HERSCHEL/PACS

SPU HIGH LEVEL SOFTWARE TO DPU
Interface Description

Document Ref.: PACS-TW-ID-001

Issue: 6.0

Prepared by: **Ahmed Nabil BELBACHIR** Date: **13 December 2005**

Checked and Approved by: **Helmut FEUCHTGRUBER**

Approved by: **Stefano PEZZUTO**

Distribution Record

Issue / Revision	Draft 0	Draft 1	Draft 2	1	1.1	1.2	2
Distribution Date	10.05.2000	06.07.2000	16.08.2000	19.01.2001	16.03.2001	23.05.2001	02.07.2001
Issue / Revision	2.1	3	3.1	3.2	3.3	3.4	4.0
Distribution Date	27.08.2001	07.09.2001	05.10.2001	23.10.2001	22.01.2002	03.04.2002	11.09.2002
Issue / Revision	4.1	4.2	4.3	4.4	4.5	4.6	5.0
Distribution Date	25.03.2003	30.04.2003	14.05.2003	28.05.2003	03.07.2003	29.07.2003	29.03.2004
Issue / Revision	5.1	6.0					
Distribution Date	16.04.2004	14.12.2005					

INTERNAL			**EXTERNAL**		
Department	**Name**	**Qty**	**Company**	**Name**	**Qty**
UVIE/ASTRO	F. Kerschbaum	1	MPE	H. Feuchtgruber	1
UVIE/ASTRO	R.Ottensamer	1			
			CSL		0
			IAC		0
			IFSI	R. Orfei, S. Pezzuto	1
			MPE	PACS Project Office (pacs@mpe.mpg.de) PACS Warm Electronic (pacs-we@ster.kuleuven.ac.be)	1
			Electronic Archives at Leuven http://pacs.ster.kuleuven.ac.be		1

Document Change Record

Document Title **HERSCHEL – PACS** **SPU HLSW to DPU Interface Description**		
Issue	**Date**	**Reason for Change**
Draft 0	09/05/2000	Initial issue
Draft 1	06/07/2000	Discussions in PACS OBS#2 + Comments
Draft 2	16/08/2000	Comments from IFSI CSL and Discussions in MPE
1	16/01/2001	Discussions with MPE and IFSI and document structure update
1.1	16/03/2001	Comments from OBS#4
1.2	22/05/2001	Comments from HLSW meeting in Vienna
2	28/06/2001	Comments from OBS#5
2.1	15/08/2001	Adapted to the new PS-ICD

3	31/08/2001	Comments from IFSI and MPE
3.1	03/10/2001	Comments from OBS#6
3.2	19/10/2001	Comments from MPE and IFSI
3.3	17/01/2002	See ECPs N°: ECP-020101, ECP-020102, ECP-020103, ECP-020104, ECP-020105, ECP-020106, ECP-020107, ECP-020108, ECP-020109, ECP-020110, ECP-020111, ECP-020110, ECP-020112, ECP-020113 and ECP-020114
3.4	19/03/2002	See ECPs N°: ECP-020301, ECP-020302, ECP-020303, ECP-020304, ECP-020305 and ECP-020306
4.0	11/09/2002	See Attached ECPs
4.1	25/03/2003	See Attached ECPs
4.2	30/04/2003	See Attached ECPs
4.3	14/05/2003	See Attached ECP
4.4	28/05/2003	See Attached ECPs
4.5	03/07/2003	See Attached ECP
4.6	28/07/2003	See Attached ECP
5.0	26/03/2004	See Attached ECPs
5.1	16/04/2004	See Attached ECPs
6.0	13/12/2005	See Attached ECPs

Contents

List of Figures

List of Tables

1. Introduction

The reduction/compression SW of the '**P**hotodetector **A**rray **C**amera **S**pectrometer' (PACS) Instrument is developed by (T)UVIE. The way the compression should be done for the science data is read from the DEC/MEC header and tables stored on board. The compressed data should be transferred to the DPU via one IEEE 1355/spacewire data link per board. AD003 describes the critical interfaces between separately developed sub-units of PACS while this document depicts the SPU to DPU interface.

1.1. Purpose of the Document

This document defines the transmission protocol between SPU HLSW and DPU. The DPU is the slave (1355 link start-up protocol), which sends commands. The SPU HLSW, which is the master (1355 link start-up protocol), acknowledges and executes the commands. Compressed/uncompressed data are sent to DPU according to DEC/MEC header and tables stored on board. Additionally, the HLSW HK parameters are sent to the DPU at a data rate compatible with the overall PACS HK.

In its present form, the document contains the protocol of DPU-SPU HLSW interface.

