_DATA), this field points to the corresponding HDU.

4.4.2.1.4 HDU 3 (Mast Unit SOH)
This HDU contains the command reply to MU_SEND_HOUSEKEEPINGS command. Each SOH (according to the command dictionary) is reported in one separate column.
The columns order is the following:

- 27 numeric SOH in their raw integer values
- 24 digital SOH in their Boolean values
- Millisecond offset sync the last sync of the command reply.
- 27 numeric SOH converted to physical units using the appropriate transfer functions.

4.4.2.1.5 HDU 4 (Body Unit SOH)

This HDU contains the command reply to COLLECT_SOH command. Each SOH (according to the command dictionary) is reported in one separate column.
The columns order is the following:

- 83 numeric SOH in their raw integer values
- Millisecond offset sync the last sync of the command reply.
- 29 numeric SOH converted to physical units using the appropriate transfer functions.

4.4.2.2 HDUs for specific data types

The data content for HDU 5 through 7 depends on the data type(technique).

Table 4-5 Format of SuperCam FITS EDRs

		LIBS	RAMAN	TRLS	IRSpec	Passive	Micro	Image	SOH	AF
Primary (HDU0)	Header	Generic Header, Status Flags, File information (EDR file, Creator, …), HDU infos, GSE params.								
	Data	-	-	-	-	-	-	-	-	-
HDU1	Header									
	Data	Reserved								
HDU2	Header									
	Data	Timeline (BinTable)								
HDU3	Header									
	Data	MU SOH (BinTable)								
HDU4	Header									
	Data	BU SOH (BinTable)								
HDU5	Header									
	Data	Laser Data	Laser Data	Laser Data	Temp, RF, Status			Red		AF data
HDU6	Header									
	Data	Darks (bef)	Darks (bef)	Darks (bef)	Darks	Darks	AudioData	Green1		
HDU7	Header									
	Data	Active	Active	Active	Active			Green2		
HDU8	Header									
	Data	Darks (aft)	Darks (aft)	Darks (aft)				Blue		

4.4.3 Cruise EDR Structure

Field	Num Bytes	Description
Packet header marker	2	0xEBAA – Start of packet header marker.
RCE time sync	4	Last time sync value received from RCE
Data definition version	2	Version of the instrument command dictionary. Defines specifics of each command parameter, and the structure of individual data products like SOH.
NV xmit buffer count	4	Non-volatile count of the number of times the generic buffer has been transmitted to the RCE. Acts as a unique identifier for the buffer.
BU software version	4	Unique identifier for the software version, is a unix timestamp of when the software was built.
CNDH firmware version	4	Unique identifier for the CNDH firmware version, is a unix timestamp of when the firmware was built.
BU hardware identifier	4	Hardware identifier for a specific BU CNDH board. Used to know which SOH engineering unit conversion factors to apply to the data.
SE firmware version	4	Unique identifier for the CNDH firmware version, is a unix timestamp of when the firmware was built.
SE hardware identifier	4	Hardware identifier for a specific SE board. Used to know which SOH engineering unit conversion factors to apply to the data.
MU version identifier	4	Version of the MU, response from the MU send version command.
MU hardware identifier	4	Hardware identifier for a specific MU board. Used to know which housekeeping engineering unit conversion factors to apply to the data.
Xmit data ID	4	The data_id parameter sent with the XMIT_DATA command
Reserved	4	Not used in the current version. Filled with 0's.
Number of sections	4	Number of command or data sections contained in this buffer. Does not include the header section or the trailer section.
Generic data buffer size	4	The number of bytes of the entire generic buffer. Includes commands, data, header, and trailer. Every byte from the packet header marker through the Fletcher checksum.

4.4.3.1 Cruise Sequence ScamSci Data format

Each ScamSci data file will have the following: (except 08-MU_RMI_test)

1) Implicit Data Size

   4 Byte Implicit Data size

2) Instrument Transfer Frame header (8 bytes)

   1 Byte Op code (eb)
   3 Byte status flags
   4 Byte size of data

3) Followed by the Generic Data Packet Header (56 bytes):

Data Size in bytes	Description	Value
2	Packet header marker	0xEBAA – Start of packet header marker.
4	RCE time sync	Last time sync value received from RCE
2	Data definition version	Version of the instrument command dictionary. Defines specifics of each command parameter, and the structure of individual data products like SOH.
4	NV xmit buffer count	Non-volatile count of the number of times the generic buffer has been transmitted to the RCE. Acts as a unique identifier for the buffer.
4	BU software version	Unique identifier for the software version, is a unix timestamp of when the software was built.
4	CNDH firmware version	Unique identifier for the CNDH firmware version, is a unix timestamp of when the firmware was built.
4	BU hardware identifier	Hardware identifier for a specific BU CNDH board. Used to know which SOH engineering unit conversion factors to apply to the data.
4	SE firmware version	Unique identifier for the CNDH firmware version, is a unix timestamp of when the firmware was built.
4	SE hardware identifier	Hardware identifier for a specific SE board. Used to know which SOH engineering unit conversion factors to apply to the data.
4	MU version identifier	Version of the MU, response from the MU send version command.
4	MU hardware identifier	Hardware identifier for a specific MU board. Used to know which housekeeping engineering unit conversion factors to apply to the data.
4	Xmit data ID	The data_id parameter sent with the XMIT_DATA command
4	Reserved	Not used in the current version. Filled with 0's.
4	Number of sections	Number of command or data sections contained in this buffer. Does not include the header section or the trailer section.
4	Generic data buffer size	The number of bytes of the entire generic buffer. Includes commands, data, header, and trailer. Every byte from the packet header marker through the Fletcher checksum.

4) XmitData Packets (defined below)

5) Followed by Generic Data Packet Footer

Data Size in Bytes	Description	Value

2	Packet trailer marker	0xEBFF – Start of packet trailer marker.
4	Millisecond count	Number of milliseconds since last time sync value.
4	Pad length	Number of pad bytes added to the buffer to make it 4 byte word aligned. Valid values are 0, 1, 2, and 3
<as req'd>	Pad data	Fill values of 0x00 as needed to 4 byte align the buffer. Will not exist if pad length is zero.
4	Fletcher checksum	Fletcher checksum across all bytes of the data buffer starting with the packet header marker through the pad data bytes. Uses the same calculation as defined for the RCE transfer frame.

6) Instrument Transfer Frame Checksum

4 Byte

Format of Cruise Sequence Science Data Packet Format

01-pwr_on_off_to_IDLE
XmitDataID : 10100

Data Size	Description	Value
2 byte	TIME_SYNC	0xecf8
4 byte	millisec offset	1619095
4 byte	byte size of data/args	4
4 byte	command argument(s)	pad = 0
2 byte	DO_BIST	0xecf4
4 byte	millisec offset	185
4 byte	byte size of data/args	0
2 byte	Built In Self Test DATA	0xedf4
4 byte	millisec offset	190
4 byte	byte size of data/args	4
4 byte	Data reply	
2 byte	SE_RESET	0xec67
4 byte	millisec offset	490
4 byte	byte size of data/args	0
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	1100
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	1105
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	XMIT_DATA	0xec8e
4 byte	millisec offset	1440
4 byte	byte size of data/args	4
4 byte	command argument(s)	dataID = 0x00010100

01-pwr_on_off_to_IDLE
XmitDataID : 10101

Data Size	Description	Value
2 byte	ENABLE_MU	0xec77
4 byte	millisec offset	2120
4 byte	byte size of data/args	1
1 byte	command argument(s)	enableMU = 1
2 byte	MU_BOOT_NOP	0xec90
4 byte	millisec offset	4375
4 byte	byte size of data/args	0
2 byte	MU_BOOT_JUMP	0xec93
4 byte	millisec offset	4590
4 byte	byte size of data/args	1
1 byte	command argument(s)	afswNum = 4
2 byte	MU_NOP	0xeca0
4 byte	millisec offset	4815
4 byte	byte size of data/args	0
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	5060
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	5180
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	5390
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	5395
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	XMIT_DATA	0xec8e
4 byte	millisec offset	5755
4 byte	byte size of data/args	4
4 byte	command argument(s)	dataID = 0x00010101

01-pwr_on_off_to_IDLE
XmitDataID : 10102

Data Size	Description	Value
2 byte	MU_CONFIGURE_UART	0xecc0
4 byte	millisec offset	6625
4 byte	byte size of data/args	1
1 byte	command argument(s)	baudRate = 3
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	7385
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	7415
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	7750
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	7755
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	XMIT_DATA	0xec8e
4 byte	millisec offset	8080
4 byte	byte size of data/args	4
4 byte	command argument(s)	dataID = 0x00010102

01-pwr_on_off_to_IDLE
XmitDataID : 10103

Data Size	Description	Value
2 byte	DEBUG_INFO	0xecf7
4 byte	millisec offset	8970
4 byte	byte size of data/args	1
1 byte	command argument(s)	arg = 0
2 byte	DEBUG_INFO DATA	0xedf7
4 byte	millisec offset	8970
4 byte	byte size of data/args	4
4 byte	Data reply	Last Cmd Received
2 byte	DEBUG_INFO	0xecf7
4 byte	millisec offset	9375
4 byte	byte size of data/args	1
1 byte	command argument(s)	arg = 1
2 byte	DEBUG_INFO DATA	0xedf7
4 byte	millisec offset	9375
4 byte	byte size of data/args	4
4 byte	Data reply	# of valid cmds received
2 byte	DEBUG_INFO	0xecf7
4 byte	millisec offset	9755
4 byte	byte size of data/args	1
1 byte	command argument(s)	arg = 2
2 byte	DEBUG_INFO DATA	0xedf7
4 byte	millisec offset	9755
4 byte	byte size of data/args	4
4 byte	Data reply	# of rejected cmds
2 byte	DUMP_ERROR_LOG	0xec59
4 byte	millisec offset	10085
4 byte	byte size of data/args	0
2 byte	ERROR_LOG DATA	0xed59
4 byte	millisec offset	10085
4 byte	byte size of data/args	4096
4096 byte	Data reply	Error log
2 byte	XMIT_DATA	0xec8e
4 byte	millisec offset	10460
4 byte	byte size of data/args	4
4 byte	command argument(s)	dataID = 0x00010103

02-BU_MU_reset_test
XmitDataID : 10200

Data Size	Description	Value
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	50700
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	50730
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	51010
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	51015
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	XMIT_DATA	0xec8e
4 byte	millisec offset	51300
4 byte	byte size of data/args	4
4 byte	command argument(s)	dataID = 0x00010200

02-BU_MU_reset_test
XmitDataID : 10201

Data Size	Description	Value
2 byte	RESET	0xecf2
4 byte	millisec offset	52040
4 byte	byte size of data/args	0
2 byte	SE_RESET	0xec67
4 byte	millisec offset	54280
4 byte	byte size of data/args	0
2 byte	MU_BOOT_NOP	0xec90
4 byte	millisec offset	54880
4 byte	byte size of data/args	0
2 byte	MU_BOOT_JUMP	0xec93
4 byte	millisec offset	55065
4 byte	byte size of data/args	1
1 byte	command argument(s)	afswNum = 4
2 byte	MU_NOP	0xeca0
4 byte	millisec offset	55295
4 byte	byte size of data/args	0
2 byte	MU_CONFIGURE_UART	0xecc0
4 byte	millisec offset	55480
4 byte	byte size of data/args	1
1 byte	command argument(s)	baudRate = 3
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	56245
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	56275
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	56530
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	56535
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	XMIT_DATA	0xec8e
4 byte	millisec offset	56855
4 byte	byte size of data/args	4

4 byte	command argument(s)	dataID = 0x00010201

03-BU_CCD_TEC_cooling_test
XmitDataID : 10300

Data Size	Description	Value
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	84270
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	84300
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	84610
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	84615
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	ENABLE_TEC	0xec78
4 byte	millisec offset	84980
4 byte	byte size of data/args	1
1 byte	command argument(s)	enableTec = 1
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	85175
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	85180
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	90565
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	90570
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	95865

4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	95870
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	101195
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	101200
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	106500
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	106505
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	112075
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	112080
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	117360
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	117365
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	122675
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	122680

4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	127975
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	127980
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	133255
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	133260
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	138565
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	138570
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	143845
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	143850
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	149160
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	149165
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	ENABLE_TEC	0xec78

4 byte	millisec offset	149495
4 byte	byte size of data/args	1
1 byte	command argument(s)	enableTec = 0
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	149745
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	149775
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	150080
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	150085
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	XMIT_DATA	0xec8e
4 byte	millisec offset	150455
4 byte	byte size of data/args	4
4 byte	command argument(s)	dataID = 0x00010300

04-MU_thermal_test

XmitDataID : 10400

Data Size	Description	Value
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	267340
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	267370
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	267630
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	267635

4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_DC_DC_CHAIN_ON_OFF	0xecb0
4 byte	millisec offset	267940
4 byte	byte size of data/args	1
1 byte	command argument(s)	onOff = 1
2 byte	MU_CONFIGURE_THERMAL_CONTROL	0xecc5
4 byte	millisec offset	270125
4 byte	byte size of data/args	22
22 byte	command argument(s)	cwl_set_point = 25 cwl_temp_hk = 16 cwl_proportional_gain = 255 cwl_integral_gain = 8 cwl_derivative_gain = 0 laser_set_point = 25 laser_ref_temperature = 1 laser_proportional_gain = 150 laser_integral_gain = 2 laser_derivative_gain = 0 laser_doubler_set_point = 25 laser_doubler_ref_temperature = 3 laser_doubler_proportional_gain = 80 laser_doubler_inegral_gain = 1 laser_doubler_derivative_gain = 0 irs_tec_mode = 1 irs_tec_set_point = -50 irs_tec_integral_threshold = 10 irs_tec_proportional_gain = 200 irs_tec_integral_gain = 20 irs_tec_max_pwm = 20 irs_tec_pwm_open_loop = 0
2 byte	MU_THERMAL_CONTROL_ON_OFF	0xecb3
4 byte	millisec offset	270300
4 byte	byte size of data/args	1
1 byte	command argument(s)	onOff = 0x0F
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	280505
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	280535
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	280810
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52

4 byte	millisec offset	280815
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	291150
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	291180
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	291470
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	291475
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	301805
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	301835
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	302145
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	302150
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	312525
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	312555
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52

4 byte	millisec offset	312825
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	312830
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	323090
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	323120
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	323400
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	323405
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	333685
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	333715
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	333990
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	333995
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	344260
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	344290

4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	344570
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	344575
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	354885
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	354915
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	355165
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	355170
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	365480
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	365510
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	365790
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	365795
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	376130
4 byte	byte size of data/args	4

4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	376160
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	376475
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	376480
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	386870
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	386900
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	387180
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	387185
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	397460
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	397490
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	397760
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	397765
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH

2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	408035
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	408065
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	408345
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	408350
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_THERMAL_CONTROL_ON_OFF	0xecb3
4 byte	millisec offset	408630
4 byte	byte size of data/args	1
1 byte	command argument(s)	onOff = 0
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	418825
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	418855
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	419110
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	419115
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	429425
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	429455
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH

2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	429705
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	429710
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	440020
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	440050
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	440345
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	440350
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	450650
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	450680
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	451055
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	451060
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	461410
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3

4 byte	millisec offset	461440
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	461745
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	461750
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	472035
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	472065
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	472355
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	472360
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	482640
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	482670
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	482945
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	482950
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	493250

4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	493280
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	493520
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	493525
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	503835
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	503865
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	504130
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	504135
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	514450
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	514480
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	514810
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	514815
4 byte	byte size of data/args	364

364 byte	Data reply	BU SOH
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	525215
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	525245
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	525610
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	525615
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	535970
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	536000
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	536305
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	536310
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	546605
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	546635
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	546945
4 byte	byte size of data/args	0

2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	546950
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	557225
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	557255
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	557525
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	557530
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_DC_DC_CHAIN_ON_OFF	0xecb0
4 byte	millisec offset	557835
4 byte	byte size of data/args	1
1 byte	command argument(s)	onOff = 0
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	558015
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	558045
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	558295
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	558300
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	XMIT_DATA	0xec8e
4 byte	millisec offset	558605
4 byte	byte size of data/args	4
4 byte	command argument(s)	dataID = 0x00010400

05-BU_CCD_test
XmitDataID : 10500

Data Size	Description	Value
2 byte	SE_RESET	0xec67
4 byte	millisec offset	642290
4 byte	byte size of data/args	0
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	642915
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	642945
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	643210
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	643215
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_DC_DC_CHAIN_ON_OFF	0xecb0
4 byte	millisec offset	643540
4 byte	byte size of data/args	1
1 byte	command argument(s)	onOff = 1
2 byte	MU_LASER_CONVERTERS_ON_OFF	0xecb1
4 byte	millisec offset	645720
4 byte	byte size of data/args	1
1 byte	command argument(s)	onOff = 0
2 byte	CONFIGURE_HVPS	0xec73
4 byte	millisec offset	645900
4 byte	byte size of data/args	3
3 byte	command argument(s)	enableHvps = 1 hvpsGain = 3000
2 byte	CONFIGURE_INTENSIFIER	0xec72
4 byte	millisec offset	649555
4 byte	byte size of data/args	8
8 byte	command argument(s)	openTimeTicks = 64 durationTimeTicks = 1000
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	649735
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF

2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	649765
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	650045
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	650050
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	CONFIGURE_CCD_HORZ_TIMING	0xec70
4 byte	millisec offset	650325
4 byte	byte size of data/args	33
33 byte	command argument(s)	spectrometerSelect = 7 debug = 0 mode2D = 1 integrationTime = 12 ROclockLimit = 74 R1clockHi = 0 R1clockLo = 66 R2clockHi = 62 R2clockLo = 72 R3clockHi = 68 R3clockLo = 31 ORSTclockHi = 62 ORSTclockLo = 69 ADCclockHi = 19 ADCclockLo = 51 clampHi = 62 clampLo = 14
2 byte	CONFIGURE_CCD_VERT_TIMING	0xec6f
4 byte	millisec offset	650540
4 byte	byte size of data/args	15
15 byte	command argument(s)	spectrometerSelect = 7 colTimerStop = 536 coli1Hi = 179 coli1Lo = 447 coli2Hi = 358 coli2Lo = 90 coli3Hi = 1 coli3Lo = 269
2 byte	CONFIGURE_CCD_REGIONS	0xec71
4 byte	millisec offset	650720
4 byte	byte size of data/args	17
17 byte	command argument(s)	spectrometerSelect = 4 skipRows1 = 40 sumRows1 = 120 region2enable = 1 skipRows2 = 25 sumRows2 = 110 region3enable = 1 skipRows3 = 5 sumRows3 = 130 lastSkipRows = 85
2 byte	CONFIGURE_CCD_REGIONS	0xec71
4 byte	millisec offset	650895
4 byte	byte size of data/args	17

