# Introduction

## Scope

The Energetic Particle Detector (EPD) suite consists of four sensors measuring electrons, protons, and ions from helium to iron, and operating at partly overlapping energy ranges from 2 keV up to 200 MeV/n. The EPD sensors are:

* SupraThermal Electrons, Ions, & Neutrals (STEP)
* Suprathermal Ion Spectrograph (SIS)
* Electron Proton Telescope (EPT)
* High Energy Telescope (HET)

The EPD sensors share the Instrument Control Unit (ICU) that is composed by the Common Data Processing Unit and the Low Voltage Power Supply (CDPU/LVPS), which is the sole power and data interface of EPD to the spacecraft. In order to improve the reliability, the ICU is a redundant system working in a cold redundant configuration. This means that nominal or redundant ICU units are never switched ON simultaneously. CDPU shares information with the STIX, MAG and SWA investigation to allow synchronized high data rate burst-mode operations following on-board identification of predefined triggering events in the EPD data. The CDPU is designed to manage sensors control and monitoring, the sensors timing clock, and sensor data collection, compression, and packetization for telemetry. CDPU is also responsible of the S/C telecommand reception and delivery, if necessary, to the sensors. All sensors have their front-end electronics co-located with the instrument.

STEP consists of a single unit having two view cones in opposite directions. SIS consists of two sensor heads with roughly opposite (160°) view directions sharing a common electronics box. EPT-HET has multiple view cones sharing a common electronics box. There are two identical EPT-HET units.

The overall energy coverage achieved with the EPD sensors is 0.002 MeV to 20 MeV for electrons, 0.003 MeV to 100 MeV for protons, 0.008 MeV/n to 200 MeV/n for heavy ions (species-dependent). This energy and species coverage well satisfies and for a large part exceeds the requirements defined for EPD in the Solar Orbiter Payload Definition Document and in the report of the Joint Science and Technology Definition Team (JSTDT) for the Solar Orbiter/Sentinels mission.

# Applicable Documents

AD1 EID-A:Solar Orbiter Experiment Interface Document, Part A. SOL-EST-RCD-0050 issue 5. 03/2015

AD2 EID-B: Solar Orbiter EPD Experiment Interface Document, Part B. SO-EPD-PO-IF-0001 issue 3 rev 3

AD3 Interface Control Document (ICD). SO-EPD-ICU-IF-0001 issue 4 rev 0

AD4 Solar Orbiter Operations Requirements Document. SO-ESC-RS-05001 issue 1 rev 8

AD5 Software. ECSS-E-ST-40C. 06/03/2009

AD6 Solar Orbiter (SOLO) Energetic Particle Detector (EPD) Instrument Control Unit (ICU). SO-EPD-ICU-DD-004 issue 5 rev 0

AD7 EPD Telemetry and Telecommand ICD. SO-EPD-PO-IF-0003 issue 2 rev 9

AD11 Requirements and clarifications on software development and qualification for Solar Orbiter instruments. SOL-EST-RS-3188 issue 1 rev 0

AD12 EPD Sensors Data Interface Control Document. SO-EPD-PO-IF-0005 issue 1 rev 6

# Reference Documents

RD1 CCSDS Packet Telemetry. CCSDS 102.0-B-5. Blue Book. CCSDS 102.0-B-5. 11/2000

RD2 Ground systems and operations - Telemetry and telecommand packet utilization. ECSS-E-70-41A. 30/01/2003

# Terms, Definitions and Abbreviations

AD Applicable Document

ASW Application Software

BSW Boot Software

C&DH Control and Data Handling

CDPU Common Data Processing Unit

EGSE Electrical Ground Support Equipment

EPD Energetic Particle Detector

GSS Ground Support System

ICU Instrument Control Unit

IRQ Interrupt Request

FPGA Field Programmable Gate Array

LVPS Low Voltage Power Supply

OBDH On-Board Data Handling

RAM Random Access Memory

RD Reference Document

S/C Spacecraft

SDRAM Synchronous Dynamic RAM

SVVP Field Programmable Gate Array

SW Software Verification and Validation Plan

TBC To Be Confirmed

TC Telecommand

TM Telemetry

# Software Overview

## Function Purpose

### Purpose

The EPD Software Requirements Document defines and describes the operations, interfaces, performance, and quality assurance requirements of the EPD's ICU software. The requirements described in this document are derived from [AD1](#AD1), [AD2](#AD2), [AD3](#AD3) and [AD4](#AD4) (EID-A, EID-B, Interface Control Document and Solar Orbiter Operations Requirements Document).

### Scope

This document applies to the EPD's ICU software. It does not provide information for the Ground Support Software (GSS), which is maintained separately as part of the Electronic Ground Support Equipment (EGSE). This document supplies information applicable to the Software Requirements Specification.

## Environmental Considerations

### General Description

[ICU\_IF](#ICU_IF) illustrates the top-level interfaces to the ICU, which affect the Instrument Control Unit Software.