17 byte	command argument(s)	spectrometerSelect = 2 skipRows1 = 150 sumRows1 = 200 region2enable = 0 skipRows2 = 0 sumRows2 = 0 region3enable = 0 skipRows3 = 0 sumRows3 = 0 lastSkipRows = 165
2 byte	CONFIGURE_CCD_REGIONS	0xec71
4 byte	millisec offset	651085
4 byte	byte size of data/args	17
17 byte	command argument(s)	spectrometerSelect = 1 skipRows1 = 150 sumRows1 = 200 region2enable = 0 skipRows2 = 0 sumRows2 = 0 region3enable = 0 skipRows3 = 0 sumRows3 = 0 lastSkipRows = 165
2 byte	DO_SPECTRA	0xec7c
4 byte	millisec offset	651295
4 byte	byte size of data/args	20
20 byte	command argument(s)	spectrometerSelect = 7 enableLaser = 0 collectID = 0x00010500 laserRepRate = 78125 shotsPerSpectra = 1 numSpectra = 1 performStatistics = 0 config = 0 doNotStoreSpectra = 0
2 byte	VIO Data	0xee02
4 byte	millisec offset	654085
4 byte	byte size of data/args	2212440
2212440 byte	VIO 2D Data	uint array(2148x515)
2 byte	UV Data	0xee03
4 byte	millisec offset	654235
4 byte	byte size of data/args	2212440
2212440 byte	UV 2D Data	uint array(2148x515)
2 byte	Transmission - VNIR Data	0xee01
4 byte	millisec offset	654365
4 byte	byte size of data/args	2212440
2212440 byte	Tranmission 2D Data	uint array(2148x515)
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	654620
4 byte	byte size of data/args	4
4 byte	command argument(s)	hkMask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	654650
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH

2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	654930
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	654935
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_DC_DC_CHAIN_ON_OFF	0xecb0
4 byte	millisec offset	655220
4 byte	byte size of data/args	1
1 byte	command argument(s)	onOff = 0
2 byte	CONFIGURE_HVPS	0xec73
4 byte	millisec offset	655425
4 byte	byte size of data/args	3
3 byte	command argument(s)	enableHvps = 0 hvpsGain = 0
2 byte	SE_RESET	0xec67
4 byte	millisec offset	656575
4 byte	byte size of data/args	0
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	657175
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	657205
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	657475
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	657480
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	XMIT_DATA	0xec8e
4 byte	millisec offset	657795
4 byte	byte size of data/args	4
4 byte	command argument(s)	dataID = 0x00010500

06-spectra_collection_test
XmitDataID : 10600

Data Size	Description	Value
2 byte	SE_RESET	0xec67
4 byte	millisec offset	733545
4 byte	byte size of data/args	0
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	734160
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	734190
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	734555
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	734560
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_DC_DC_CHAIN_ON_OFF	0xecb0
4 byte	millisec offset	734925
4 byte	byte size of data/args	1
1 byte	command argument(s)	onOff = 1
2 byte	MU_LASER_CONVERTERS_ON_OFF	0xecb1
4 byte	millisec offset	737135
4 byte	byte size of data/args	1
1 byte	command argument(s)	onOff = 1
2 byte	CONFIGURE_HVPS	0xec73
4 byte	millisec offset	737385
4 byte	byte size of data/args	3
3 byte	command argument(s)	enableHvps = 1 hvpsGain = 3000
2 byte	CONFIGURE_INTENSIFIER	0xec72
4 byte	millisec offset	741060
4 byte	byte size of data/args	8
8 byte	command argument(s)	openTimeTicks = 64 durationTimeTicks = 1000
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	741305
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF

2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	741335
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	741670
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	741675
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	CONFIGURE_CCD_HORZ_TIMING	0xec70
4 byte	millisec offset	742075
4 byte	byte size of data/args	33
33 byte	command argument(s)	spectrometerSelect = 7 debug = 1 mode2D = 0 integrationTime = 12 ROclockLimit = 74 R1clockHi = 0 R1clockLo = 66 R2clockHi = 62 R2clockLo = 72 R3clockHi = 68 R3clockLo = 31 ORSTclockHi = 62 ORSTclockLo = 69 ADCclockHi = 19 ADCclockLo = 51 clampHi = 62 clampLo = 14
2 byte	CONFIGURE_CCD_VERT_TIMING	0xec6f
4 byte	millisec offset	742290
4 byte	byte size of data/args	15
15 byte	command argument(s)	spectrometerSelect = 7 colTimerStop = 536 coli1Hi = 179 coli1Lo = 447 coli2Hi = 358 coli2Lo = 90 coli3Hi = 1 coli3Lo = 269
2 byte	CONFIGURE_CCD_REGIONS	0xec71
4 byte	millisec offset	742500
4 byte	byte size of data/args	17
17 byte	command argument(s)	spectrometerSelect = 4 skipRows1 = 40 sumRows1 = 120 region2enable = 1 skipRows2 = 25 sumRows2 = 110 region3enable = 1 skipRows3 = 5 sumRows3 = 130 lastSkipRows = 85
2 byte	CONFIGURE_CCD_REGIONS	0xec71
4 byte	millisec offset	742745
4 byte	byte size of data/args	17

17 byte	command argument(s)	spectrometerSelect = 2 skipRows1 = 150 sumRows1 = 200 region2enable = 0 skipRows2 = 0 sumRows2 = 0 region3enable = 0 skipRows3 = 0 sumRows3 = 0 lastSkipRows = 165
2 byte	CONFIGURE_CCD_REGIONS	0xec71
4 byte	millisec offset	742965
4 byte	byte size of data/args	17
17 byte	command argument(s)	spectrometerSelect = 1 skipRows1 = 220 sumRows1 = 200 region2enable = 0 skipRows2 = 0 sumRows2 = 0 region3enable = 0 skipRows3 = 0 sumRows3 = 0 lastSkipRows = 95
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	743185
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	743215
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	743570
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	743575
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_CONFIGURE_LASER	0xecc2
4 byte	millisec offset	743915
4 byte	byte size of data/args	7
7 byte	command argument(s)	pumping_current = 0 pumping_duration = 220 pockels_delay = 0 shot_frequency = 10 laser_selection = 0x47 num_shots = 20
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	746165
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	746195
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52

4 byte	millisec offset	746530
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	746535
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	DO_SPECTRA	0xec7c
4 byte	millisec offset	746955
4 byte	byte size of data/args	20
20 byte	command argument(s)	spectrometerSelect = 4 enableLaser = 0 collectID = 0x00010600 laserRepRate = 78125 shotsPerSpectra = 20 numSpectra = 1 performStatistics = 0 config = 0 doNotStoreSpectra = 0
2 byte	Transmission - VNIR Data	0xee01
4 byte	millisec offset	748885
4 byte	byte size of data/args	12888
12888 byte	Raman Data	uint array (6444)
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	749070
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	749100
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	749470
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	749475
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	XMIT_DATA	0xec8e
4 byte	millisec offset	749760
4 byte	byte size of data/args	4
4 byte	command argument(s)	dataID = 0x00010600

06-spectra_collection_test
XmitDataID : 10601

Data Size	Description	Value
2 byte	MU_CONFIGURE_LASER	0xecc2
4 byte	millisec offset	752635
4 byte	byte size of data/args	7
7 byte	command argument(s)	pumping_current = 0 pumping_duration = 200 pockels_delay = 0 shot_frequency = 3 laser_selection = 0x52 num_shots = 40
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	754810
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	754840
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	755125
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	755130
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	DO_SPECTRA	0xec7c
4 byte	millisec offset	755455
4 byte	byte size of data/args	20
20 byte	command argument(s)	spectrometerSelect = 7 enableLaser = 0 collectID = 0x00010601 laserRepRate = 260417 shotsPerSpectra = 40 numSpectra = 1 performStatistics = 0 config = 0 doNotStoreSpectra = 0
2 byte	VIO Data	0xee02
4 byte	millisec offset	755455
4 byte	byte size of data/args	4296
4296 byte	VIO Spectra Data	uint array (2148)
2 byte	UV Data	0xee03
4 byte	millisec offset	755460
4 byte	byte size of data/args	4296

4296 byte	UV Spectra Data	uint array (2148)
2 byte	Transmission - VNIR Data	0xee01
4 byte	millisec offset	768485
4 byte	byte size of data/args	12888
12888 byte	Transmission Spectra Data	uint array(6444)
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	768640
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	768670
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	768960
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	768965
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_LASER_CONVERTERS_ON_OFF	0xecb1
4 byte	millisec offset	769245
4 byte	byte size of data/args	1
1 byte	command argument(s)	onOff = 0
2 byte	MU_DC_DC_CHAIN_ON_OFF	0xecb0
4 byte	millisec offset	769455
4 byte	byte size of data/args	1
1 byte	command argument(s)	onOff = 0
2 byte	CONFIGURE_HVPS	0xec73
4 byte	millisec offset	769630
4 byte	byte size of data/args	3
3 byte	command argument(s)	enableHvps = 0 hvpsGain = 0
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	770775
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	770805
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52

4 byte	millisec offset	771105
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	771110
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	XMIT_DATA	0xec8e
4 byte	millisec offset	771440
4 byte	byte size of data/args	4
4 byte	command argument(s)	dataID = 0x00010601

07-laser_subsystem_test
XmitDataID : 10700

Data Size	Description	Value
2 byte	SE_RESET	0xec67
4 byte	millisec offset	794010
4 byte	byte size of data/args	0
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	794620
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	794650
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	794970
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	794975
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_DC_DC_CHAIN_ON_OFF	0xecb0
4 byte	millisec offset	795285
4 byte	byte size of data/args	1
1 byte	command argument(s)	onOff = 1
2 byte	MU_LASER_CONVERTERS_ON_OFF	0xecb1
4 byte	millisec offset	797470
4 byte	byte size of data/args	1
1 byte	command argument(s)	onOff = 1
2 byte	CONFIGURE_HVPS	0xec73

4 byte	millisec offset	797695
4 byte	byte size of data/args	3
3 byte	command argument(s)	enableHvps = 1 hvpsGain = 3000
2 byte	CONFIGURE_INTENSIFIER	0xec72
4 byte	millisec offset	801360
4 byte	byte size of data/args	8
8 byte	command argument(s)	openTimeTicks = 64 durationTimeTicks = 1000
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	801590
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	801620
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	801935
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	801940
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_CONFIGURE_LASER	0xecc2
4 byte	millisec offset	802305
4 byte	byte size of data/args	7
7 byte	command argument(s)	pumping_current = 140 pumping_duration = 220 pockels_delay = 0 shot_frequency = 10 laser_selection = 0x47 num_shots = 20
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	804495
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	804525
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	804850
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	804855

4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_CONFIGURE_LASER	0xecc2
4 byte	millisec offset	805170
4 byte	byte size of data/args	7
7 byte	command argument(s)	pumping_current = 125 pumping_duration = 200 pockels_delay = 0 shot_frequency = 3 laser_selection = 0x52 num_shots = 40
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	807360
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	807390
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	807710
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	807715
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_LASER_CONVERTERS_ON_OFF	0xecb1
4 byte	millisec offset	808025
4 byte	byte size of data/args	1
1 byte	command argument(s)	onOff = 0
2 byte	MU_DC_DC_CHAIN_ON_OFF	0xecb0
4 byte	millisec offset	808250
4 byte	byte size of data/args	1
1 byte	command argument(s)	onOff = 0
2 byte	CONFIGURE_HVPS	0xec73
4 byte	millisec offset	808440
4 byte	byte size of data/args	3
3 byte	command argument(s)	enableHvps = 0 hvpsGain = 0
2 byte	SE_RESET	0xec67
4 byte	millisec offset	809650
4 byte	byte size of data/args	0
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	810290
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF

2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	810320
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	810755
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	810760
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	XMIT_DATA	0xec8e
4 byte	millisec offset	811170
4 byte	byte size of data/args	4
4 byte	command argument(s)	dataID = 0x00010700

08-MU_RMI_test
XmitDataID : 10800

Data Size	Description	Value
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	841460
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	841490
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	841790
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	841795
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_CONFIGURE_HSSL	0xecc1
4 byte	millisec offset	842080
4 byte	byte size of data/args	1
1 byte	command argument(s)	hsslClock = 1
2 byte	MU_DC_DC_CHAIN_ON_OFF	0xecb0
4 byte	millisec offset	842290
4 byte	byte size of data/args	1
1 byte	command argument(s)	onOff = 1
2 byte	MU_RMI_ON_OFF	0xecb2
4 byte	millisec offset	844475
4 byte	byte size of data/args	1
1 byte	command argument(s)	onOff = 1
2 byte	MU_CONFIGURE_RMI_TABLE	0xecca
4 byte	millisec offset	846650
4 byte	byte size of data/args	1
1 byte	command argument(s)	tabel_num = 0
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	846835
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	846865
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH

2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	847170
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	847175
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_TAKE_RMI_IMAGE	0xecd2
4 byte	millisec offset	847500
4 byte	byte size of data/args	8
8 byte	command argument(s)	exposure_time = 300 hdr_mode = 0 roi_size = 2048 roi_y_position = 0 memory_bank = 0
2 byte	Take RMI IMAGE Reply	0xedd2
4 byte	millisec offset	848275
4 byte	byte size of data/args	173
173 byte	Data reply	
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	848430
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	848460
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	848755
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	848760
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HSS_DATA	0xecd4
4 byte	millisec offset	849040
4 byte	byte size of data/args	6
6 byte	command argument(s)	memory_bank = 0 num_words = 0x400000 type = 0
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	855955
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	855985

4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	856250
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	856255
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	XMIT_IMAGE	0xec80
4 byte	millisec offset	856580
4 byte	byte size of data/args	0
2 byte	XMIT_IMAGE	0xec80
4 byte	millisec offset	866110
4 byte	byte size of data/args	0
2 byte	XMIT_IMAGE	0xec80
4 byte	millisec offset	875745
4 byte	byte size of data/args	0
2 byte	XMIT_IMAGE	0xec80
4 byte	millisec offset	882125
4 byte	byte size of data/args	0
2 byte	MU_TAKE_RMI_IMAGE	0xecd2
4 byte	millisec offset	903305
4 byte	byte size of data/args	8
8 byte	command argument(s)	exposure_time = 600 hdr_mode = 0 roi_size = 2048 roi_y_position = 0 memory_bank = 1
2 byte	Take RMI IMAGE Reply	0xedd2
4 byte	millisec offset	904380
4 byte	byte size of data/args	173
173 byte	Data reply	
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	904550
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	904580
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	904900
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52

4 byte	millisec offset	904905
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HSS_DATA	0xecd4
4 byte	millisec offset	905230
4 byte	byte size of data/args	6
6 byte	command argument(s)	memory_bank = 1 num_words = 0x400000 type = 0
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	912160
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	912190
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	912465
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	912470
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	912805
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	912835
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	913105
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	913110
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	XMIT_IMAGE	0xec80
4 byte	millisec offset	913430
4 byte	byte size of data/args	0
2 byte	XMIT_IMAGE	0xec80
4 byte	millisec offset	923030

4 byte	byte size of data/args	0
2 byte	XMIT_IMAGE	0xec80
4 byte	millisec offset	932955
4 byte	byte size of data/args	0
2 byte	XMIT_IMAGE	0xec80
4 byte	millisec offset	939595
4 byte	byte size of data/args	0
2 byte	MU_RMI_ON_OFF	0xecb2
4 byte	millisec offset	959740
4 byte	byte size of data/args	1
1 byte	command argument(s)	onOff = 0
2 byte	MU_DC_DC_CHAIN_ON_OFF	0xecb0
4 byte	millisec offset	960215
4 byte	byte size of data/args	1
1 byte	command argument(s)	onOff = 0
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	960460
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	960490
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	960930
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	960935
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	XMIT_DATA	0xec8e
4 byte	millisec offset	961365
4 byte	byte size of data/args	4
4 byte	command argument(s)	dataID = 0x00010800

09-MU_IR_spectrometer_test
XmitDataID : 10900

Data Size	Description	Value
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	980550
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	980580
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	980850
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	980855
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_DC_DC_CHAIN_ON_OFF	0xecb0
4 byte	millisec offset	981175
4 byte	byte size of data/args	1
1 byte	command argument(s)	onOff = 1
2 byte	MU_CONFIGURE_THERMAL_CONTROL	0xecc5
4 byte	millisec offset	983360
4 byte	byte size of data/args	22
22 byte	command argument(s)	cwl_set_point = 25 cwl_temp_hk = 16 cwl_proportional_gain = 255 cwl_integral_gain = 8 cwl_derivative_gain = 0 laser_set_point = 25 laser_ref_temperature = 1 laser_proportional_gain = 150 laser_integral_gain = 2 laser_derivative_gain = 0 laser_doubler_set_point = 25 laser_doubler_ref_temperature = 3 laser_doubler_proportional_gain = 80 laser_doubler_inegral_gain = 1 laser_doubler_derivative_gain = 0 irs_tec_mode = 1 irs_tec_set_point = -50 irs_tec_integral_threshold = 10 irs_tec_proportional_gain = 200 irs_tec_integral_gain = 20

		irs_tec_max_pwm = 20 irs_tec_pwm_open_loop = 0
2 byte	MU_THERMAL_CONTROL_ON_OFF	0xecb3
4 byte	millisec offset	983550
4 byte	byte size of data/args	1
1 byte	command argument(s)	onOff = 0x08
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	1013755
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	1013785
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	1014060
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	1014065
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	1044385
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	1044415
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	1044765