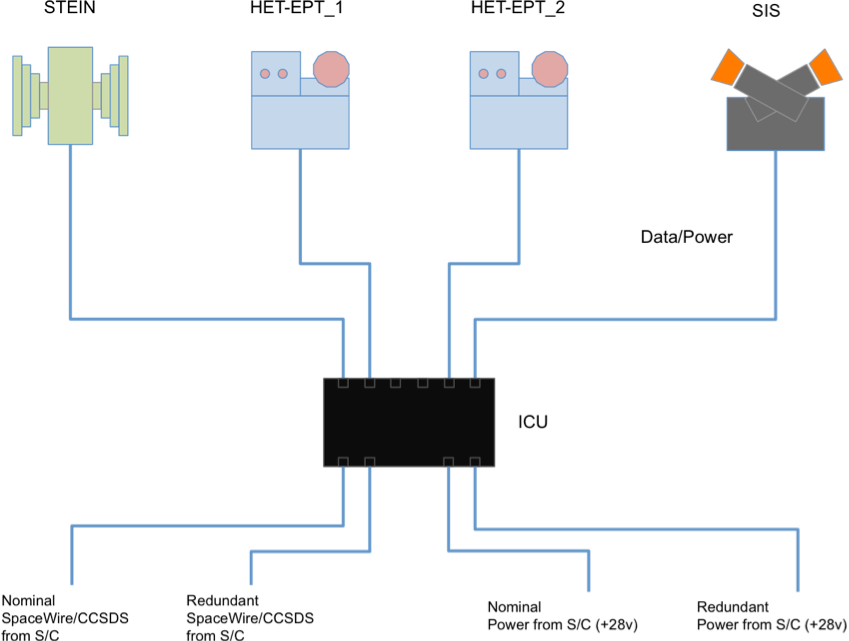


Figure 1: ICU interfaces

EPD's CDPU provides the interface between the Solar Orbiter Spacecraft C&DH system and the EPD sensors (STEP, HET, EPT and SIS). All information transfer between the EPD sensors and the Spacecraft/Ground flows through the CDPU, including telemetry, commands and status. The sensors communicate with the CDPU over a dedicated serial interface.

### Hardware Overview

The ICU is composed by the CDPU and the LVPS. Both units are redundant, working in a cold configuration. Thus we have the CDPU/LVPS nominal unit and the CDPU/LVPS redundant unit. According to the cold redundancy configuration pattern, if one unit is switched ON, the other must be OFF.[ICU\_HW](#ICU_HW)shows the basic HW diagram of the ICU.

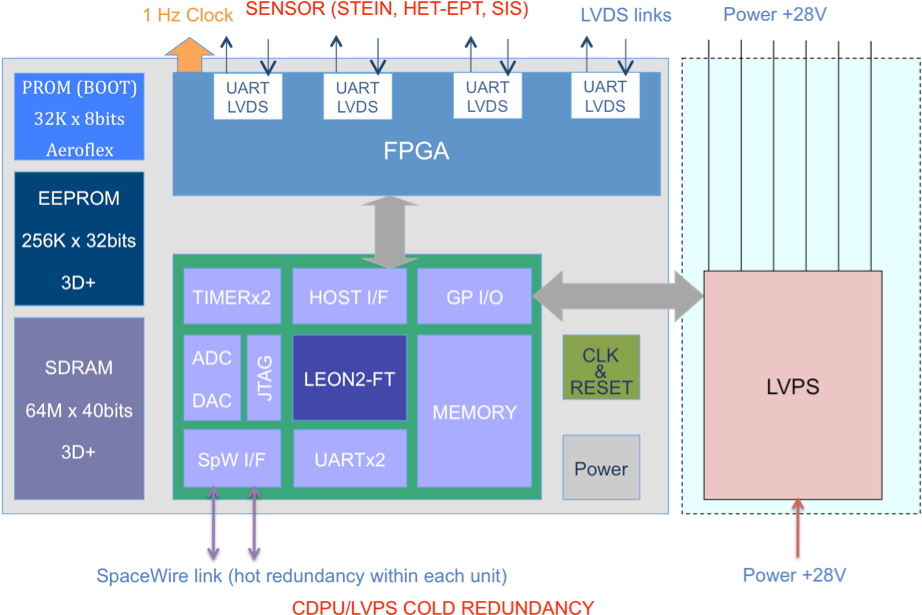


Figure 2: ICU hardware schema

As it can be observed, the design is based on a FPGA (the RTAX2000 from Actel) that includes a LEON2 soft processor, the UART-LVDS links with the sensors, the SpaceWire cores and the interfaces with the LVPS and the memory. The RAM memory uses the SDRAM 3DSD2G40VS5238MS (1.56 Gbits in a 40 bit configuration to implement the EDAC) part and the EEPROM memory uses the 3DEE8M32VS8094 (8 Mbits in a 32 bit configuration) part, both from 3Dplus.