4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	1044770
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	1075185
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	1075215
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	1075505
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	1075510
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	1105845
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	1105875
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	1106195
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	1106200
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_CONFIGURE_IR_FULL	0xecc8
4 byte	millisec offset	1106495
4 byte	byte size of data/args	4
4 byte	command argument(s)	accumulation_table_number = 4 analog_integration_time = 1 accumulation_offset = 0 driver_power = 0
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3

4 byte	millisec offset	1108685
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	1108715
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	1109065
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	1109070
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_TAKE_IR_SPECTRUM	0xecd1
4 byte	millisec offset	1109470
4 byte	byte size of data/args	1
1 byte	command argument(s)	spectra_mode = 1
2 byte	Take IR_SPECTRUM Reply	0xedd1
4 byte	millisec offset	1110095
4 byte	byte size of data/args	2824
2824 byte	Data reply	see mu command dictionary
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	1110265
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	1110295
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	1110655
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	1110660
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_THERMAL_CONTROL_ON_OFF	0xecb3
4 byte	millisec offset	1110995
4 byte	byte size of data/args	1
1 byte	command argument(s)	onOff = 0
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3

4 byte	millisec offset	1141240
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	1141270
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	1141540
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	1141545
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	1171875
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	1171905
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	1172175
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	1172180
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	1202505
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	1202535
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	1202825
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	1202830

4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	1233180
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	1233210
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	1233555
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	1233560
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_DC_DC_CHAIN_ON_OFF	0xecb0
4 byte	millisec offset	1233845
4 byte	byte size of data/args	1
1 byte	command argument(s)	onOff = 0
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	1234030
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	1234060
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	1234355
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	1234360
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	XMIT_DATA	0xec8e
4 byte	millisec offset	1234650
4 byte	byte size of data/args	4
4 byte	command argument(s)	dataID = 0x00010900

10-MU_focus_table_motor_move_test
XmitDataID : 10A00

Data Size	Description	Value
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	1357915
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	1357945
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	1358330
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	1358335
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_DC_DC_CHAIN_ON_OFF	0xecb0
4 byte	millisec offset	1358685
4 byte	byte size of data/args	1
1 byte	command argument(s)	onOff = 1
2 byte	MU_MOTOR_ON_OFF	0xecb5
4 byte	millisec offset	1360890
4 byte	byte size of data/args	1
1 byte	command argument(s)	onOff = 1
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	1363150
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	1363180
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	1363455
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	1363460
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_STEPPER_MOTOR_MOVE	0xecd0

4 byte	millisec offset	1363785
4 byte	byte size of data/args	8
8 byte	command argument(s)	num_displacements = 1 num_steps_per_displacement = 14000 direction = 1 bypass_on_off = 0 sampling_on_off = 0 gain = 0
2 byte	MU_STEPPER_MOTOR_MOVE	0xecd0
4 byte	millisec offset	1363785
4 byte	byte size of data/args	8
8 byte	command argument(s)	
2 byte	Stepper Motor Move Reply	0xedd0
4 byte	millisec offset	1408910
4 byte	byte size of data/args	6
6 byte	Data reply	see MU command dictionary
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	1409070
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	1409100
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	1409405
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	1409410
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_STEPPER_MOTOR_MOVE	0xecd0
4 byte	millisec offset	1409740
4 byte	byte size of data/args	8
8 byte	command argument(s)	num_displacements = 1 num_steps_per_displacement = 14000 direction = 0 bypass_on_off = 0 sampling_on_off = 0 gain = 0
2 byte	MU_STEPPER_MOTOR_MOVE	0xecd0
4 byte	millisec offset	1409740
4 byte	byte size of data/args	8
8 byte	command argument(s)	
2 byte	Stepper Motor Move Reply	0xedd0
4 byte	millisec offset	1522040
4 byte	byte size of data/args	6

6 byte	Data reply	see MU command dictionary
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	1522235
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	1522265
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	1522640
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	1522645
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_STEPPER_MOTOR_MOVE	0xecd0
4 byte	millisec offset	1523025
4 byte	byte size of data/args	8
8 byte	command argument(s)	num_displacements = 1 num_steps_per_displacement = 5300 direction = 1 bypass_on_off = 0 sampling_on_off = 0 gain = 0
2 byte	MU_STEPPER_MOTOR_MOVE	0xecd0
4 byte	millisec offset	1523025
4 byte	byte size of data/args	8
8 byte	command argument(s)	
2 byte	Stepper Motor Move Reply	0xedd0
4 byte	millisec offset	1575155
4 byte	byte size of data/args	6
6 byte	Data reply	see MU command dictionary
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	1575310
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	1575340
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	1575605
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52

4 byte	millisec offset	1575610
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	MU_MOTOR_ON_OFF	0xecb5
4 byte	millisec offset	1575925
4 byte	byte size of data/args	1
1 byte	command argument(s)	onOff = 0
2 byte	MU_DC_DC_CHAIN_ON_OFF	0xecb0
4 byte	millisec offset	1576110
4 byte	byte size of data/args	1
1 byte	command argument(s)	onOff = 0
2 byte	MU_SEND_HOUSEKEEPINGS	0xeca3
4 byte	millisec offset	1576320
4 byte	byte size of data/args	4
4 byte	command argument(s)	hk_mask = 0xFFFFFFFF
2 byte	MU SOH DATA	0xeda3
4 byte	millisec offset	1576350
4 byte	byte size of data/args	68
68 byte	Data reply	MU SOH
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	1576620
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	1576625
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	ENABLE_MU	0xec77
4 byte	millisec offset	1576950
4 byte	byte size of data/args	1
1 byte	command argument(s)	onOff = 0
2 byte	COLLECT_SOH	0xec52
4 byte	millisec offset	1577130
4 byte	byte size of data/args	0
2 byte	BU SOH DATA	0xed52
4 byte	millisec offset	1577135
4 byte	byte size of data/args	364
364 byte	Data reply	BU SOH
2 byte	XMIT_DATA	0xec8e
4 byte	millisec offset	1577465
4 byte	byte size of data/args	4
4 byte	command argument(s)	dataID = 0x00010A00

4.5 EDR Product Validation

Validation of the M2020 EDRs will fall into two primary categories: automated and manual. Automated validation will be performed on every EDR product produced for the mission. Manual validation will only be performed on a subset.

Automated validation will be performed as a part of the archiving process and will be done simultaneously with the archive volume validation. Validation operations performed will include such things as verification that the checksum in the label matches a calculated checksum for the data product (i.e., that the data product included in the archive is identical to that produced by the real-time process), a validation of the PDS syntax of the label, a check of the label values against the database and against the index tables included on the archive volume, and checks for internal consistency of the label items. The latter include such things as verifying that the product creation date is later than the earth received time, and comparing the geometry pointing information with the specified target. As problems are discovered and/or new possibilities identified for automated verification, they will be added to the validation procedure.

Manual validation of the data will be performed both as spot-checking through-out the life of the mission, and comprehensive validation of a sub-set of the data (for example, a couple of days' worth of data). These products will be viewed by a human being. Validation in this case will include inspection of the spectra or other data object for errors (like missing lines) not specified in the label parameters, verification that the target shown / apparent geometry matches that specified in the labels, verification that the product is viewable using the specified software tools, and a general check for any problems that might not have been anticipated in the automated validation procedure.

5 RDR Product Specification

RDR data products described in this document will be generated by the SuperCam Science Team using team-developed software at Los Alamos National Laboratory (LANL) in New Mexico, Institut de Recherche en Astrophysique et Planétologie (IRAP) and Centre National d'Etudes spatiales (CNES) in France. The RDRs produced will be "processed" data. Supercam will produce two levels of RDRs, calibrated (CDRs) and derived (DDRs). The input will be one or more image or spectrum EDR or RDR data products and the output will be formatted according to this SIS. Additional metadata may be added by the software to the product label. There may be multiple versions of image and spectrum RDRs. A SuperCam PDS User Guide is included as part of the SuperCam PDS archive as an introduction to the RDR data produced by SuperCam. This document is a quick reference on how to find the SuperCam data and identify the technique from the file name as well as how access the data.

5.1 Calibrated RDR General Processing

Processing is different for each SuperCam technique, as described in this section. In general, each RDR process inherits the metadata from its source EDR/RDR, modifying a portion of the metadata as necessary to reflect the subsequent output RDR. The technical details of each processing step for the different SuperCam products or techniques is documented in Wiens et al. (2013) (Section 1.3, Applicable Documents).

5.1.1 Libs Processing (CDRs)

The LIBS CDRs are located in the data_calibrated_spectra collection. The data processing for LIBS benefits from the experience on ChemCam. In many aspects, the ChemCam data reduction and calibration pipeline can be adapted to SuperCam's LIBS spectra. The main steps of data processing and calibration that have been defined by the team are the following:

1. **Saturation check.** If the raw DN for the channel exceeds 64500 counts.
2. **Dark subtraction**. The mean of the active spectra are subtracted from each active spectrum.
3. **Computation of Statistics. (if shot to shot)** Calculate mean, median and standard deviation of all spectra.
4. **Stitching of the spectral bands in the VNIR range.** The three parts of the VNIR spectrometer (a.k.a. green, orange and red) will be merged, selecting the best signal-to-noise vs resolution compromise.
5. **Denoise.** A wavelet transform algorithm developed by Olivier Forni is applied to each spectrum.
6. **Wavelength calibration**. A pattern-recognition algorithm developed by team members has been tested on ChemCam data. A similar algorithm will be used to calibrate SuperCam data.
7. **Correction for the Instrument Response function**. Conversion from counts to photons.

8. **Continuum removal.** Wavelet transform algorithm developed by the team.
9. **Convert Photons-to-radiance**.

5.1.2 Raman Processing (CDRs)

The Raman CDRs are located in the data_calibrated_spectra collection. The data processing steps for Raman products that have been defined by the team are the following:

1. **Saturation check.** If the raw DN for the channel exceeds 64500 counts.
2. **Remove Cosmic Rays** for both active and dark spectra through a sigma-clipping algorithm. The parameters of this algorithm can be tuned (number of iterations, sigma-levels).
3. **Dark subtraction**. Dark spectra (or their mean) are subtracted from each active spectrum.
4. **Computation of Statistics. (if shot to shot)** Calculate mean, median and standard deviation of all spectra.
5. **Stitching of the spectral bands in the VNIR range.** The three parts of the VNIR spectrometer (a.k.a. green, orange and red) will be merged, selecting the best signal-to-noise vs resolution compromise**.**
6. **Denoise.** A wavelet transform algorithm developed by team member O. Forni is applied to each spectrum.
7. **Wavenumber calibration.** Wavelength calibration for Raman will be done using LIBS on the calibration targets. For Raman, this will be converted to Raman shift in wavenumbers. (1/wavelength in cm e.g. cm^-1 or 1/cm) The calibration targets (e.g. diamond) will be used to ensure that the calibration is accurate.
8. **Correction for the Instrument Response function**. Convert counts to photons.

5.1.3 TRLS Processing (CDRs)

The TRLS CDRs are located in the data_calibrated_spectra collection. The data processing steps for TRLS that have been defined by the team are the following:

1. **Saturation check.** If the raw DN for the channel exceeds 64500 counts.
2. **Remove Cosmic Rays** for both active and dark spectra through a sigma-clipping algorithm. The parameters of this algorithm can be tuned (number of iterations, sigma-levels).
3. **Dark subtraction**. Dark spectra (or their mean) are subtracted from each active spectrum.
4. **Computation of Statistics. (if shot to shot)** Calculate mean, median and standard deviation of all spectra.
5. **Stitching of the spectral bands in the VNIR range.** The three parts of the VNIR spectrometer (a.k.a. green, orange and red) will be merged, selecting the best signal-to-noise vs resolution compromise**.**
6. **Denoise.** A wavelet transform algorithm developed by team member O. Forni is applied to each spectrum.

7. **Wavenumber calibration.** Wavelength calibration for TRLS will be done using LIBS on the calibration targets. TRLS data will be provided in wavelength (nm) and will use the LIBS wavelength calibration.
8. **Correction for the Instrument Response function**. Convert counts to photons.

5.1.4 VIS Processing (CDRs)

The VIS CDRs are located in the data_calibrated_spectra collection. The data processing steps for VIS that have been defined by the team are following:

1. **Saturation check.** If the raw DN for the channel exceeds 64500 counts.
2. **Dark Subtraction. (in DN)**
3. **Conversion to Radiance.**
4. **Conversion to reflectance** by division by a reference (Sol 20 or Sol 184 depending on the instrument's settings) and by scaling the geometry of the illumination: multiplication by the ratio of the cosines of the solar incidence and by the ratio of the squares of the Mars- to-Sun distance.

5.1.5 IRSpec Processing (CDRs)

The IRS CDRs are located in the data_calibrated_spectra collection. The data processing steps for IRS that have been defined by the team are the following:

1. **Saturation check.** If the raw DN for the channel exceeds 60000 counts.
2. **Dark smoothing** to only keep the low frequency trend (Savitzky-Golay algorithm), assuming that high frequencies consist of noise and readout spikes
3. **Dark subtraction**
4. **Radiometric calibration** using ITF_{20} (Sol 20 White SCCT measurement as reference) data is converted into radiance (this is in a column in HDU 6 of the CDR)
5. **Thermal calibration.** A correction vector to the ITF is derived from a model and the actual temperature of the IR electronic board and applied to reflectance data. It corrects the dependence of the instrument's response to the RF power shift due to IR board temperature variations.
6. **Glitch removal.** Glitches are EMI/EMC artifacts from an unknown origin, randomly occurring as signal decrease on individual spectral channels. They are related to RF power perturbations which affect the AOTF diffraction efficiency. Glitches are removed by detection on RF power measurements and interpolation to undisturbed neighboring spectral channels
7. **Spikes removal.** Spikes occur on both Signal and Dark measurements, as random negative or positive peaks several times above the noise level. They are statistically detected, using a 3 σ filter and removed like the glitches.
8. **Conversion to reflectance.** Radiance data is divided by a solar spectrum calculated through an illumination model, taking into account the geometry and time of the observation. Corrected reflectance is given as a column (I_F) in HDU 6 of the CDR
9. **Atmospheric correction.** Reflectance data is corrected from the atmospheric (mainly CO2) absorption features by maximizing the correlation with a reference spectrum fitted to the actual

column density. Atmospheric correction spectrum is given as a column (I_F_Atm) in HDU 6 of the CDR.

5.1.6 Microphone Processing

The Microphone CDRs are located in the data_calibrated_audio collection. The data processing for the Microphone that have been defined by the team are the following:

1. **Conversion DN to volt.**
2. **Conversion Volt to pascal**
3. **Mean signal subtraction**

5.1.7 RMI Processing

The RMI CDRs are located in the data_radcal_rmi collection. The RMI CDRs are Partially radiometrically corrected. The data processing steps for RMI that have been defined by the team are the following:

1. **Demosaicing (if input is Bayer filter).** Convert Bayer's input (2048x2048 RGGB channel) to a 2048x2048 image for each of the RGB channel. This is done using the Malvar algorithm.
2. **Crop out-of-field pixels.** MA mask with a radius of 468 pixels and centered in x=512, y=520 is applied, so that any pixel outside this mask is set to 0.
3. **White Balance.** The RGB channels are corrected using this scaling factors: [1, 0.781, 0.797] to achieve a good white balance, based on the first white paint observations performed on Mars.
4. **Flatfield correction.** Each plane is corrected by "vignetting effect" using a preliminary flat field image observed during the ground testing. This is being improved as we obtained more Mars data.

5.2 Calibrated RDR Product Types

Descriptions for the various RDR product types are provided in this section. Refer to table 5-1 to identify the SuperCam technique and product type for the CDR. The CDR product id can be found in the type fields of the filename (Section 6.1.3).

Table 5-1 Mars 2020 SuperCam CDR product ID

SuperCam Technique	Actives	Darks	Type	CDR Product ID
LIBS	All	All		CL0
	All	Stats		CL1
	Stats	Stats		CL2
	All	None		CL3
	Stats	None		CL4

	None	All		CL5
	None	Stats		CL6
2D	-	-		CL9
Non-nominal	-	-		CLS
RAMAN	All	All		CR0
	All	Stats		CR1
	Stats	Stats		CR2
	All	None		CR3
	Stats	None		CR4
	None	All		CR5
	None	Stats		CR6
2D	-	-		CR9
Non-nominal	-	-		CRS
TRLS	All	All		CT0
	All	Stats		CT1
	Stats	Stats		CT2
	All	None		CT3
	Stats	None		CT4
	None	All		CT5
	None	Stats		CT6
2D	-	-		CT9
Non-nominal	-	-		CTS
VIS	-	All		CP0
	-	Stats		CP1
	-	IRSpec		CP2
		Double VIS		CP3
		IRScan		CP4
Passive 2D	-	-		CP9
Non-nominal	-	-		CPS
Microphone	-	-	Libs-sync	CA0
	-	-	Libs-cont	CA1
	-	-	Standalone	CA2
RMI	-	-	Radcal	CI_

5.3 Calibrated RDR Product Format

The main file format for CDRs is FITS (Flexible Image Transport System). FITS is a flexible, portable, file standard that allows to structure data in an efficient way. It is widely used in astronomy and there exist a number of tools that allow to read their content. A FITS file consists of a number of partitions (called HDU for Header & Data Unit), each one divided into a “header” and a “data” section. The header is made of several ASCII cards that contain a key (8-digit, alphanumeric), a value (numeric, literal or Boolean) and a comment section. The data section is very flexible: it can contain ASCII TABLES, BINARY TABLE and IMAGES. For a thorough description of the FITS format, see https://fits.gsfc.nasa.gov

5.4 Calibrated RDR Product Structure

The SuperCam RDRs will include detached PDS4 labels. The CDR FITS structure will depend on which kind of data it contains. The following table describes the content of each

HDU. HDU 0 to 4 consist only of metadata and are common to all file types. HDU 0 to 4 will contain the same information as EDR HDUs 0 to 4. Some additional information about the data processing parameters, the pipeline and some quality checks will be added to each section. Successive HDUs will contain data specific for each analysis.

The data section of HDU 1 and 2 will be in the form of an ASCII table. Important information from each of these data will be converted into a header key of that HDU for rapid access. The data section of HDU 3 and 4 will contain SOH information from the Mast Unit and Body Unit, respectively.

		Science product (Clean Data Record – CDR)						
	Activity	LIBS	RAMAN	TRLS	VIS	IRSpec	Micro	RMI
	Activity tag	CL[0-6,S]	CR[0-6,S]	CT[0-6,S]	CP[0,1,S]	CP2	CA[0-2]	CI
HDU #	HDU section	Content						
0	Header	Generic Header, Status Flags, Activity keywords, Data Processing Keywords						
1	Data	ODL Label						
2	Data	Timeline (I-cmds and arguments)						
3	Data	MU SOH						
4	Data	BU SOH						
5	Data	Laser Data	Laser Data	Laser Data	Empty	IRS Spectel SOH	Laser Data/Empty HDU	Red
6	Data	Shot-to-shot spectra	Shot-to-shot spectra	Delay Spectra	Shot-to-shot spectra	Spectra	Time series	Green
7	Data	Statistics spectra	Statistics spectra	Statistics spectra	Statistics spectra	Wavelength	Saturation Mask	Blue
8	Data	Wavelength	Wavelength	Wavelength	Wavelength	IRS Acquisition SOH		
9	Data	Saturation Mask	Saturation Mask	Saturation Mask	Saturation Mask	Data Quality		
10					Shot-to-shot reflectance spectra			
11					Stastistics reflectance spectra			

5.4.1 HDU common to all data products

5.4.1.1 HDU 0 (Primary HDU)

Historically, the primary HDU of a FITS file is reserved for the primary header. The data part of the primary HDU can only be of Image format. Because of the specificity of the SuperCAM data files, it is kept empty. The header is composed of several cards, which are composed by a 8-digits (alphanumeric) key, a value and a comment. In order to keep things tidy, the header is divided in several sections.

- Standard FITS keys
- Target information, extracted from the JPL header or from the GSE experimentInfoLog file.
- Generic buffer header
- File producer and date
- Data processing parameters
- Dataset Children, specifying the organization of the file.

5.4.1.2 HDU 1 (Reserved)

This will contain the ODL LABEL, which includes IDPH and S-command Arguments.

5.4.1.3 HDU 2 (Timeline)

The data in this HDU contains the timeline (the I-command and their arguments as they are encountered during the parsing of the EDR file). This data is stored in the form of a FITS bintable that contain the following columns:

- ms_since_last_sync (double) : this specifies the offset in millisecond at which the command was sent (or command reply received) since the last RSYNC (specified in the primary header in the keywork RCE_SYNC (in epoch).
- cmd_name (string): the command name according to the SuperCam command dictionary
- opcode (string): the command opcode
- type (string): can be *cmd* (command), *data, spectra*.
- data_length (double): specifies the length of the data (or command arguments, or spectra array) that followed the opcode in the generic buffer.
- args (string): contains a string representation of the python dictionary containing the command arguments to that command. Dictionary is empty in case type = data or type = spectra
- data (string): contains a string representation of the hex values of the command replies. In case the data is reported elsewhere (spectra,replies of MU_SEND_HOUSEKEEPINGS, MU_RECORD_SOUND, COLLECT_SOH, MU_TAKE_IR_SPECTRUM, MU_SEND_LASER_DATA), this field points to the corresponding HDU.

The header part consists of the FITS basic header (table descriptions, number of rows, etc). Some of the command arguments will be extensively used during the data processing. These values (to be defined by the science and data processing teams) will be extracted and reported in specific FITS label keywords.

5.4.1.4 HDU 3 (Mast Unit SOH)

This HDU contains the command reply to MU_SEND_HOUSEKEEPINGS command. Each SOH (according to the command dictionary) is reported in one separate column.
The columns order is the following:

- 27 numeric SOH in their raw integer values
- 24 digital SOH in their Boolean values
- Millisecond offset sync the last sync of the command reply.
- 27 numeric SOH converted to physical units using the appropriate transfer functions.

5.4.1.5 HDU 4 (Body Unit SOH)

This HDU contains the command reply to COLLECT_SOH command. Each SOH (according to the command dictionary) is reported in one separate column.
The columns order is the following:

- 83 numeric SOH in their raw integer values
- Millisecond offset sync the last sync of the command reply.
- 29 numeric SOH converted to physical units using the appropriate transfer functions.

5.4.2 Technique-dependent HDSs

HDUs 5 to 9 are specific to the SuperCam science module that has been used to obtain the data. See Table 5-2 for the contents of each type. For example, a Raman RDR file contains Laser Data, Shot-to-shot Spectra, Statistics Spectra, Wavelength, and Saturation HDUs.

5.4.2.1 HDU 5 – Laser Data

A table of the laser data. One row for each laser shot. The columns are Shot Number, Stack voltage Raw, Pumping Current Raw, Optical Power Raw, Stack voltage (Volts), Pumping Current (Amps), Optical Power (mJ).

5.4.2.2 HDU 5 – IRS Acquisition SOH

A table of IRS state of health values. The columns are table acquisition number, Analog integration time value, power offset of the AOTF driver, nacc offset, AOTF initial temperature, configuration of the measurement (DET_SEL, MEAS_MODE), measurement initial temperature, IRS TEC setpoint, AOTF final temperature, TEC hot side temperature, IR board temperature, Detector selected, IRS mode (SPECTRUM or SCAN).

5.4.2.3 HDU 5 – Red

A table of the partially radiometrically corrected images red pixels 2048 x 2048.

5.4.2.4 HDU 6 - Shot-to-shot Spectra / Spectra

A table of the processed calibrated spectra data. The columns are either one column for radiance value for each shot if all shots recorded or empty if statistics are collected. One row for each pixel.

5.4.2.5 HDU 6 – Delay Spectra
A table specific to TRLS. One row for each pixel. The columns are for each spectra taken.

5.4.2.6 HDU 6 – Spectra
A table specific to the IRS spectrometer. One row for each spectral channel (spectel). The columns are DN, Radiance, Atmosphere corrected radiance (Rad_Atm_corr), and reflectance (observed signal / modeled solar irradiance).

5.4.2.7 HDU 6 – Time Series
A table specific to the microphone data. A measurement of frequency vs time. The "SOUND" HDU of a microphone recording hosts each sample of the microphone acquisition. In the case of a continuous recording (CA1 and CA2 files), the sound is sampled at the frequency specified in the primary header key MIC_SAMP (25000 or 100000 Hz), and its duration is in the key MIC_DURA (in seconds). For the pulsed mic (CA0) it is the concatenation of each of the 60-msec recordings at a frequency of MIC_SAMP around each laser pulse. The pulsed microphone synchronization is described in Maurice, S., et al (2021).

5.4.2.8 HDU 6- Green
A table of the partially radiometrically corrected image green pixels 2048 x 2048.

5.4.2.9 HDU 7 – Statistics Spectra
A table of the mean, median and standard deviation of all shots in the shot-to-shot spectra table.

5.4.2.10 HDU 7 – Wavelength (IRSpec only)
A table specific to IRSpec data one row for each spectel and two columns, one for the wavelength at the center of each spectel, corrected for AOTF temperature dependence, and one for Full width half maximum of the AOTF line spread function.

5.4.2.11 HDU 7 – Saturation Mask (Microphone only)
A table of logical expressions, TRUE or FALSE, to indicate if data has reached saturation level (>64500 DN). One row for each wavelength. The columns are either one column for each shot if all shots recorded or statistics values, mean, median and standard deviation of all shots, if statistics recorded.

5.4.2.12 HDU 7 – Blue
A table of the partially radiometrically corrected image blue pixels 2048 x 2048.

5.4.2.13 HDU 8 – Wavelength (LIBS, TRLS, VIS)
A table to indicate the wavelength and instrument response function(IRF) for each pixel. The wavelength is a mapping of CCD pixels to the expected wavelength of light incident on that pixel. Stored as a 16 bit signed floating point number. The IRF maps the number of photons incident on the telescope aperture to an expected signal recorded by the CCD. It is in units of “photons/DN” where DN is Digital Number, which is just another name for “CCD counts.”

5.4.2.14 HDU 8 – Wavelength (RAMAN)

A table to indicate the wavenumber and the instrument response function (IRF) for each pixel. The wavenumber is Raman Shift (cm^-1), which represent the difference between the incident laser frequency and the scattered frequency. The IRF maps the number of photons incident on the telescope aperture to an expected signal recorded by the CCD. It is in units of "photons/DN" where DN is Digital Number, which is just another name for "CCD counts."

5.4.2.15 HDU 9 – Saturation (LIBS, RAMAN, TRLS, VIS)

A table of logical expressions, TRUE or FALSE, to indicate if data has reached saturation level (>64500 DN). One row for each wavelength. The columns are either one column for each shot if all shots recorded or statistics values, mean, median and standard deviation of all shots, if statistics recorded.

5.4.2.16 HDU 9 – Data Quality (IRSpec)

For the IRSpec data this table indicates data quality. One row for each spectel. There are four columns: Saturation, Glitch, Spike and StDev. The values Saturation Flitch and Spike are set to 1 for True, 0 for False, NaN if a particular spectel was not used in that particular observation. The StDev is the Standard Deviation of the I_F spectrum. Glitches are identified where there is an RF Power Offset. IRSpec that have glitches may be reprocessed at a later date.

5.4.2.17 HDU 10 – Shot-to-shot Reflectance Spectra (VIS)

A table of the processed calibrated spectra data for VIS. The columns represent the reflectance value for all the spectra recorded (HDU6). One row for each pixel.

5.4.2.18 HDU 11 – Statistics Reflectance Spectra (VIS)

A table of the mean, median and standard deviation of all shots in the shot-to-shot spectra table (as HDU7) but in units of reflectance.

5.5 Derived RDR General Processing

The file format for DDRs varies depending on the technique. Not all techniques will have a derived data record. Each DDR will have a different filename convention. For a list of the Derived RDR product types, see Table 4-3.

5.5.1 LIBS DDR

LIBS Elemental Abundances

SuperCam uses laser-induced breakdown spectroscopy (LIBS) to obtain semi-quantitative elemental abundances from rasters of small (300–600 μm in diameter) observation points at distances of 2-7 m. Results are currently obtained for oxides of the major elements: SiO_2, TiO_2, Al_2O_3, FeO_T (total iron content assessed as FeO), MgO, CaO, Na_2O, K_2O, and the totals of these elements, all given as weight percent (wt%). Quantitative chemistry is based on multivariate statistical models trained on laboratory spectra of a suite of diverse, well-characterized reference samples, collected on the flight body unit coupled to the engineering

qualification model of the SuperCam mast unit. This suite of spectra and corresponding known compositions are available in the file libs_spectral_library_reference.csv in the calibration_supercam collection (https://pds-geosciences.wustl.edu/m2020/urn-nasa-pds-mars2020_supercam/calibration_supercam/libs_spectral_library_reference.csv). The file also includes metadata indicating which samples were used as training and test sets for each of the major elements, which were excluded as outliers, and information about the geologic type and origin of the samples. For details about data preprocessing and multivariate model selection, as well as known limitations of the current calibration, please refer to Anderson et al., 2021 (https://doi.org/10.1016/j.sab.2021.106347).

The LIBS elemental abundances are stored in a csv file named *supercam_libs_moc.csv* and is located in the data_derived_spectra collection. The table of results is organized with each average spectrum as a row, and abundances of each major element oxide as columns. An additional column for each major element contains the standard deviation of the derived abundances of that element for each of the individual laser shots on that location. This standard deviation helps to identify points with increased variation between shots, which may indicate changes in composition with laser ablation depth. Above the results, the Minimum, First quartile, Median, Third quartile, and Maximum value of the training set for each major element oxide are listed. These are useful for evaluating whether a derived composition exceeds the range over which the multivariate model is expected to be reliable. It is the nature of many multivariate models that they do not extrapolate beyond the training set range very well, so an abundance approaching or exceeding the training set maximum value may indicate a higher true composition. The best estimate of the accuracy is the RMSEP; this is listed above the abundances of each major element oxide. At the right-hand side of the table, the "Total" column lists the sum of major element oxides. The abundances are not normalized to 100 wt.%, so this value can vary. Low totals can indicate that a target contains significant amounts of un-quantified elements (e.g. sulfur, hydrogen, etc.) A final column lists the total emission of the spectrum prior to normalization. This can be used to assess the quality of the spectrum: low signal may indicate a poor-quality spectrum (though users should visually inspect spectra as well).

5.5.2 RAMAN DDR

Raman Features:

The Raman Derived Data file is a table of features with columns for summary of findings, number of features identified and then a column for raman shift(s) of that feature, id of feature, potential minerals observed and confidence of the identification. This list only shows the data that has identifiable features. By identifiable features, we mean all those Raman features that are related to the sample itself. These are identified by the SuperCam Raman experts by evaluating the relative intensity of the features with respect to the noise and spectrum background, by checking that the width corresponds to values as expected from Raman features from SuperCam and by evaluating the presence of the feature in the multiple shots that are acquired per point. The Raman derived data product is a csv file in the data_derived_spectra collection of the SuperCam bundle with the name supercam_raman.csv. Also included is an xml PDS4 label with the same base name and an extension of ".xml".

The CSV file contains the following fields: (all fields are ASCII string except Number of Features).

File: PDS Logical Identifier for CDR referenced
Target: Martian target name
Summary: summary of detections on the sample, and any other comments. Motivation of feature identification for low confidence features.
Number_of_features: number of Raman features detected on the spectra.
Then, for each feature there are the following fields, repeated Number_of_features times:
Feature1_RamanShift: Raman shift in which Feature 1 is found in wavenumbers (cm-1)
Feature1_ID: Assignment and or identification of the feature
Feature1_PotentialMinerals: Potential minerals related to the feature
Feature1_Confidence: confidence of the identification. It can be low, medium or high.

The Feature Confidence Level is a qualitative parameter obtained as a combined assessment made by the SuperCam team. This assessment is based upon the experience gained by the analysis and study of SCAM representative spectra both on Mars and on laboratory setups, relying also on the analysis of factors such as signal to noise ratios and peak position. Additionally, it also takes into account information from other techniques, e.g. LIBS and VISIR, and imagery (these other data are not necessarily from the exact same position the Raman spectrum of interest were taken and is instead made on a higher level).

5.5.3 TRLS DDR

(TBD) This section will be completed in advance of the SuperCam RDR PDS Peer Review Part 2.

5.5.4 VIS DDR

The VIS CDR now are both in units of Radiance and Reflectance (cf. section 6.1.2.4, VIS processing). Further processing of the VIS and by-products of the VIS CDRs are under investigation by the team.

5.5.5 IRS DDR

(TBD) This section will be completed in advance of the SuperCam RDR PDS Peer Review Part 2.

5.5.6 RMI Mosaic DDR

The RMI Mosaic derived RDR or DDR, can be found in the data_special_rmi collection. There are two possible formats of the RMI mosaic. The first is a file composed of one or more RMI images of the same target region or sequence, annotated with shot locations. The second format is a fused mosaic without shot annotation. The format of the RMI mosaic DDRs is a png file. If the mosaic is documentation of a SuperCam science observation it may contain markers indicating location of science observation points, as well as a key to

indicate the SuperCam technique used. The RMI mosaic may also be just an image of land scape or region. The product and RMI records contained in the png are described in a separate PDS4 xml label with the same base name and an extension of ".xml".

5.5.7 Master Target DDR

The master target DDR is an observation log of the SuperCam science data product targets. It includes information like sol, sequence number, EDR filename, CDR filename, target name, distance to the target, etc. This file is contained in the data_observation_log collection. The file is a comma-separated values file. The file name is M2020_SCAM_MASTERLIST.CSV. The product and fields are described in a separate PDS4 xml label with the same base name and an extension of ".xml".

5.6 Derived RDR Product Types

5.6.1 LIBS DDR Processing

Laboratory data are masked to remove noisy portions of the spectra, and then the values from each spectrometer are normalized to the total signal from the spectrometer, so that the full spectrum sums to 5. In some cases, "peak binning" is applied prior to normalization. This binning uses the average training set spectrum to define local minima and maxima, and then each individual spectrum is summed between local minima, with the resulting values assigned the wavelength of the corresponding local maximum. This reduces the size of the spectra, consolidates signal from weak/broad lines, and increases robustness to wavelength shifts. In some cases, the spectra are also standardized: each spectral channel is mean-centered and scaled by the standard deviation. This reduces the influence of the strongest lines and amplifies the influence of weaker lines, resulting in improved results in some cases.

For each major element, the laboratory data were subdivided into five folds, with similar distribution of compositions ensured by sorting on the composition of interest prior to fold assignment. All spectra of the same target were included in the same fold. One fold was held out as an independent test set, the remaining four folds were used for model cross validation and optimization. A variety of multivariate regression models were evaluated, including Ordinary Least Squares (OLS), Partial Least Squares (PLS), Least Absolute Shrinkage and Selection Operator (LASSO), Orthogonal Matching Pursuit (OMP), Least Angle Regression (LARS), Support Vector Regression (SVR), Elastic Net, Gradient Boosting Regression (GBR), Random Forest Regression (RF). All of these methods were implemented using the scikit-learn library. In addition, Local Elastic Net and blended submodel regression were used. These methods were implemented via the Python Hyperspectral Analysis Tool (PyHAT). Each model was optimized by selecting the parameters that minimized the Root Mean Squared Error of Cross Validation (RMSECV). Final models were then chosen based on the accuracy when predicting the independent test set (Root Mean Squared Error of Prediction; RMSEP), as well as performance when obtaining the derived abundances of the SuperCam calibration targets, and through assessment of initial abundances of Mars targets.

For SiO_2, the derived abundances are an average of the results of GBR and PLS models. For TiO_2, RF was the most accurate model. For Al_2O_3, the reported compositions are an average of the abundances derived from RF, two different implementations of PLS, and Local Elastic Net. For FeOT and CaO, RF was the most accurate model. For MgO and Na_2O, GBR was the most accurate model. For K_2O, the reported compositions are an average of Elastic Net, SVR, Local Elastic Net, LASSO, and PLS results.

6 Standards Used in Generating Products

6.1 File Naming Standards

As described in section 4.2, SuperCam has several different data product types, depending on the technique used. The filename **Product Type field** (field 24-26) is used to identify the technique used.