The LVPS provides +28v non-regulated power supply to the sensors. The switching and monitoring is performed by the CDPU. In [LVPS\_HW](#LVPS_HW) a simplified diagram of the LVPS hardware is shown.

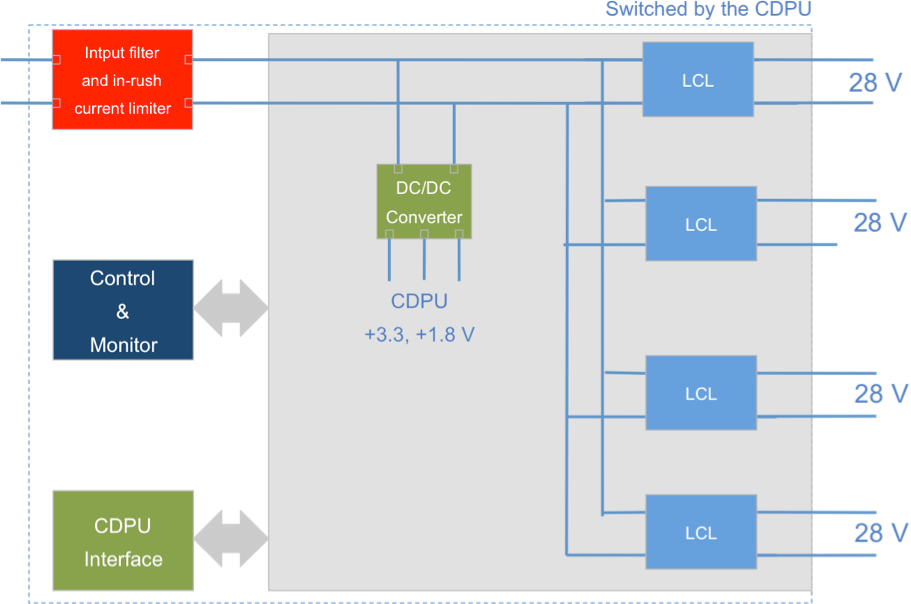


Figure 3: LVPS hardware schema

### Modes

In [ICU\_SW\_Op\_modes](#ICU_SW_Op_modes) the modes of operation of ICUSW are represented.

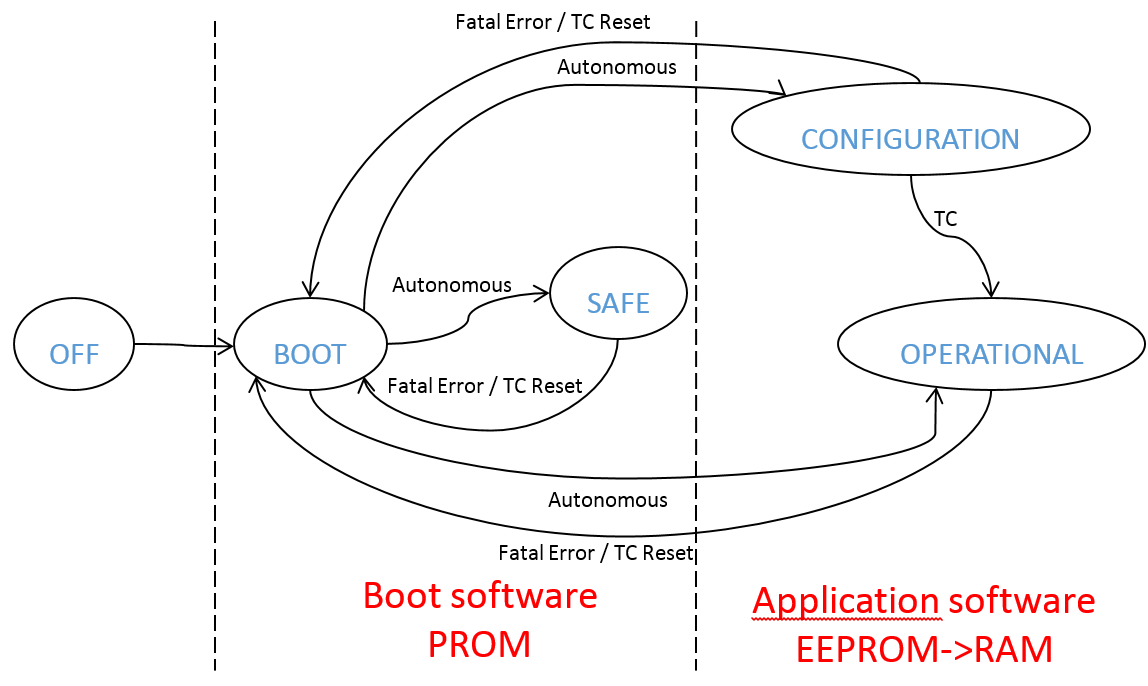


Figure 4: ICU SW operation modes

The ICUSW shall be able to support the following modes:

* OFF: The ICU is not powered.
* BOOT: The ICU is just switched on, a soft reset from other mode has been performed, or a mode transition TC has been received. This mode is in charge of the initial ICU configuration and checking. It also tries to load the application software and pass control to it. The boot software of the ICUSW shall implement the BOOT mode.
* SAFE: is a mode in which EPD is powered but in a safe configuration, which can be maintained indefinitely (e.g. main control unit powered but sensors are off). In this mode EPD will generate only non-science telemetry. The configuration of the instrument in SAFE mode will be unambiguous (i.e. only one defined configuration). The boot software of the ICUSW shall implement the SAFE mode, and it will trigger a transition from BOOT mode to SAFE mode when application software can not be executed. In SAFE mode the damaged application software images stored in EEPROM can be restored by means of patch telecommands-
* CONFIGURATION: this mode shall be used for required operations to activate or configure the instrument. In this mode non-science data may be generated, depending on the instrument configuration (e.g. sensors activated or not).
* OPERATIONAL: this mode is the mode in which EPD is fully operative and generates non-science as well as science data.

Autonomous transitions are the following:

* BOOT to CONFIGURATION. This transition is triggered autonomously after a reset when application software is correctly loaded, the target mode is CONFIGURATION, and the control is passed to it.
* BOOT to OPERATIONAL. This transition is triggered autonomously after a reset when application software is correctly loaded, the target mode is OPERATIONAL, and the control is passed to it.
* BOOT to SAFE. This transition is triggered when application software can not be loaded and the boot software remains in control of the ICU.
* SAFE to BOOT. This transition is triggered by a soft reset (fatal error detected such as bus error, memory access error, etc.) or a TC.
* CONFIGURATION to BOOT. This transition is triggered by a soft reset (fatal error detected such as bus error, memory access error, etc.), by a recovery action (minor error detected such as driver or controller error etc.) or a TC.
* OPERATIONAL to BOOT. This transition is triggered by a soft reset (fatal error detected such as bus error, memory access error, etc.), by a recovery action (minor error detected such as driver or controller error etc.) or a TC.

## Relation Other Systems

### User Interfaces

The EPD Instrument has no direct in-flight human user interfaces. The Mission Operation Center (MOC) manages all human interaction with EPD.

### Hardware Interfaces

The detailed EPD's ICU hardware to software interfaces are shown in [ICU\_HW\_IFs](#ICU_HW_IFs).

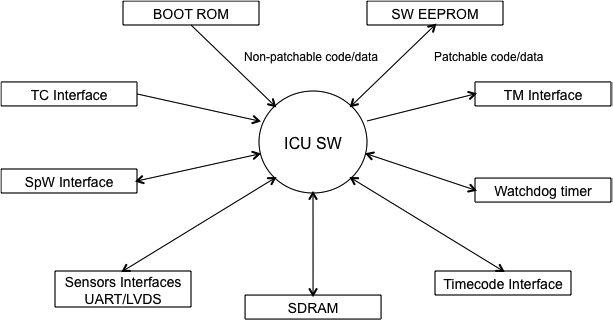


Figure 5: ICU hardware interfaces

### External Software Interfaces

SRG EPD's ICU uses the RTEMS operating system to perform task control functions and provide some resource management services.

ICU is planning on using the following capabilities of RTEMS:

A detailed list of each used function is provided in the SRG EPD ICU Software Detailed Design Document.

* Prioritized, preemptive task scheduler.
* Inter-process communication.
* Resource locks (semaphores).
* Timers.

### Communication Interfaces

#### Sensor Interface

All EPD sensors commands and telemetry are communicated via UART protocol at 115200 bauds using Low-Voltage Differential Signalling (LVDS) according to the EPD Serial Data/Command Protocol shown in [EPD\_Serial](#EPD_Serial) below. A cold redundant configuration will be used for data transmission/reception links between the CDPU and the sensors.

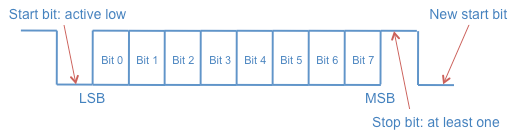


Figure 6: EPD Serial Data/Command Protocol

Data communication between the sensors and the CDPU is started every second by the CDPU that acts as a master. CDPU will provide a 1 Hz hardware clock (one pulse per second or 1PPS) to the sensors [PPS\_signal](#PPS_signal)). This 1PPS signal is generated by the CDPU and synchronized with the S/C through the time codes, according to [ECSS-E-50-12](#RD3), provided by SpaceWire. The CDPU drives the 1PPS signal directly to the sensors via a cabled hardware signal. A specific FPGA hardware entity, integrated in the CDPU, is in charge of generating the 1PPS signal synchronized with the time code reception. LVDS will be used to send 1PPS clock signal, in a cold redundant configuration, to the sensors. This approach eliminates the software latency and jitter of processor interrupt response and avoids the latency of UART transmission. Note that all the signals (transmission/reception and 1PPS clock) both, nominal and redundant are differential signals, so no common ground is needed in any case.