Table 6-1 Product Type for SuperCam Technique

Technique	LIBS	RAMAN	TRLS	VIS	IR	MIC
Product Type	EL#	ER#	ET#	EP(0,1,3)	EP(2,4)	EA#

6.1.1 EDR Filename

Each M2020 EDR data product can be uniquely identified by incorporating into the product filename at minimum the Instrument ID, SCLK (or UTC), Product Type identifier, and Version number. The convention is illustrated in Section 6.1.2 below. Note that the SuperCam non-imaging products discussed in this SIS do not use the RMI specific fields “geometry”, “thumbnail”, and “camera specific”, so the first two fields are incorporated into the “product id” field, increasing it to five characters. **SuperCam non-imaging products use the “camera-specific” three N fields in SNNN for the point number from the command arguments DPO in the data product.**

The primary attributes of the filename nomenclature are:

a) Uniqueness - It must be unique unto itself without the file system’s directory path. This Protects against product overwrite as files are copied/moved within the file system and external to the file system, if managed correctly.

b) Metadata - It should be comprised of metadata fields that keep file bookkeeping and sorting intuitive to the human user. Even though autonomous file processing will be managed via databases, there will always be human-in-the-loop that puts a premium on filename intuition. Secondly, the metadata fields should be smartly selected based on their value to ground processing tools, as it is less CPU-intensive to extract information from the filename than from the label.

NOTE: Metadata information in the filename also resides in the product label.

The metadata fields have been selected based on Mars Exploration Rovers (MER) and Phoenix lessons learned. In general, the metadata fields are arranged to achieve:

a) Sortability - At the beginning of the filename resides a primary time oriented field such as Spacecraft Clock Start Count (SCLK). This allows for sorting of files on the M2020 file system by spacecraft data acquisition time as events occurred on Mars.

b) Readability - An effort is made to alternate Integer fields with ASCII character fields to Optimize differentiation of field boundaries for the human user.

6.1.2 SuperCam Non-Imaging EDR Filename Specifics

Table 6-2 describes the contents of the SuperCam Non-imaging EDR fields.

Table 6-2 Contents of the SuperCam Non-imaging EDR filename fields

Field	Contents	Description
1-2	LS	Instrument
3	_	Color Filter - Not used for SuperCam non-imaging EDR so default is underscore '_'
4	_	Special Processing flag, applicable to RDRs only. EDRs always have "_".
5-8	Sol number	Primary timestamp
9	_	Venue "_" : Flight (surface or cruise)
10-19	SCLK	Secondary time stamp: Spacecraft clock count, whole number of seconds
20		Underscore for readability. Always set to " "
21-23	SCLK fraction	Spacecraft clock count, fractional seconds
24-26	Product type	SuperCam EDR product type.
27	_	Geometry "_" : raw geometry
28		Thumbnail. Not used for non-imaging so default is " "
29-31	Site	Site location count from the RMC where the data was acquired.
32-35	Drive	Drive count (position within a Site location) from the RMC where the data was acquired.
36-44	Sequence	Sequence-ID : Identifier indicating the command sequenced the image was acquired from. Specific values will be assigned by the uplink team

45	_	Camera specific identifier. Not used for non-imaging so default is “ ”
46-48	Point number	Point number in raster, or 001 for single point
49		Downsample resolution. This will be “ ” in non-imaging ED
50-51	__	Compression. This will be “__” in non-imaging modes
52	P	Producer – Principal investigator of instrument
53-54	Version	Version number
55	.	Separator for filename and extension. Always set to “.”
56-59	fits	Extension

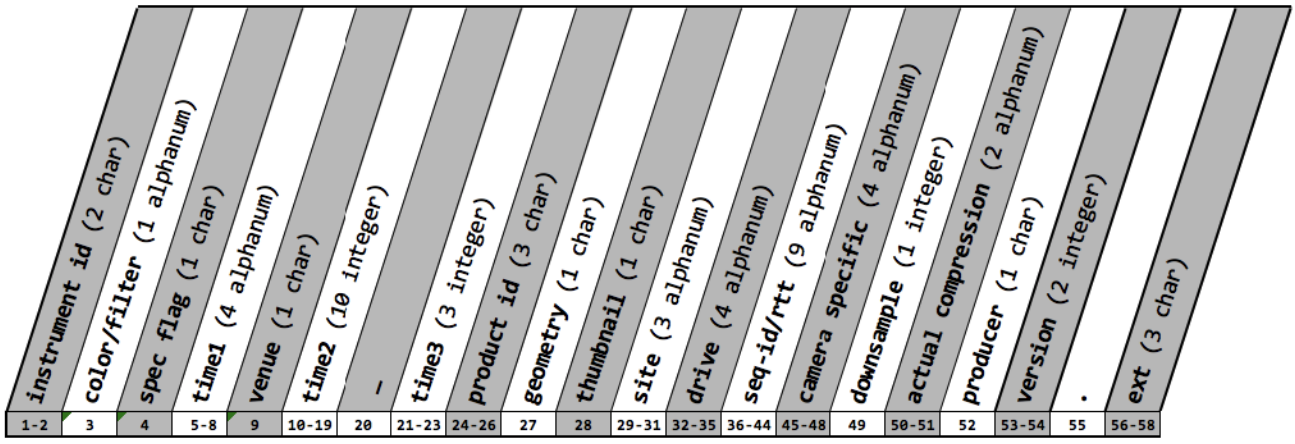
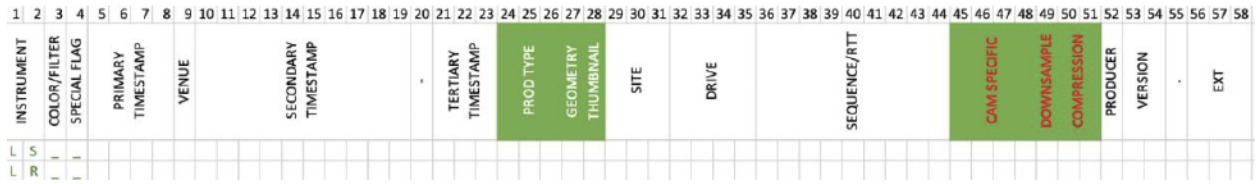


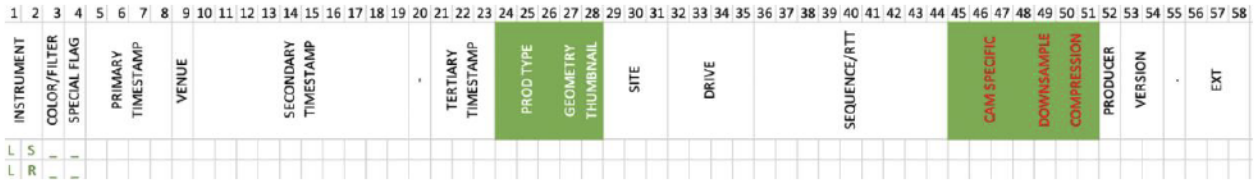
Figure 11 M2020 EDR Filename Template

Table 6-3 M2020 EDR Filename Convention



Field	Position (size, type)	Description
Instrument	01 (2, a)	Instrument Identifier: • FL : Front Hazcam Left (RCE-A) • FR : Front Hazcam Right (RCE-A) • FA : Front Hazcam Anaglyph (RCE-A) • FC : Front Hazcam Colorglyph (RCE-A) • BL : Front Hazcam Left (RCE-B) • BR : Front Hazcam Right (RCE-B) • BA : Front Hazcam Anaglyph (RCE-B) • BC : Front Hazcam Colorglyph (RCE-B) • CC : Cache Cam • EA : EDL Parachute Uplook Cam A (PUC-A) • EB : EDL Parachute Uplook Cam B (PUC-B) • EC : EDL Parachute Uplook Cam C (PUC-C) • ED : EDL Rover Downlook Cam (RDC) • EL : EDL Lander Vision System (LVS) • ES : EDL Descent Stage Downlook Cam (DSD) • EU : EDL Rover Uplook Cam (RUC) • HN : Mars Helicopter Navigation Cam • HS : Mars Helicopter Scout Cam • ZL : Mastcam-Z Left • ZR : Mastcam-Z Right • ZA : Mastcam-Z Anaglyph • ZC : Mastcam Colorglyph • WS : MEDA Skycam • NL : Navcam Left • NR : Navcam Right • NA : Navcam Anaglyph • NC : Navcam Colorglyph • PC : PIXL Micro Context Cam (MCC) • RL : Rear Hazcam Left • RR : Rear Hazcam Right • RA : Rear Hazcam Anaglyph • RC : Rear Hazcam Colorglyph

Field	Position (size, type)	Description
		• SC : SHERLOC Context Imager (ACI) • SI : SHERLOC Imaging (Watson) • IA : SHERLOC Imaging (Watson) Anaglyph (<G6.3) • SA: SHERLOC Imaging (Watson) Anaglyph (G6.3+) • IC : SHERLOC Imaging (Watson) Colorglyph (<G6.3) • SC: SHERLOC Imaging (Watson) Colorglyph (G6.3+) – note, what about ACI? • IL : Watson treated as left-eye (<G6.3) • SL: Watson treated as left-eye (G6.3+) • IR : Watson treated as right-eye (<G6.3) • SR: Watson treated as right-eye (G6.3+) • SR: SuperCam RMI (<G6.3) • LR : Supercam RMI (G6.3+) Non-imaging instrument identifiers: • SA : SuperCam All Spectrometers (<G6.3) • SD : SuperCam Diagnostic (<G6.3) • SG : SuperCam Generic (<G6.3) • SL : SuperCam Libs (<G6.3) • SM : SuperCam Mineral Data (<G6.3) • SP : SuperCam Passive (<G6.3) • LS : SuperCam all non-imaging EDRs (G6.3+) • ME : MEDA Environment • OX : MOXIE • PE : PIXL Engineering • PS : PIXL Spectrometer • SS : SHERLOC Spectrometer • XM : RIMFAX Mobile • XS : RIMFAX Stationary
Color/Filter	03 (1, i/a)	Color flag (see Section X.X CAM SIS): • E : Raw Bayer pattern • M : Grayscale image (Monochrome/Panchromatic) • A : Upper green bayer cells (G1 see Section X.X CAM SIS) • D : Lower green bayer cells (G2 see Section X.X CAM SIS) • O : Other • _ : N/A (SuperCam all types of EDR)



Field	Position (size, type)	Description

For all color cameras, the flag is set based on the color space depending on whether the product is a 3-band product, or an individual band.

Color Type	3-Band	Band 1	Band 2	Band 3
RGB	F	R	G	B
XYZ	T	X	Y	Z
xyY	C	J	K	L
HIS	P	H	S	I

For certain instruments, this flag may take on additional values indicating illumination or filter:

For PIXL, the flag can be set based on the LEDs used to illuminate the target.

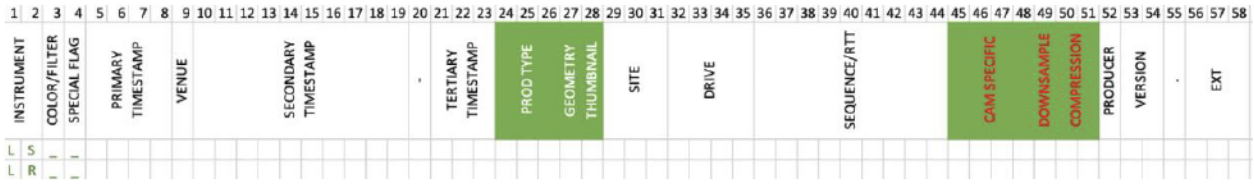
LED Color	Flag
Red	R
Green	G
Blue	B
Multiple	W
UV	U
Off	_

For MCZ and SHERLOC Imaging the flag can be filter or cover state.

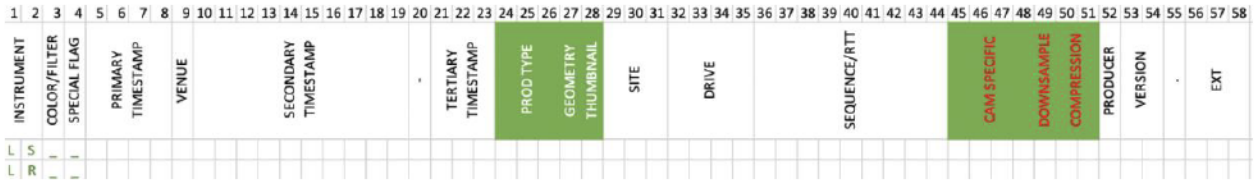
Instrument	Filters	Description
Mastcam-Z	0 – 7	See section X CAM SIS on MCZ filters
SHERLOC		See section X in CAM SIS on SHERLOC filters

Other color flags may be defined in the future.

Only "E", "F", or "M" or Filter/LED can appear in EDRs.



Field	Position (size, type)	Description
Special flag	04 (1, a)	Special Processing flag, applicable to RDRs only. EDRs always have "_". The special processing character is used to indicate off-nominal or special processing of the image. Examples include use of different correlation parameters, special stretches to eliminate shadows, reprocessing with different camera pointing, etc. The meaning of any individual character in this field (other than "_" which means nominal processing) will be defined on an ad-hoc basis as needed during the mission. Within one Sol or a range of sols, the character will be used consistently. So, this field can be used to group together all derived products resulting from one kind of special processing. An attempt will be made to maintain consistency across different sols as well, but this may not always be possible; thus the meaning of characters may change across different individual or ranges of sols. A database will be maintained containing all special processing designators that are used, the sols they relate to, and a description of the special processing that was done. This information will be included in the PDS archive.
Primary timestamp	05 (4, i/a)	Primary timestamp that is of coarser granularity than the Secondary timestamp. Value type is based on four scenarios: <u>Flight Cruise</u> Year-DOY (4 alphanumeric) - This field stores two metadata items in the order: a) One alpha character in range "A-Z" to designate Earth Year portion of the UTC-like time value, representing Years 2017 to 2042 b) Three integers in range "001-365" representing Day-of-Year (DOY) <u>Flight Surface</u> Sol (4 integer) - This field stores the 4-integer Sol (Mars solar day) of the <u>first</u> (i.e., lowest Clock time) acquired instrument data. <u>Ground Test in which SCLK in NOT reset</u> When SCLK continuously increments and does NOT repeat, there are two variants: a) Year-DOY (4 alphanumeric) - This field stores two metadata items in the order: 1. One alpha character in range "A-Z" to designate Earth Year portion of the UTC-like time value, representing Years 2017 to 2042 2. Three integers in range "001-365" representing Day-of-Year (DOY) – OR – b) Sol (4 integer) - This field stores the 4-integer Sol (Mars solar day) of the <u>first</u> (i.e., lowest Clock time) acquired instrument data. <u>Ground Test in which SCLK is reset</u> When SCLK is reset and repeats, we lose time "uniqueness". So, we have to change from SCLK to using "wall clock" derived from ERT and represent with a UTC-like format:

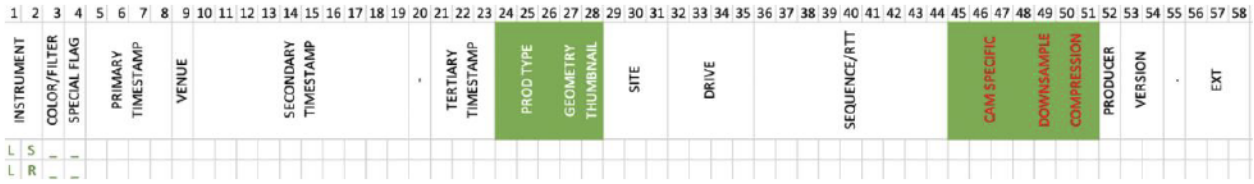


Field	Position (size, type)	Description
		DOY-Year (4 alphanumeric) - This field stores two metadata items in reverse order compared to the previous "Year-DOY" cases, indicating that the Secondary Time field (described later) contains ERT a) Three integers in range "001-365" representing Day-of-Year (DOY) One alpha character in range "A-Z" to designate Earth Year portion of the UTC-like time value, representing Years 2017 to 2042 The valid values, in their progression, are as follows (non-Hex):

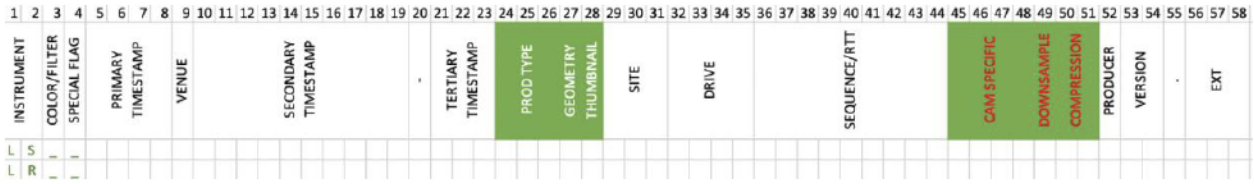
Scenario	Time Type	Value Format	Valid Values	Time Range
Flight Cruise	Year-DOY	[A-Z]<ddd>	"**A001**", "**A002**",…,"**A365**" "**B001**", "**B002**", … "**B365**" • • • "**Z001**", "**Z002**", … "**Z365**"	2017 - Days 1-365 2018 - Days 1-365 • • • 2042 - Days 1-365
		<aaaa>	"_ _ _ _" (4 underscores)	Value is out of range
Flight Surface	Sol	<nnnn>	"**0000**", "**0001**", … "**9999**"	0 thru 9999
		<aaaa>	"_ _ _ _" (4 underscores)	Value is out of range
Ground Test where SCLK is NOT reset	Year-DOY	(same as Flight Cruise)	(same as Flight Cruise)	(same as Flight Cruise)
	Sol	<nnnn>	"**0000**", "**0001**", … "**9999**"	0 thru 9999
		<aaaa>	"_ _ _ _" (4 underscores)	Value is out of range
Ground Test where SCLK is reset	DOY-Year	<ddd>[A-Z]	"**001A**", "**002A**", …,"**365A**", "**001B**", "**002B**",…, "**365B**", • • • "**001Z**", "**002Z**", … "**365Z**"	2017 - Days 1-365 2018 - Days 1-365 • • • 2042 - Days 1-365
		<aaaa>	"_ _ _ _" (4 underscores)	Value is out of range
Venue	9 (1, a)	Mission venue identifier: • _ : Flight (surface or cruise) • A : AVSTB (Avionics Testbed) • F : FSWTB (Flight Software Testbed)		

Field	Position (size, type)	Description
		• M : MSTB (Mission Systems Testbed) • S : "Scarecrow" • V : VSTB (Validation Systems Testbed)
Secondary timestamp	10 (10, i)	(10 integer) Secondary Timestamp that is of finer granularity than the Primary timestamp. Value type is based on four scenarios: Flight Cruise SCLK – This field stores the 10-integer SCLK (seconds). Which specific SCLK count (Start or End) is used depends on the instrument, but nominally it is the starting count of the first (i.e., lowest Clock time) acquired instrument data. Flight Surface SCLK – Same as for "Flight Cruise" Ground Test in which SCLK in NOT reset SCLK – Same as for "Flight Cruise" Ground Test in which SCLK is reset ERT - This field stores the ERT time portions Month, Day-of-month, Hour and Seconds as 10 integers in a UTC-like format The valid value formats are as follows:

Scenario	Time Type	Value Format	Valid Values	Time Range
Flight Cruise	SCLK	<sssssssss> (Seconds)	"**0000000000**", "**0000000001**", • • • "**9999999999**"	0 thru 9999999999
		<aaaaaaaaaa> (Alphabetic)	"_____" (10 underscores)	Value is out of range
Flight Surface	SCLK	(same as Flight Cruise)	(same as Flight Cruise)	(same as Flight Cruise)



Field	Position (size, type)	Description
		<table><tr><td>Ground Test where SCLK is NOT reset</td><td>SCLK</td><td>(same as Flight Cruise)</td><td>(same as Flight Cruise)</td><td>(same as Flight Cruise)</td></tr><tr><td rowspan="2">Ground Test where SCLK is reset</td><td rowspan="2">ERT</td><td><MMDDHHmmss> (Month, Day-of-month, Hour, Minute, Second)</td><td>“**0101010000**”, “**0101010001**”, • • • “**1231235959**”</td><td>January 1, 01:00:00 thru December 31, 23:59:59</td></tr><tr><td><aaaaaaaaaa> (Alphabetic)</td><td>“_ _ _ _ _ _ _ _ _ _” (10 underscores)</td><td>Value is out of range</td></tr></table>
_	20 (1, a)	Underscore for readability. Always set to “_”.
Milliseconds	21 (3, i)	Milliseconds of either the SCLK or UTC.
Product type	24 (3, a)	Product identifier. See Tables 4-1 and 4-2 for details. This field may be extended to 5 characters in non-imaging EDRs, as linearization and thumbnail do not apply.
Geometry	27 (1, a)	Linearization flag: • _ : Non-linearized (raw geometry) • L : Product has been linearized with nominal stereo partner • A : Product has been linearized with an actual stereo partner. Note that for the “A” case, an image can have multiple stereo partners and the linearized images will be different for each partner. A user will need to look in the ODL/VICAR label to determine which partner was used for linearization. **Note**: This field may be subsumed into the product ID field in non-imaging EDRs All image EDRs are Raw geometry (“_”).
Thumbnail	28 (1, a)	Thumbnail flag: • T : Product is a thumbnail N : Product is a non-thumbnail (full-frame, sub-frame, downsample) **Note:** This field may be subsumed into the product ID field in non-imaging EDRs.
Site	29 (3, i/a)	Site identifier (see section X.X on Site frames): Site location count from the RMC where the data was acquired.

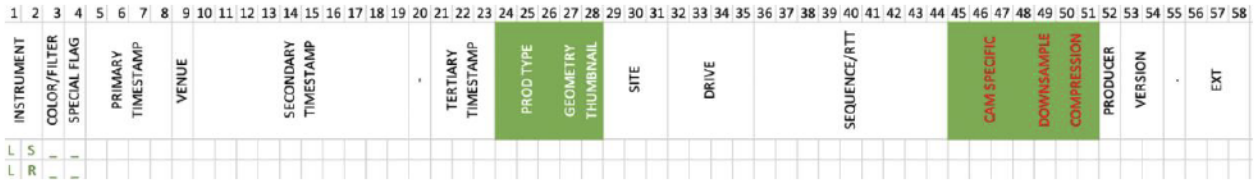


Field	Position (size, type)	Description

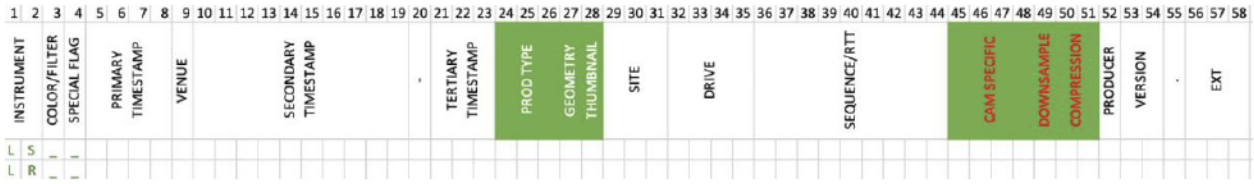
Values	Range
000, 001, …, 999	0 thru 999
A00, A01, …, A99	1000 thru 1099
B00, B01, …, B99	1100 thru 1199
…	…
Z00, Z01, … Z99	3500 thru 3599
AA0, AA1, …, AA9	3600 thru 3609
AB0, AB1, …, AB9	3610 thru 3619
…	…
ZZ0, ZZ1, …, ZZ9	10350 thru 10359
AAA, AAB, …, AAZ	10360 thru 10385
ABA, ABB, …, ABZ	10386 thru 10411
…	…
ZZA, ZZB, …, ZZZ	27910 thru 27935
0AA, 0AB, …, 0AZ	27936 thru 27961
0BA, 0BB, …, 0BZ	27962 thru 27987
…	…
7CA, 7CB, …, 7CZ	32720 thru 32745
7DA, 7DB, …, 7DV	32746 thru 32767
_ _ _	Value out of range

Drive	32 (4, i/a)	Drive identifier: Drive count (position within a Site location) from the RMC where the data was acquired.

Values	Range
0000, 0001, …, 9999	0 thru 9999
A000, A001, …, A999	10000 thru 10999
B000, B001, …, B999	11000 thru 11999
…	…
Z000, Z001, … Z999	35000 thru 35999
AA00, AA01, …, AA99	36000 thru 36099
AB00, AB01, …, AB99	36100 thru 36199



Field	Position (size, type)	Description
		… … AZ00, AZ01, …, AZ99 — 38500 thru 38599 BA00, BA01, …, BA99 — 38600 thru 38699 BB00, BB01, …, BB99 — 38700 thru 38799 … … LJ00, LJ01, …, LJ35 — 65500 thru 65535 _ _ _ _ — Value is out of range
Seq-ID/RTT	36 (9, i/a)	Identifies sequence-ID or Round-Trip Tracking token (RTT). • Sequence-ID : Identifier indicating the command sequenced the image was acquired from. Specific values will be assigned by the uplink team. • RTT : Unique identifier used for tracking activities. Used in filename for PIXL ONLY. Note: All cameras will have an RTT associated with their images but not included in the filename. PIXL is a special case where RTT and PMC are required for filename uniqueness.
Camera specific	45 (4, i/a)	Camera specific identifier. This will be "____" for non-imaging EDRs, except for SuperCam, where it will be used for point information "_###" where ### is a three digit number. For SuperCam Cruise, the value will be hardcoded as "_001" Each instrument has a unique format: ECAM Tile — STT_ ECAM Reconstructed — S0DR MCZ — SZZZ SCAM — _NNN PIXL MCC — PPPP SHERLOC MEDA Skycam EDL Cameras HELI — S000



Field	Position (size, type)	Description
		Format key: • D : Single-resolution or multi-resolution downsampling (ECAM Reconstructed images only). Multi-resolution value represents the highest resolution tile(s) of the reconstructed image.

Valid values	Description
A	Single resolution (Alone)
L	Multi-resolution 4x4 (Low-res)
M	Multi-resolution 2x2 (Med-res)
N	Multi-resolution 1x1 (No downsample)

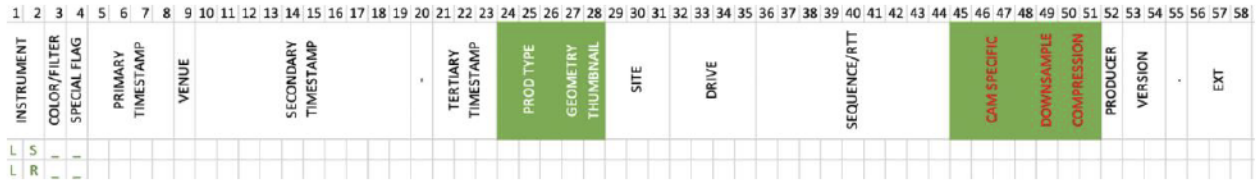
- NNN : Observation counter (RMI only). Counter increments by one for each image command used in an RMI sequence (raster). Valid values are 000-999.
- PPPP : PIXL motion counter. Valid values .
- R : Reconstruction counter (ECAM Reconstructed only). When a new tile is added that changes either the geometry or color of the reconstructed image, the counter increments by one. Valid values are 0-9, then A-Z.
- S : Stereo partner counter. Details are .

All EDRs are "_".

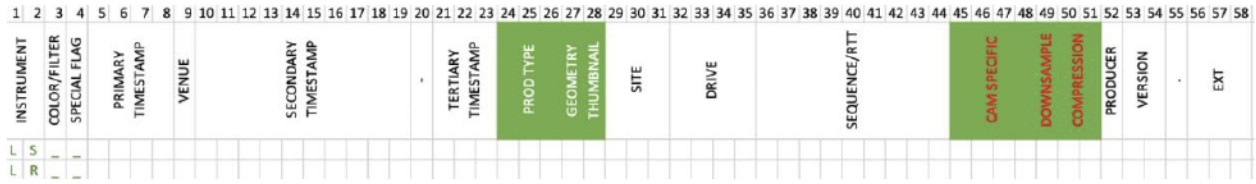
Valid values	Description
_	Nominal (synchronized) stereo or No nominal stereo partner
A-Z then 0-9	Details

- TT: Tile counter (ECAM Tile only). Identifies a unique tile within a reconstructed image. Valid values are 00-99.
- ZZZ : Focus value in millimeters (MCZ only). Valid values are 000-999.
- _ : Literal underscore. As last character of camera specific field, identifies an ECAM Tile.
- 0 : Literal zero. Undefined field placeholder.
- **SuperCam non-imaging uses the three N fields in SNNN for the point number from the command arguments DPO in the data product.**

Instrument teams with undefined ("0") fields may define values at a later date.



Field	Position (size, type)	Description
Downsample	49 (1, i/a)	Downsample resolution identifier. This will be "_" in non-imaging modes unless downsample is feasible. This value (n) indicates the level of downsampling applied to the image by the following equation: Resolution = 2^n x 2^n Valid values / Resolution: 0 — 1x1 1 — 2x2 2 — 4x4 3 — 8x8 … — ... For the case of reconstructed ECAM images, this value represents the highest resolution tile(s) in the reconstructed image (least amount of downsampling).
Compression	50 (2, i/a)	Compression type identifier. This will be "__" in non-imaging modes There are several modes of compression available, varying per instrument. The uplink team will decide which algorithm will provide the best results on a per sequence basis. For ECAM reconstructed images, compression value represents the compression level of the best tile(s) in the reconstruction. Type / Valid values / Description: JPEG (lossy) — 00 — Thumbnail 01-99 — Jpeg quality level A0 — Jpeg quality level 100 ICER (lossy) — I1, I2, …, I8 — 1 bpp, 2 bpp, …, 8 bpp I9 — Anything higher than 8 bpp Lossless — LI — ICER LL — LOCO LM — Malin LU — Uncompressed
Producer	52 (1, a)	Identifier for the institution/team that created this product:



Field	Position (size, type)	Description
		• J : JPL (IDS/MIPL) • P : Principal investigator of instrument. **Instrument / PI** ECAM — JPL MCZ — ASU (Tempe, AZ) SCAM RMI — IRAP (France) PIXL MCC — JPL SHERLOC — JPL MEDA Skycam — Ministry of Education and Science (Spain) EDL Cameras — JPL HELI RTE/NAV — JPL • A – I, K – O, Q – Z : Co-I to be identified per instrument at the discretion of the instrument PI. • _ : undefined/other Other producer codes will be added in the future.
Version	53 (2, a)	Product version number. Increments by one whenever a previously generated file with an otherwise identical filename exists. **Values / Range** 00, 01, 02 …, 99 — 0 thru 99 A0, A1, …, A9 — 100 thru 109 AA, AB, …, AZ — 110 thru 135 B0, B1, B2 …, B9 — 136 thru 145 BA, BB, …, BZ — 146 thru 171 … — … Z0, Z1, …, Z9 — 1000 thru 1009 ZA, ZB, …, ZZ — 1010 thru 1035 _ _ — Value is out of range

Field	Position (size, type)	Description
		Every version need not exist. E.g. version 01, 02, and 04 may exist but not 03. In general, the highest-numbered version represents the best version of that product. This field increments independently of all fields.
.	55 (1, a)	Separator for filename and extension. Always set to "."
Extension	56 (3, a)	File extension: • VIC : Used for VICAR files. • IMG : Same as .VIC with ODL label. • TIF : TIFF formatted image file. (no label) • JPG : JPEG formatted image file. (no label) • PNG : PNG formatted image file. (no label) • TXT : ASCII text file. • iv : Inventor-format file • ht : Height-map file (VICAR format) • rgb : Skin file in SGI RGB format • obj: Mesh file in Wavefront OBJ format • mtl : OBJ Material file • png : Lower case for meshes • mlp : MeshLab project file • xml : PDS4 label file • DAT: binary data file • fits: G6.3+ special extension for SuperCam non-imaging FITS files. Note that this is the single file type with 59 characters. Note: other file extensions to be added in the future

6.1.3 SuperCam CDR Filename Specifics

The SuperCam CDR Filename will capture many elements of the EDR filename. An important addition to the CDR filename will be the Target (name of the target of observation). This will be a maximum 20 character entry with '_' (underscore) to pad the filename if the target is less than 20 characters. If the target name is longer than 20 characters it is truncated in the filename. Each CDR will be complemented with an XML PDS4 label.

Table 6-4 SuperCam CDR File Naming Scheme

Field	Contents	Description
1-4	SCAM	Instrument
5		Underscore for readability
6-9	Sol number	Sol (Martian day)
10	_	Underscore for readability
11-20	SCLK	Spacecraft clock count, whole number of seconds
21	_	Underscore for readability
22-24	SCLK fraction	Spacecraft clock count, fractional seconds
25	_	Underscore for readability
26-28	Product type	SuperCam CDR product type.
29	_	Underscore for readability
30-38	Sequence	Sequence number
39	_	Underscore for readability
40-59	Target name	Name of target of observation, padded with underscores if needed
60	_	Underscore for readability
61-62	Point number	Point number in raster, or 01 for single point
63	P	Producer: "P" for Principal Investigator
64-65	Version	Version number
66-70	.fits	Extension

Example CDR Filename:
SCAM_0298_0625302894_968_CL0_SCAM15219_TargetName20charslen_01P01.fits

The SuperCam Microphone wav file will have a different filename format.

Table 6-5 SuperCam Microphone CDR File Naming Scheme

Field	Contents	Description
1-5	ASCAM	Instrument
6		Underscore for readability
7-13	Sol number	Sol (Martian day) preceded by "SOL"
14	_	Underscore for readability

15-24	SCLK	Spacecraft clock count, whole number of seconds
25	_	Underscore for readability
26-28	SCLK fraction	Spacecraft clock count, fractional seconds
29	_	Underscore for readability
30-32	Product type	SuperCam CDR product type.
33	_	Underscore for readability
34-42	Sequence	Sequence number
43	_	Underscore for readability
44-63	Target name	Name of target of observation, padded with underscores if needed
64	_	Underscore for readability
65-66	Point number	Point number in raster, or 01 for single point
67	P	Producer: "P" for Principal Investigator
68-69	Version	Version number
70-73	.wav	Extension

Example Microphone wav Filename:
ASCAM_SOL0092_0675108131_916_CA0_scam03092_Neeznaa_scam_____10P01.wav

6.1.4 SuperCam DDR Filename Specifics

SuperCam RMI Mosaics DDR
The filename convention is described in the following table:

Table 6-6 SuperCam RMI CDR File Name Scheme

Field	Contents	Description
1-5	ASCAM	Instrument
6		Underscore for readability
7-13	Sol number	Sol (Martian day) preceded by "SOL"
14	_	Underscore for readability
15-23	Sequence	Sequence number
24	_	Underscore for readability
25-27	MOS	Mosaic tag
28	_	Underscore for readability
29-32	FR08	Resolution and bit scale (Full resolution 8 bit)
33	_	Underscore for readability
34-36	Image order	AFT = Last image on top or FUS = Fused.
37		Underscore for readability
38-40	RGB	RGB color model Image enhancement
41	_	Underscore for readability
42	L	Indicated LIBS science product
43	R or V	R = Raman V= VIO+TRANS
44	I	Indicates IR spectrometer science product
45	_	Underscore for readability
46-65	Target name	Name of target of observation, padded with underscores if needed (20 characters) or truncated if filename longer than 20 characters
66		Underscore for readability
67	P	Producer : "P" for Principal Investigator
68-69	Version	Version number
70-73	.png	Extension

Example RMI Mosaic Filename:
ASCAM_SOL0092_scam02092_MOS_FR08_AFT_RGB_L___Naadiin_scam_____P01.png
ASCAM_SOL0092_scam02092_MOS_FR08_FUS_RGB_L___Naadiin_scam_____P01.png

6.2 PDS Standards

The M2020 camera instrument EDR data product complies with Planetary Data System standards for file formats and labels, as specified in the PDS Standards Reference [Section 1.3]. See Section 3.2 for a description of the PDS Label and the specific conventions adopted by M2020.

6.3 Time Standards

The EDR ODL labels use keywords containing time values. Each time value standard is defined according to the keyword description. See Appendix C.