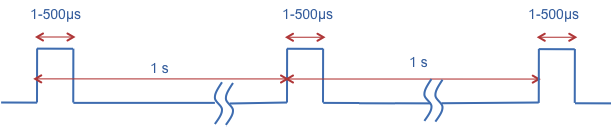


Figure 7: 1PPS signal characteristics

Data is sent over the command and telemetry interfaces as a transaction. Each transaction contains a single Sensor Transfer Frame (STF). Each STF contains command or telemetry data. The general format of the STF is shown in [STF](#STF). Command and telemetry-specific formats are described below.

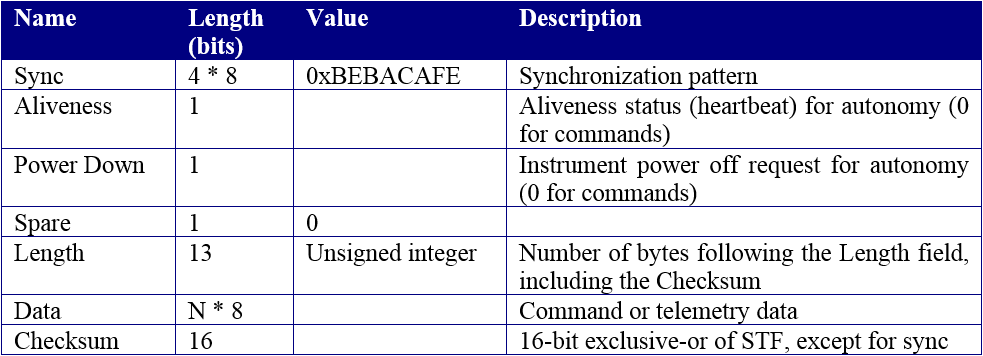


Table 1: Sensor Transfer Frame (STF) Format

#### S/C Interface

Each CDPU is a remote terminal on the spacecraft SpaceWire link. The implementation will be compliant up to the [RD2](#RD2) and [RD3](#RD3). Two separate SpaceWire links in a hot redundant configuration will be employed. SpaceWire is the command and data interface with the S/C.

EPD command messages are contained in CCSDS telecommand packets. The spacecraft forwards CCSDS telecommand packets to the EPD when they contain any of the EPD identifiers (The EPD Process IDs are defined in [AD7](#AD7), section 10). EPD CDPU strips off the CCSDS packet headers and analyzes the destination identifier. If the identifier is associated to the CDPU, the telecommand is processed directly by the CDPU. If the identifier is associated to any sensor, the CDPU forward the command to the specific sensor. Command responses from the sensor are sent to the CDPU that transmits them to the S/C in CCSDS telemetry packets format. Command responses generated by the CDPU itself are also CCSDS formatted and transmitted to the S/C.

The list of telecommands and telecommands responses are defined in [AD7](#AD7).

The data interface is used for transfer data (science & housekeeping) and command response from EPD sensors to the CDPU and from CDPU to S/C. Transfer of data and command response from sensors to CDPU shall take place according to the Serial Data/Command Protocol described in Figure 5. Transfer of data and command response from CDPU to the S/C shall take place according to the CCSDS recommendations defined in [RD1](#RD1) and [RD2](#RD2).

#### LVPS Interface

The interface between the CDPU and the LVPS is an internal interface. It is used to switch ON/OFF the sensors and to monitor voltages and currents.

### Time

The Orbiter DMS maintains a time code pattern as Spacecraft Elapsed Time (SCET).

The SCET (see section 4.8.4 of [AD1](#AD1)) is considered as the Central Time Reference (CTR) and it is used as on-board reference for the correlation of time on Ground. The Central Time Reference (CTR) is maintained at spacecraft level and distributed to the Instruments in order to synchronize instruments with the DMS and AOCS and allow instruments to time-stamp their telemetry packets. The CDPU receives the time from the S/C formatted as a CCSDS telecommand packet. The time format is spacecraft elapsed time, in seconds and fractions of a second. The received time corresponds to the next SpaceWire time code. The time packet must arrive at least 300 ms before the time code (see [AD1](#AD1): R-838).