6.4 Coordinate Frame Standards

The M2020 Frame Manager defines several dozen coordinate frames, which can be used for commanding pointing among other things. This section describes in detail the subset of Frames used by the products and processes in this SIS. The INSTRUMENT_COORD_FRAME_ID label (a command echo) is the only place in this SIS where the full set of frames can appear.

A subset of these frames needed for a specific image or data set are defined by the *_COORDINATE_SYSTEM groups.

6.4.1 Rover Navigation (Rover Nav) Frame

The Rover Nav frame (RNAV) is the one used for surface navigation and mobility. By definition, the frame is attached to the rover, and moves with it when the rover moves while on the surface. X is aligned on the rover's geometric center with positive toward the rover front, and Y is aligned with the middle wheels' rotation axis (center) and positive toward the starboard (right side facing forward) for the deployed rover and suspension system on a flat plane. Z is positive toward the rover nadir. The (0,0,0) origin is defined to be at the nominal surface, which is a fixed position with respect to the rover body. The actual surface will likely not be at exactly Z=0 due to the effects of suspension sag, rover tilt, rocker bogie angles, etc. The +X axis points to the front of the rover, +Y to the right side, and +Z down (perpendicular to the chassis deck). See Figure 14 for a diagram.

Note that the PLACES database maintains both telemetered and re-localized versions of the Site and Rover Nav frames at every available index.

Table 6-7 Coordinate Frames used for M2020 Surface Operations

Frame Name (Label Keyword Value)	Short Name (SAPP FDD)	Reference Frame (Used to Define)	Coordinate Frame	
			Origin	Orientation
ROVER_NAV_FRAME	RNAV	Enclosing SITE_FRAME	Attached to rover	Aligned with rover
ROVER_MECH_FRAME	RMECH	Enclosing SITE_FRAME	Attached to rover	Aligned with rover
LOCAL_LEVEL_FRAME	LL	Enclosing SITE_FRAME	Attached to rover (coincident with Rover Nav Frame)	North/East/Nadir

Frame Name (Label Keyword Value)	Short Name (SAPP FDD)	Reference Frame (Used to Define)	Coordinate Frame	
			Origin	Orientation
SITE_FRAME	SITE(n)	Previous SITE_FRAME	Attached to surface	North/East/Nadir
RSM_HEAD_FRAME	RSM_HEAD	ROVER_NAV_FRAME	Attached to mast head	Aligned with pointing of mast head. This corresponds to RSM_HEAD in the Frame Manager
Arm Frames: ARM_BASE_FRAME ARM_TURRET_FRAME ARM_DRILL_FRAME ARM_GDST_ FRAME ARM_TOOL_FRAME ARM_WATSON_FRAME ARM_SHERLOC_FRAME ARM_PIXL_FRAME	Arm Frames: RA_BASE TURRET DRILL GDRT TOOL WATSON SHERLOC PIXL	ROVER_NAV_FRAME	Attached to the tool	Aligned with tool

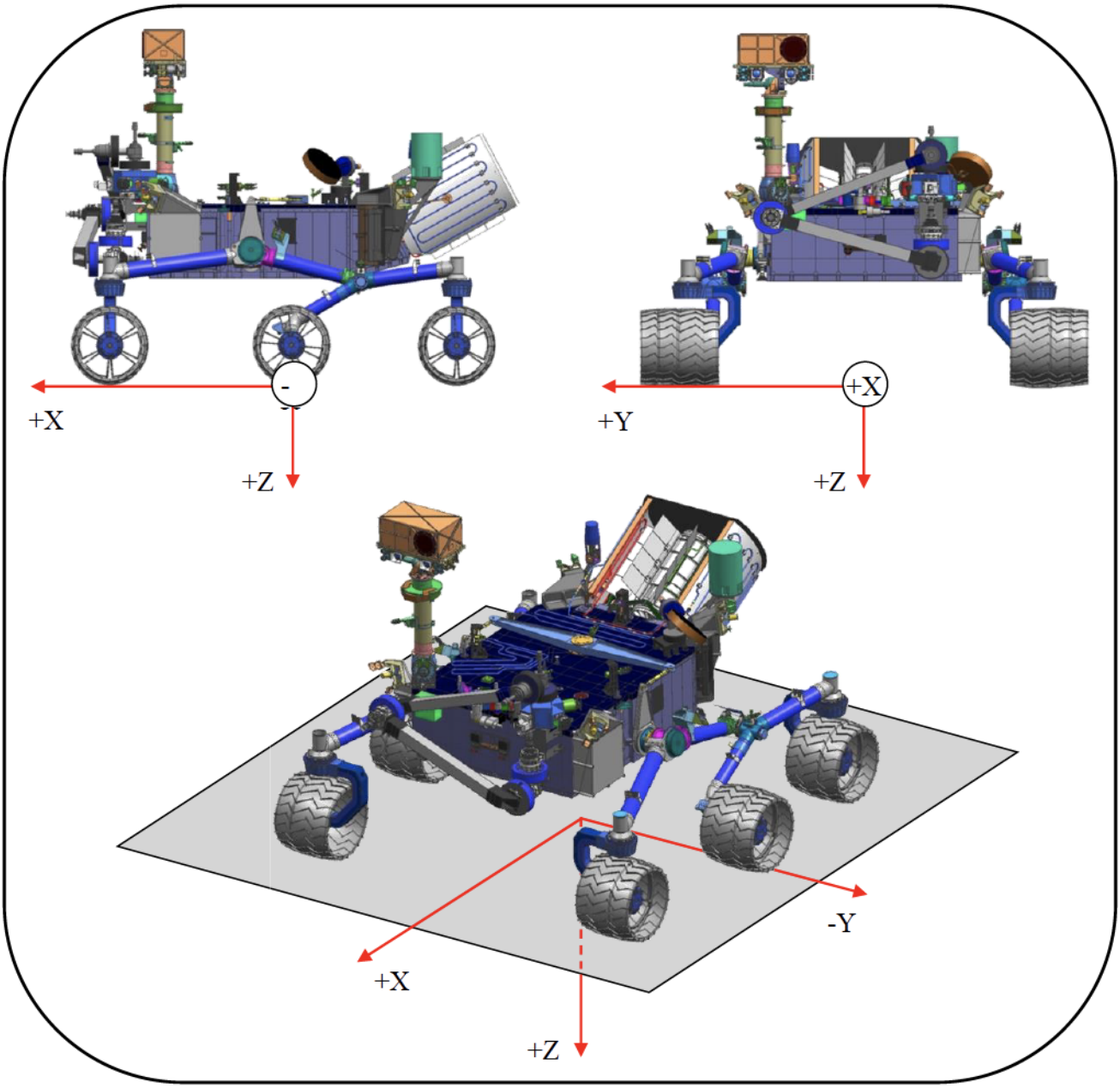


Figure 12 Rover Navigation (RNAV) Coordinate Frame

The Rover Nav frame is specified via an offset from the current Site frame, and a quaternion that represents the rotation between the two. A new instance of the Rover Nav frame, with a potentially unique offset/quaternion, is created every time the ROVER_MOTION_COUNTER increments.

Orientation of the rover (and thus Rover Nav) with respect to Local Level or Site is also sometimes described by Euler angles as shown in Figure 14, where ψ is heading, θ is attitude or pitch, and ϕ is bank or roll.

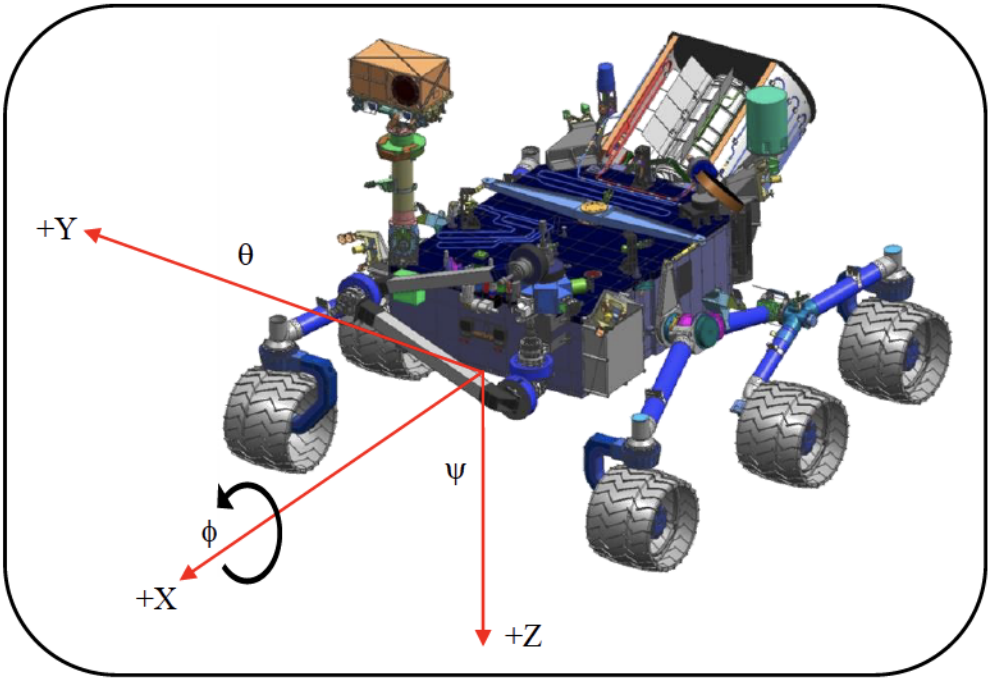


Figure 13 Yaw, Pitch, and Roll Definitions

6.4.2 Rover Mechanical (Rover Mech) Frame

The Rover Mechanical (RMECH) frame is oriented identically to the Rover Nav frame. The origin is forward of Rover Nav by x=0.09002 meters. In other words, given a point expressed in Rover Mech, if you add (0.09002, 0.0, 0.0) you will get the same point expressed in Rover Nav. Rover Mech is not used by any nominal products (EDR or RDR) but could appear in certain special products, generally having to do with arm kinematics.

6.4.3 Local Level Frame

The Local Level frame is coincident with the Rover Nav frame, i.e. they share the same origin at all times. The orientation is different, however. The +X axis points North, +Z points down to nadir along the local gravity vector, and +Y completes the right-handed system. Thus the orientation matches the orientation of Site frames.

Local Level frames are defined by an offset from the current Site frame, with an identity quaternion.

6.4.4 Site Frame

Site frames are used to reduce accumulation of rover localization error. They are used to provide a common reference point for all operations within a local area. Rover Nav and Local Level frames are specified using an offset from this origin. When a new Site is declared, that becomes the new reference, and the offset is zeroed. In this way, long-term

localization error is relegated to the offset between Sites, becoming irrelevant to local operations, because the positions are reset with each new Site.

When a Site frame is declared, it is identical to the Local Level frame, sharing both orientation and position. However, the Site frame is fixed to the Mars surface; when the rover moves, Local Level moves with it but Site stays put. Therefore, for the Site frame, +X points North, +Z points down to nadir along the local gravity vector, and +Y completes the right-handed system.

Sites are indexed, meaning there are multiple instances. Site 1 by definition represents the landing location. New Sites are declared as needed during operations, as the rover moves away from the local area. See Figure 15.

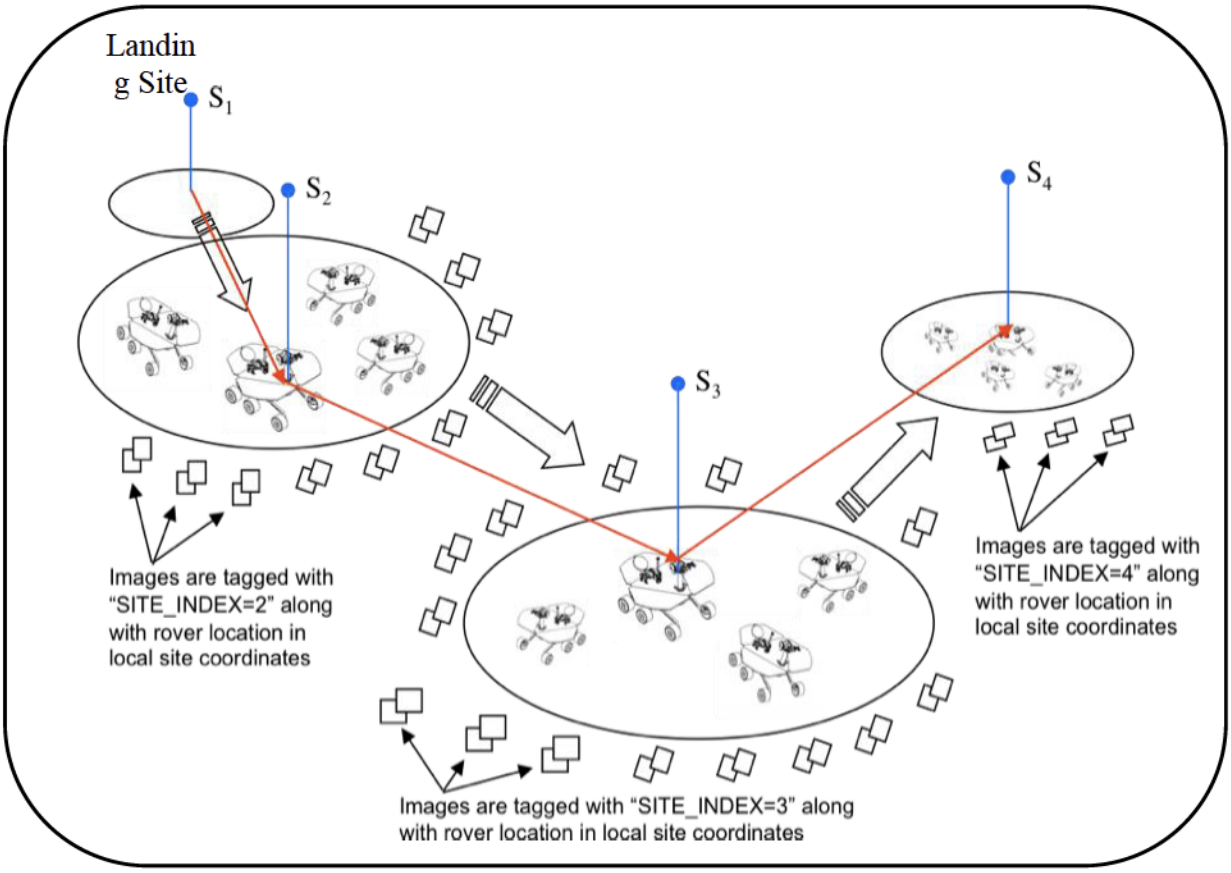


Figure 14 Site and Rover Frame Examples. Figure shows how new coordinate system origins (i.e. "Site", designated above as S_1 through S_4) are designated as the rover traverses.

The PLACES database [Ref 10] stores the set of all site-to-site offsets; such offsets are not in every image label.

6.4.5 RSM Frame

The RSM frame is attached to the Remote Sensing Mast (RSM) camera head, and moves with it. See the PPPCS for specific definition. It is expressed as an offset and quaternion from the Rover Nav frame.

6.4.6 Arm Frames

The frame representing the currently selected arm tool is reported in the arm coordinate system group. The selected tool, given by ARTICULATION_DEV_INSTRUMENT_ID, is arbitrary for any given image and may be surprising; for example SHERLOC-WATSON may not be the selected tool for a SHERLOC-WATSON image. The various tool frames are attached to and aligned with the tool in some manner specific to that tool. See the PPPCS for actual frame definitions.

7 Applicable Software

The instrument data downlink processing software is focused on rapid reduction, calibration, and visualization (in the case of images) of products in order to make discoveries, to accurately and expeditiously characterize the geologic environment around the rover, and to provide timely input for operational decisions concerning rover navigation and Arm target selection. Key software tools have been developed at JPL as part of the IDS and APSS subsystems, and at LANL/IRAP/CNES by the SuperCam team. These toolsets can be used to process data to yield substantial scientific potential in addition to their operational importance.

7.1 Utility Programs

Table 7-1 lists (in no particular order) the primary software tools that will be used to process and manipulate downlinked M2020 instrument payload data. Instrument data processing software executed by teams working the IDS and APSS subsytems at JPL will be capable of reading and writing image and spectra data in PDS format. Within IDS, the "M2020edrgen" program will generate EDRs and the Mars Program Suite of VICAR programs will generate RDRs in PDS format. An IDS pipeline system will deliver the products to M2020's OCS as rapidly as possible after receipt of telemetry. Note that non-image data products for the most part will not use the image-based utility programs, but may use new programs not yet defined.