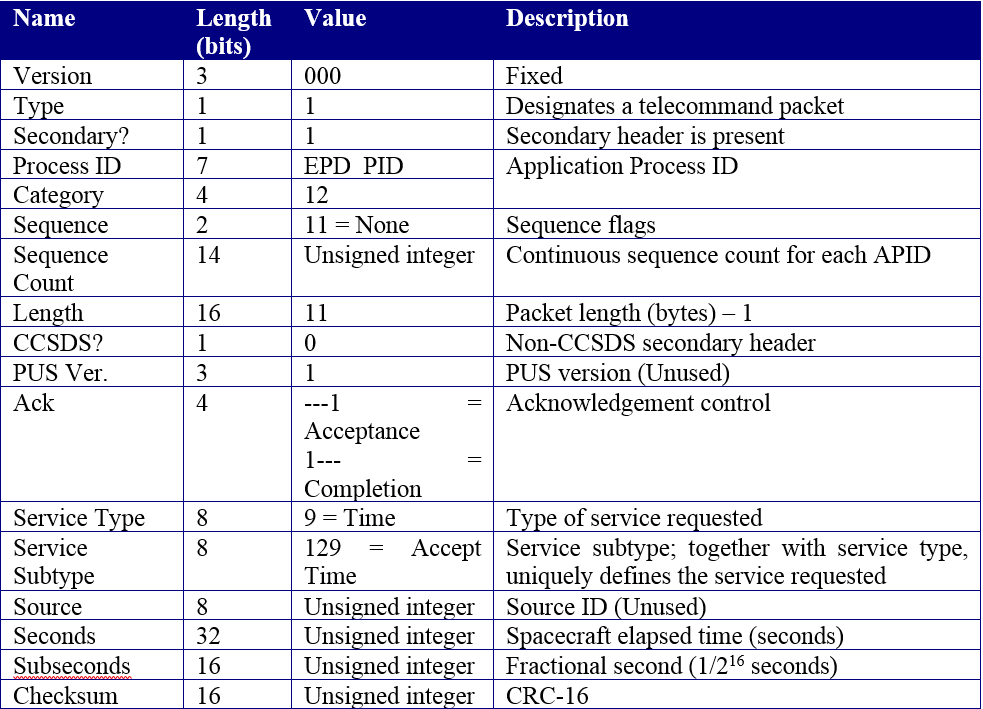


Table 2: Time Telecommand Packet Format

## Constraints

# Requirements

Each of the EPD sensors has some data processing capabilities either by software running in the sensor DPUs/microcontrollers or by logic functions. The main software, however, is running in the CDPU. The principal functions of the CDPU software are:

* To collect scientific data from the instruments periodically 1 Hz.
* To accumulate high-cadence science data from the sensors to specified lower time resolution packages.
* To collect housekeeping data from the four units.
* To time tag the data if necessary.
* To perform lossless compression of the data if necessary.
* To format the data for transmission by the S/C.
* To accept and syntactically check TC from the S/C.
* To deliver these commands to the individual sensors for execution.
* To accept from the ground and distribute to the sensors tables and software.
* To handle the burst mode.

A Real-Time Operating System is proposed to provide the basic facilities such as multitasking, timing and semaphore/mutex synchronization. A high-level software development tool (UML2, ROOM or similar) could be employed in the software design process.

All software will comply with the requirements defined in the section 4.9.1 of the [EID-A](#AD1), namely:

1. All on-board software will comply with the software standard ECSS-E-40 by [AD5](#AD5)tailored by[AD11](#AD11).
2. Different areas of memory will be employed to store code, fixed constants and variable parameters of the in-flight software.
3. The PROM memory will include minimum boot software able to manage EEPROM updates and hardware test functions.
4. Independent software module memory updates will be supported.
5. Blocking the instrument due to S/W maintenance will be avoided. In case of blocking it can be cleared by a power cycling of the instrument.
6. Instrument safety hazard will be avoided in the software design process.
7. On-board software will acquire, store and telemeter the employed resources, including memory usage, central processor unit (CPU) usage and I/O usage.
8. Before enabling the on-board software a mechanism to verify its integrity will be provided.
9. A single telecommand will be used to enable the on-board software.
10. Telecommand and telemetry source packets will be the only means to communicate ground and on-board software.
11. If a processor reset is detected, an event report will be generated.
12. If a processor overload condition is detected, an event report will be generated.
13. If an unexpected arithmetic overflow condition is detected, an event report will be generated.
14. If an illegal program instruction is encountered during execution of a program code, an event report will be generated.
15. If a data bus error is detected by software, an event report will be generated.
16. If a memory corruption is detected by an EDAC, an event report will be generated.
17. If a checksum error is detected, an event report will be generated.
18. A modularized structure will be employed in the software design process.

The ICU shall acquire science data from the sensors every 1 s. The information provided by every sensor shall be stored in a ring buffer of 10 minutes length with 1 s resolution. Every sensor unit will have its own independent ring buffer. According to [AD12](#AD12) 9.5.2, after a programmable number of seconds the received packets with 1 s resolution shall be transformed in one packet. This transformation requires some specific data integration that depends on every sensor. The integrated packet shall be time-tagged with the time of the last integrated packet.

Burst trigger is a per-sensor process, not all EPD sensors shall be in burst mode at the same time. The burst can be triggered internally (by an EPD sensor as is defined in [AD12](#AD12)] 10.3), that it will be reported to the spacecraft by a PUS service 3 Telemetry, or externally (by events detected in other(s) Solar Orbiter instrument). The EPD can associate the external trigger of Burst mode to the reception of a specific data on the PUS CCSDS service 20 packet according to [AD12](#AD12) 10.2. This packet will be sent from spacecraft to the EPD at a maximum rate of 8Hz.