Table 7-1 Key Software Toolsets

Name	Description	Primary Development Responsibility
SuperCam Ops Software	LIBS software development is primarily performed in IDL code, with LANL (New Mexico) as lead institute.	Dot Delapp (LANL)
	RMI software development is primarily performed in IDL code, with IRAP (France) as lead institute.	David Baratoux (IRAP)
M2020edrgen	Fetches the Data Product Object (DPO) records from M2020 Data Product (DP) files, reconstructing the file from the telemetry data into a PDS-labelled EDR data product.	Alice Stanboli, Costin Radulescu(JPL / MIPL)
Mars Program Suite	Stereo image processing software using EDRs or calibrated images (RDRs), image mosaicking software, 3-D terrain building software. VICAR code: • MARSCAHV – Generates a geometrically corrected version of the EDR, applying the C, A, H and V camera model vectors. • MARSRAD – Generates a radiometrically corrected image from a single input EDR. • MARSJPLSTEREO – Generates a disparity map from a stereo pair of input EDRs, applying a 1-D correlator (fast). • MARSCOR3 – Generates a disparity map from a stereo pair of input EDRs, applying a 2-D correlator (more robust). • MARSXYZ – Generates an XYZ image from an input disparity map.	Bob Deen (JPL / MIPL)

Name	Description	Primary Development Responsibility
	• MARSRANGE – Generates a range image from an input XYZ map. • M2020REACH – Generates an arm reachability map from an input XYZ map. • MARSINVERTER – Generates inverse lookup table (ILUT) products. • MARSDEBAYER – Generates de-Bayered images. • M2020ROUGH – Generates roughness maps. • MARSERROR – Generates XYZ and range error maps. • MARSSLOPE – Generates slope maps. • MARSBRT – Generates brightness/contrast correction file for mosaic processing. • M2020FILTER – Generates XML file for image mask files. • MARSFILTER – Generates image mask files. • MARSMASK – Applies image mask files to image files. • MARSDISPCOMPARE – Checks consistency for left-to-right and right-to-left stereo image correlations. • MARSUVW - Generates a surface normal image, wherein XYZ is computed normal to the surface. • MARSMAP – Generates a Cylindrical, Polar or Vertical projection mosaic from a list of input EDRs. • MARSMOS – Produces pinhole camera mosaics using uncorrected input images and CAHVOR camera model. • MARSMCAULEY – Generates a combination Cylindrical-Perspective projection mosaic from a list of input EDRs. • MARSTIE – Generates pointing corrections (tiepoint file) from an overlapping set of input EDRs. • MARSNAV – Generates an updated azimuth and elevation file based on comparison with existing image data that can be directly compared. • XVD – De facto image reader software capable of displaying VICAR-labeled image files.	
	• CRUMBS – 3-D terrain building software	Oleg Pariser (JPL / MIPL)
APSS / RSVP	Visualization, planning, and sequence generation software for use by Sequence Team to create Sol sequences based on activity lists generated by PSI during planning meetings. Java, C and C++ code.	Brian Cooper (JPL)

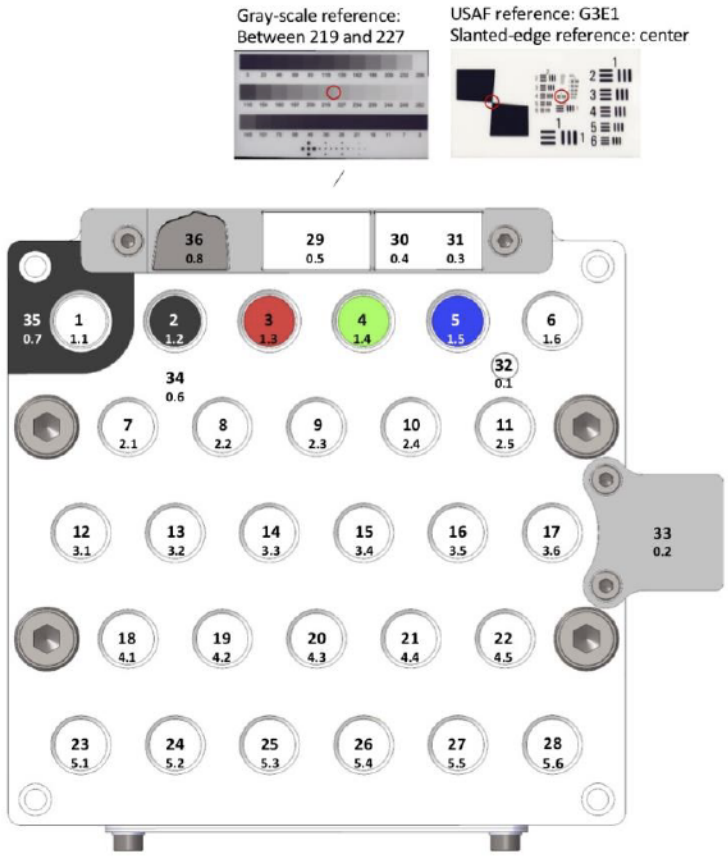
7.2 Applicable PDS Software Tools

PDS-labeled images and tables can be viewed with the program PDS4 Viewer, developed by the PDS and available for a variety of computer platforms from the PDS web site http://sbndev.astro.umd.edu/wiki/PDS4_Viewer. A Python library of PDS4 tools, from which the PDS4 Viewer is built, is available at http://sbndev.astro.umd.edu/wiki/Python_PDS4_Tools. There is no charge for this software.

7.3 Software Distribution and Update Procedures

The Mars Image Processing Program Suite is available to researchers and academic institutions. Refer to the MIPL Web site at http://www-mipl.jpl.nasa.gov for contact information.

APPENDIX A – CALIBRATION TARGETS



#	Type	Name	Position
1	IR White	IR White	1.1
2	IR black	IR black	1.2
3	Red	Red	1.3
4	Green	Green	1.4
5	Cyan	Cyan	1.5
6	Ertalyte1	Organic01	1.6
7	TSRICH	TSRICH0404	2.1
8	Chert	LCMB0006	2.2
9	Calcite CA53	LCA530106	2.3
10	Ferrosilite	PMIFS0505	2.4
11	Apatite G	TAPAG0205	2.5
12	Orthose	PMIOR0507	3.1
13	Clinopyroxene	PMIDN0302	3.2
14	Olivine	PMIFA0306	3.3
15	Andesine	PMIAN0106	3.4
16	Enstatite	PMIEN0602	3.5
17	TSERP	TSERP0102	3.6
18	BHVO-2	LBHVO20406	4.1
19	JSC-1	LJSC10304	4.2
20	Ankerite	LANKE0101	4.3
21	Siderite	LSIDE0101	4.4
22	JMn-1	LJMN10106	4.5
23	NTE01	NTE010301	5.1
24	NTE02	NTE020106	5.2
25	NTE03	NTE030106	5.3
26	NTE04	NTE040106	5.4
27	NTE05	NTE050301	5.5
28	SHERG	SHERGO02	5.6
29	Gray scale	Gray scale	0.5
30	Slanted edge	Slanted edge	0.4
31	USAF	USAF	0.3
32	Diamond	Diamond	0.1
33	Ti Plate	Ti	0.2
34	Rover paint	White paint	0.6
35	Black paint	Black paint	0.7
36	SNC	NWA 10170	0.8

APPENDIX B – Data Product Names (same as APID Names)

SuperCam Sources for EDR Types

- ScamScidata (all non-imaging science)
- Engineering types include

ScamUtilTest
ScamFault
ScamSoh
ScamBUSohPowerOn
ScamBUSohPowerOff
ScamMUSohPowerOn
ScamMUSohPowerOff
ScamBUSohThermalOn
ScamBUSohThermalOff
ScamMUSohThermalOn
ScamMUSohThermalOff
ScamSohSunSafe
ScamPoint

ScamBUMemDump
ScamMUMemDump
ScamDebugDump

APPENDIX C – Product Label Keyword Definitions, Values, Sources

As described in the main text, there are three types of label keywords: VICAR, ODL, and PDS4. The VICAR and ODL labels are virtually identical and are referred to here collectively as "VICAR" labels.

This Appendix describes several tables that will be useful for understanding the details of these keywords. All of the tables are in separate files within the document collection, with names as described herein.

PDS4 Keyword Tables

This set of tables describes the PDS4 keywords (classes and attributes in PDS 4 parlance). They include pointers to the matching VICAR keywords, as well as both the generic (multimission) definition of the keyword, and the specific "nuance" or supplemental information that applies only to Mars 2020.

These tables are created by examining a set of sample labels (incorporating all types of products being created) in order to determine the PDS4 classes and attributes that are actually used by the products described in this SIS. This list is then cross-referenced against the PDS4 data dictionaries in order to find the definitions, children, valid values, and data types. This list is then augmented with "property maps" that provide the Mars 2020-specific valid values, and the "nuance" definitions.

These tables are thus much more useful for most purposes than looking at the PDS4 data dictionaries directly, because they contain *only* the keywords that are *actually* used.

The first column contains the name of the PDS4 attribute (keyword) or class (container), and the PDS4 dictionary it comes from. Along with that, when applicable, are the VICAR keyword and property name(s) from which the values are derived. The property name is the section of the VICAR label. Not every entry has a VICAR keyword; some entries are merely containers, others contain constants or values that are derived in other ways. Some of the VICAR keywords refer to the class rather than the attribute; for example a VICAR vector keyword will typically refer to the vector's class rather than the x,y,z attributes individually.

The second column contains the definition. There are two components to many definitions, as alluded to above. The first, which is always present, is the standard PDS4 definition that applies to all missions, from the PDS4 data dictionary. The second (in italics) is a Mars 2020-specific nuance to the definition, providing additional context that applies specifically to Mars 2020.

The third column is broken up into several pieces. The first is the XPath. This gives the "path" of where the item can be found in the label, tracing the hierarchy from the root (often but not always Product_Observational) down to the item itself. Each level in this hierarchy is a hyperlink, which can be clicked on to go directly to that items' definition.

Underneath the XPath is a field whose content varies based on the type. For attributes ("keyword"), this contains the valid values, when such are defined either by the PDS4 data dictionary or the Mars 2020-specific property maps. For classes (containers), the valid children are listed. Those that are blue hyperlinks are actually used by Mars 2020; clicking on them will go to that item's definition. Those that are not blue are defined by the PDS4 data dictionary but are not used by Mars 2020.

Finally, also underneath the XPath field is another column containing (for attributes only) the data type and units. All attributes should have a data type, but only some have units defined.

These label tables are the primary source of information regarding the metadata in the labels. The rest of this document describes things at a high level; the label tables (along with the ops label table, above) define specifically what each label item means.

There are two versions of the table: sorted by PDS4 name, and sorted by VICAR keyword. The tables can be used in either direction. Given a label item you don't understand, you can look it up in the table (sorted by either PDS or VICAR name, depending on which you're looking at) to find the definition. In some cases you may need to go up the hierarchy to find a meaningful definition (for example the definition for "x" is not particularly useful, but the parent or grandparent should describe what the full x,y,z value is being used for. Alternatively, given an item in the table, you can find the item in the label by following the XPath – looking down the hierarchy of elements until you find the item. Note that not all keywords are in any given label; the table encompasses image products, browse images, mosaics, meshes, calibration, and documentation files.

The cross-reference between PDS4 name and VICAR keyword can also be useful for comparing values across similar missions (MSL, MER, Phoenix, etc) that use PDS3 (the VICAR and PDS3 keywords are generally the same).

Each of the tables is provided in both HTML and PDF format. The files are:

```
Mars2020_Supercam_Labels_sort_pds.pdf
Mars2020_Supercam_Labels_sort_pds.html
Mars2020_Supercam_Labels_sort_vicar.pdf
Mars2020_Supercam_Labels_sort_vicar.html
```

APPENDIX D – Body Unit and Mast Unit SOH

These are the data fields in the BU and MU SOH tables. Refer to the PDS4 data product labels for definitions.

BU SOH	BU_SOH converted
'00_tlm format version (cmd dict version)'	
'01_last BIST result'	
'02_current status flags'	
'03_boot bank selected'	
'04_boot bank actual'	
'05_commands received'	
'06_commands completed'	
'07_commands rejected'	
'08_number of error log entries since boot'	
'09_bank A kernel calculated checksum - NV'	
'10_bank A application calculated checksum - NV'	
'11_bank B kernel calculated checksum - NV'	
'12_bank B application calculated checksum - NV'	
'13_rce comm side'	
'14_spacewire comm error count - NV'	
'15_last spacewire error status register'	
'16_single bit MRAM memory errors - NV'	
'17_single bit SDRAM memory errors - NV'	
'18_last single bit memory address'	

'19_software version'	
'20_PULSES_ID_REG'	
'21_ROVER_HSS_TX_WD_CNT_REG'	
'22_IRQ_CLR_REG'	
'23_ROVER_HSS_CNTRL_REG'	
'24_ROVER_HSS_TX_REG'	
'25_IRQ_EN_REG'	
'26_UART_TX_DATA_CNTRL_REG'	
'27_UART_CNTRL_REG'	
'28_LVPS_CTRL_REG'	
'29_MRAM_CNTRL_REG'	
'30_LED_REG'	
'31_INT_TIME_REG'	
'32_NSPS_REG'	
'33_LZR_REP_RATE_REG'	
'34_SPW_CTRL_REG'	
'35_TEST_CTRL_REG'	
'36_MU_HSS_EMU_REG'	
'37_NUM_SPECTRA_REG'	
'38_HVPS_DLY_REG'	
'39_HVPS_DUR_REG'	
'40_STATUS0_REG'	
'41_IRQ_STAT_REG'	
'42_ROVER_HSS_FRAME_STAT_REG'	
'43_UART_STAT_REG'	
'44_ROVER_HSS_STAT_REG'	
'45_SPW_STAT_REG'	
'46_CNDH_PCB_TEMP'	
'47_CNDH_1_2_V'	'C46_CNDH_PCB_TEMP'

'48_CNDH_LVPS_TEMP'	'C47_CNDH_1_2_V'
'49_CNDH_3_3_V'	'C48_CNDH_LVPS_TEMP'
'50_CNDH_VUI_UV_SPECT_TEMP'	'C49_CNDH_3_3_V'
'51_CNDH_NEG_15_V'	'C50_CNDH_VUI_UV_SPECT_TEMP'
'52_CNDH_15_V'	'C51_CNDH_NEG_15_V'
'53_CNDH_1_5_V'	'C52_CNDH_15_V'
'54_TEST_STAT_REG'	'C53_CNDH_1_5_V'
'55_IDENTITY_STAT_REG'	
'56_TEST2_STAT_REG'	
'57_MISSED_PULSE_REG'	
'58_SE_TX_TEMP'	
'59_SE_VIO_TEMP'	'C58_SE_TX_TEMP'
'60_SE_UV_TEMP'	'C59_SE_VIO_TEMP'
'61_SE_TM_TEMP'	'C60_SE_UV_TEMP'
'62_SE_24_V'	'C61_SE_TM_TEMP'
'63_SE_HVPS_REAR'	'C62_SE_24_V'
'64_SE_HVPS_CURRENT'	'C63_SE_HVPS_REAR'
'65_SE_HVPS_TEMP'	'C64_SE_HVPS_CURRENT'
'66_number of commands sent to MU'	'C65_SE_HVPS_TEMP'
'67_MU commands ACK"d by MU'	
'68_MU commands NAK"d by MU'	
'69_MU packets received success'	
'70_MU packets received error'	
'71_number of BU spectra read out - NV'	
'72_number of RMI received from MU - NV'	
'73_number of microphone recordings - NV'	
'74_number of science laser shots commanded - NV'	
'75_current absolute focus table position'	
'76_close limit switch position difference'	

'77_far limit switch position difference'	
'78_current autofocus baseline position'	
'79_current spacewire link 0 status register'	
'80_current spacewire link 1 status register'	
'81_current spacewire link 2 status register'	
'82_current spacewire link 3 status register'	
'83_current focus table position in mm'	
'84_reserved'	
'85_status_flags'	
'86_motor speed inhibited'	
'87_laser current inhibited'	
'88_CCD is clocking'	
'89_limit swithch arg changed to computed value'	
'90_focus motor inhibited'	
'02a_sys_error'	
'02b_send_evr'	
'02c_shutdown'	
'02d_sun_safe'	
'02e_bu_hvps_on'	
'02f_mu_shutter'	
'02g_mu_laser_on'	
'02h_mu_dc_chain_on'	
'02i_mu_on'	
'02j_ccd_almost_ready'	
'02k_ccd_ready'	
'02l_irs_ready'	
'02m_laser_doubler_ready'	
'02n_laser_stack_ready'	
'02o_cwl_ready'	

MU SOH	MU SOH converted
'00_Laser_osc1_Temp'	'C00_Laser_osc1_Temp'
'01_Laser_osc2_Temp'	'C01_Laser_osc2_Temp'
'02_Laser_doub1_Temp'	'C02_Laser_doub1_Temp'
'03_Laser_doub2_Temp'	'C03_Laser_doub2_Temp'
'04_Laser_15V'	'C04_Laser_15V'
'05_Laser_HV'	'C05_Laser_HV'
'06_Laser_Limiter_Temp'	'C06_Laser_Limiter_Temp'
'07_DPU_Motor_Driver_Temp'	'C07_DPU_Motor_Driver_Temp'
'08_DPU_FPGA_Temp'	'C08_DPU_FPGA_Temp'
'09_Vref_+2.5V'	'C09_Vref_+2.5V'
'10_+1.5V'	'C10_+1.5V'
'11_+3.3V'	'C11_+3.3V'
'12_Motor_Temperature'	'C12_Motor_Temperature'
'13_Primary_Mirror_Temp'	'C13_Primary_Mirror_Temp'
'14_CWL_Pd_Current'	'C14_CWL_Pd_Current'
'15_Micro_Temp'	'C15_Micro_Temp'
'16_CWL_Temp'	'C16_CWL_Temp'
'17_+3.3_IRS'	'C17_+3.3_IRS'
'18_+20V_RF_IRS'	'C18_+20V_RF_IRS'
'19_+5V_RF_IRS'	'C19_+5V_RF_IRS'
'20_-5V_RF_IRS'	'C20_-5V_RF_IRS'
'21_+5V_DT_IRS'	'C21_+5V_DT_IRS'
'22_-5V_DT_IRS'	'C22_-5V_DT_IRS'
'23_LVPS_Board_Temp'	'C23_LVPS_Board_Temp'

'24_Primary_Current'	'C24_Primary_Current'
'25_IRS_HK_Temp_AOTF'	'C25_IRS_HK_Temp_AOTF'
'26_IRS_HK_TEMP_HOT_TEC'	'C26_IRS_HK_TEMP_HOT_TEC'
'27_IRS_HK_Temp_board'	'C27_IRS_HK_Temp_board'
'28_IRS_HK_Temp_DET'	'C28_IRS_HK_Temp_DET'
'29_Remaining laser pulses'	'C29_Remaining laser pulses'
'30_ON/OFF shutter'	
'31_ON/OFF limiter'	
'32_ON/OFF High voltage 15 V'	
'33_ON/OFF Microphone'	
'34_ON/OFF RMI'	
'35_ON/OFF DC/DC'	
'36_ON/OFF heater doubler laser'	
'37_ON/OFF heater stack laser'	
'38_ON/OFF heater CWL'	
'39_Heater_Doubler_Received'	
'40_Heater_Stack_Received'	
'41_Heater_CWL_Received'	
'42_ON/OFF TEC_IR'	
'43_TEC_IR ready'	
'44_IRS_meas_ongoing'	
'45_IR_Det_sel'	
'46_Motor driver circuit thermal’	
'47_AF gain status_MSB'	
'48_AF gain status_LSB'	
'49_ON/OFF CWL'	
'50_ON/OFF CWL modulation'	
'51_ON/OFF motor driver'	
'52_Infinite limit switch status'	

'53_Close Limit switch status'	