The ICUSW shall also manage a specific set of telemetry generated by the sensor units called Quick Look Data Telemetry. This telemetry will be forwarded to the S/C using a different APID.

The transmission rate of the Bust mode telemetries and Quick Look Data Telemetry shall be programmed specifically by means of table stored in EEPROM as is defined in [AD12](#AD12) 9.5.2.

In the following paragraphs a lists of software requirements will be included. Every single requirement has the following format:

| Name | SRS-ICUSW XX-YYYY | Modes |
| Description | Requirement Description | |
| Validation | Validation Approach | |
| Parent | Requirement Traceability | |

**SRS-ICUSW XX-YYYY:** Identifies the requirement. XX is the type of requirement and YYYY is the number used to identify the requirement. Types of requirements are:

* **GE.** General
* **FU.** Functional
* **PE.** Performance
* **IF.** Interface
* **OP.** Operational
* **RS.** Resources
* **DI.** Design and Implementation Constraints
* **SE.** Security
* **PO.** Portability
* **QU.** Quality
* **RE.** Reliability
* **MA.** Maintainability
* **SA.** Safety
* **CD.** Configuration and delivery
* **DD.** Data definition and database
* **AI.** Adaptation and installation

Examples:

* SRS-ICUSW FU-0025
* SRS-ICUSW PE R-0037

**Modes:** establishes the modes in which the requirement is available, namely:

* **F.** Off
* **B.** Boot
* **S.** Safe
* **C.** Configuration
* **O.** Operational

**Description:** is the text that describes the requirement.

**Validation approach:** establishes how the requirement is verified:

* **R.** Review
* **A.** Analysis
* **I.** Inspection
* **T.** Test
* **S.** Similarity

**Parent:** contains the list of "software specification" that are cover by the requirement.

For each requirement type, subsections are provided when requirements apply only to boot software or to application software or they are common.

Other subsections or extended descriptions can be also added in order to better organize and clarify the requirements.

Extended descriptions will be added using **D:**. Example:

D: The ICU ASW shall handle the telecommand and telemetry according to the following requirements:

## General Requirements

### Common software requirements

|  |  |  |
| --- | --- | --- |
| **Name** | **GE R-00010** | F, B, S, C, O |
| **Description** | The ICUSW shall comply with the requirements regarding Onboard Processors, Software and Memory Management as specified in section 2.3.2 and Section 3.6 of [AD4](#AD4). | |
| **Validation Method** | Testing | |
| **Parent** | R-406 | |

|  |  |  |
| --- | --- | --- |
| **Name** | **GE R-00030** | F, B, S, C, O |
| **Description** | The ICUSW shall ensure that EPD complies to the associated TM-/TC-Packet Services specified as mandatory for the payload in the Operations Interface Requirements Document [AD4](#AD4)chapter 3. These services are:   * Telecommand Verification service * Housekeeping and Diagnostic Reporting service * Event Reporting service * Memory Management service * Time Management service * Test service * Information Distribution service * Science Data Transfer service | |
| **Validation Method** | Testing | |
| **Parent** | R-180 | |

## Functional Requirements

### Common software requirements

|  |  |  |
| --- | --- | --- |
| **Name** | **FU R-00180** | S, C, O |
| **Description** | The ICUSW shall provide the Telecommand Verification service (service 1) to allow adequate and unambiguous verification of acceptance, progress (where applicable) and execution of all telecommands sent from any source (sent from Ground for immediate, delayed or time-tagged execution, and sent from onboard applications). | |
| **Validation Method** | Testing | |
| **Parent** | TM-4  R-366 | |

|  |  |  |
| --- | --- | --- |
| **Name** | **FU R-00390** | S, C, O |
| **Description** | The ICUSW shall provide an "are you alive" function for testing the end-to-end connection between Ground or Command & Control Function and itself. This function implements the Telecommand "Perform Connection Test", service 17,1; and response Telemetry "Connection Test Report", service 17,2. | |
| **Validation Method** | Testing | |
| **Parent** | FTS-1  FTS-2 | |

### Application software requirements

|  |  |  |
| --- | --- | --- |
| **Name** | **FU R-00370** | C, O |
| **Description** | The ICUSW shall provide the service 3,129 in order to specify the frequency of generation of a specified housekeeping telemetry packet via Telecommand "Update HK Report Generation Period". | |
| **Validation Method** | Testing | |
| **Parent** | TM-4  R-366 | |

The frequency accepted by the ICUSW must be <= 1Hz.

|  |  |  |
| --- | --- | --- |
| **Name** | **FU R-00380** | C, O |
| **Description** | The ICUSW application software shall provide the services 3,5 and 3,6. | |
| **Validation Method** | Testing | |
| **Parent** | PERP-12 | |

Using these services, it will be possible to enable/disable by telecommand the generation of an existing housekeeping packet (and for more than one packet as part of the same command). Telecommands "Enable HK Parameter Report Generation", service 3,5; and "Disable HK Parameter Report Generation", service 3,6.

## Performance Requirements

## Interface Requirements

### Common software requirements

|  |  |  |
| --- | --- | --- |
| **Name** | **IF R-00150** | S, C, O |
| **Description** | The ICUSW shall implement the Telecommand Verification Service (service 1) of the PUS standard according to the requirements TCV-1 to TCV-7, stated as such in the Solar Orbiter Operations Requirement Document [AD4](#AD4). | |
| **Validation Method** | Testing | |
| **Parent** | TCV-1  TCV-2  TCV-3  TCV-4  TCV-5  TCV-6  TCV-7 | |

The telemetries related to this requirement is specified in the EPD Telemetry and Telecommand Document [AD7](#AD7).

his requirement does not apply to telecommand addressed to SIS.

|  |  |  |
| --- | --- | --- |
| **Name** | **IF R-00160** | F, B, S, C, O |
| **Description** | The ICUSW shall implement the Housekeeping and Diagnostic Data Reporting Service (service 3) according to the requirements PERP-1 to PERP-3 stated as such in the Solar Orbiter Operations Requirement Document [AD4](#AD4). | |
| **Validation Method** | Testing | |
| **Parent** | PERP-1  PERP-2  PERP-3 | |

For each ICUSW mode, the telemetries related to this requirement are specified in the EPD Telemetry and Telecommand Document [AD7](#AD7).

|  |  |  |
| --- | --- | --- |
| **Name** | **IF R-00170** | F, B, S, C, O |
| **Description** | The ICUSW shall implement the Event Reporting Service (service 5) according to the requirements EVRP-2, EVRP-3, EVRP-4, EVRP-5 and EVRP-7, stated as such in the Solar Orbiter Operations Requirement Document [AD4](#AD4). | |
| **Validation Method** | Testing | |
| **Parent** | EVRP-2  EVRP-3  EVRP-4  EVRP-5  EVRP-6  EVRP-7 | |

The complete set of events and telemetries related to this requirement is specified in the EPD Telemetry and Telecommand Document [AD7](#AD7).

|  |  |  |
| --- | --- | --- |
| **Name** | **IF R-00180** | F, B, S, C, O |
| **Description** | The ICUSW shall implement support for event based reporting telemetry packets. For anomaly reports, 3 levels of criticality (low, medium and high) shall be distinguished. Telemetries "Nominal/Progress Report", service 5,1; "Error/Anomaly Report - Low Severity", service 5,2; "Error/Anomaly Report - Medium Severity", service 5,3; Error/Anomaly Report - High Severity", service 5,4. | |
| **Validation Method** | Testing | |
| **Parent** | EVRP-1 | |

The assignment of events to the different service 5,x telemetries is specified in the EPD Telemetry and Telecommand Document [AD7](#AD7).

|  |  |  |
| --- | --- | --- |
| **Name** | **IF R-00390** | S, C, O |
| **Description** | The ICUSW shall implement a connection test service (Service 17) according to the specifications given chapter 3 of | |
| **Validation Method** | Testing | |
| **Parent** | R-261 | |

This test service implies that the ICUSW will answer with a service 17,2 telemetry packet to a telecommand of service 17,1.

## Operational Requirements

## Resources Requirements

### Application software requirements

The ICUSW application software will handle the information about EPD status, housekeeping and fault detection and recovery management according to the following design requirements:

|  |  |  |
| --- | --- | --- |
| **Name** | **RE R-00070** | C, O |
| **Description** | The ICUSW failure detection functions shall control the continuous production of service 5 report packets corresponding to the same anomaly by means of autonomously disabling individually the report packet generation after each reported anomaly.  The service 5,5 and 5,6 telecommands will be used to control the report packet generation. | |
| **Validation Method** | Testing | |
| **Parent** | R-264  FDIR-14  R-407  EVRP-6 | |

The service 5 telemetry for anomaly reports is defined in the EPD Telemetry and Telecommand Document [AD7](#AD7).

The service 5,5 and 5,6 telecommand that controls the report packet generation is defined in the EPD Telemetry and Telecommand Document [AD7](#AD7).

## Design Requirements

### Application software requirements

The ICUSW application software will handle the information about EPD status, housekeepingand fault detection and recovery management according to the following design requirements:

|  |  |  |
| --- | --- | --- |
| **Name** | **DI R-00030** | C, O |
| **Description** | The application software of the ICUSW will provide the EPD status information in service 3 telemetry from direct measurements from operating units rather than from secondary effects. | |
| **Validation Method** | Testing | |
| **Parent** | R-366 | |

## Security Privacy Requirements

## Portability Requirements

## SW Quality Requirements

## SW Reliability Requirements

## SW Maintainability Requirements

## SW Safety Requirements

## SW Configuration Delivery Requirements

## Data Definition DB Requirements

## Human Factors Requirements

## Adaptation Installation Requirements

# Logical Models