1.2. Acronyms and Abbreviations

ADSP	Analog device DSP
AID	Activity ID
AOCS	Attitude and Orbit Control Subsystem
ASW	Application SoftWare
BOL	BOLometers
CDMS	Command and Data Management System
CPU	Central Processing Unit
CRE	Cold Readout Electronics
CSL	Centre Spatial de Liège
DEC/MEC	Detector Controller/Mechanisms Controller
DPU	Digital Processing Unit
DRCU	Detector Readout and Control Unit
DSP	Digital Signal Processor
EEPROM	Electrically Erasable Programmable Read Only Memory
FCU	FPU Control Unit
FIRST	Far Infrared and Sub-millimeter Telescope
FPU	Focal Plane Unit
HIFI	Heterodyne Instrument for HERSCHEL (HiFi instrument)
HK	House-Keeping
HLSW	High Level SW
HS	High Speed
HSO	Herschel Space Observatory
HW	HardWare
I/F	Interface
IAC	Instituto de Astrofísica de Canarias
ID	Identification
IFSI	Istituto di Fisica dello Spazio Interplanetario
LLSW	Low Level SW
LSB	Least Significant Byte
LWL	Long WaveLength
MEM_STATUS_CNTS	MEMory STATUS CouNTerS
MFCU	Mechanisms and Focal plane electronics Control Unit
MSB	Most Significant Byte
OBCP	On-Board Control Procedure
OBS	On-Board Software
PACS	Photodetector Array Camera and Spectrometer
PHC	PhotoConductors
PMA	Program Memory Address
PMD	Program Memory Data
PRIP	Pattern Recognition and Image Processing
PROM	Programmable Read Only Memory
RAM	Random Access Memory
RCNB	Raw Channels NumBer
RCX	Raw Channel indeX
REAL	REduction ALgorithm
ROM	Read Only Memory
SAU	Smallest Addressable Unit
SID	Structure ID
SPIRE	Spectral and Photometric Imaging Receiver
SPU	Signal Processing Unit
SW	SoftWare
SWL	Short WaveLength
TBC	To Be Confirmed
TBD	To Be Defined
TBU	To Be Updated
TBW	To Be Written
TC	TeleCommand
TM	TeleMetry
TUVIE	Technical UVIE
UR	User Requirement
URD	User Requirements Document
UVIE	University of Vienna
WE	Warm Electronics

1.3. References

1.3.1. Applicable Documents

AD001	PACS-TW-GS-001	HERSCHEL/PACS SPU High Level Software Specification Document
AD002	SCI-PT-ICD-07527	Packet Structure Interface Control Document
AD003	PACS-ME-ID-001	PACS Instrument Interface Requirement Document

1.3.2. Reference Documents

RD001	BSSC(96)2	Guide to applying the ESA software engineering standards to small Software projects
RD002	DIPSAPII-DAS-31-06	SMCS332 user manual

1.4. Overview of the Document

This document is structured as follows:

- From Section 1 and Section 2, an introduction showing the SPU HLSW environment is presented.
- Section 3 contains a list of the supported DPU commands
- The DPU-SPU protocol is presented in Section 4. It consists of the Data Packet Structure (Command, Response, HK and Science Data Packet Structure).
- In Section 5, the SPU responsibility is described.

1.5. Interface Function

It consists of one 1355 data link per board running at nominal link speed (10 Mbps). The DPU-SPU HLSW communication is bi-directional.

2. General Description

The communication interface between DPU and SPU HLSW is depicted in Figure 1.

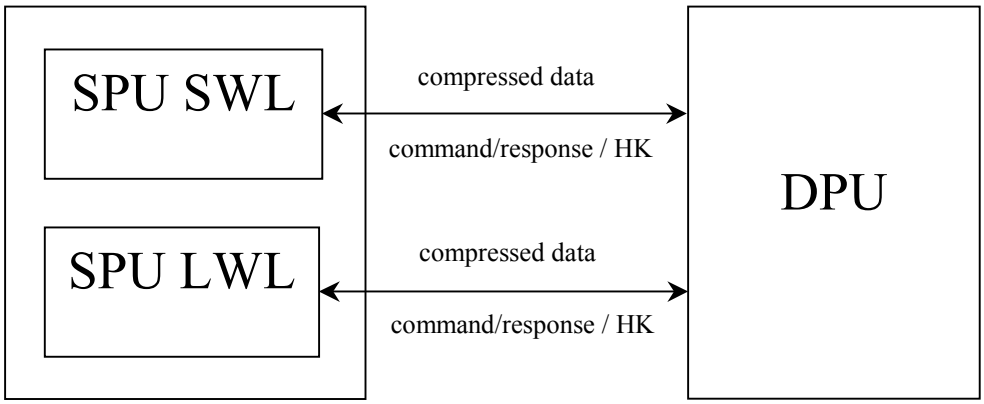

Figure 1. Communication between SPU HLSW and DPU

The SPU unit consists of two CPU boards, which work independently. They are called SWL SPU and LWL SPU. The HLSW will be identically implemented in the CPU of each SPU sub-unit (SWL and LWL). As both CPUs have equivalent functionality, we only focus on the HLSW interface to DPU of one board. The SPU contains one data input from DPU:

- Commands from DPU

and three data outputs to DPU:

- Compressed data to DPU
- HK to DPU
- Command response to DPU

The SPU HLSW is the master of the DPU-SPU HLSW connection according to the link start-up protocol described in RD002.

3. DPU Commands

The list of DPU commands to the SPU HLSW is presented in Table 1. See document AD001 for more details.

Command		Description	Comments
LOAD		Load	write data on the SPU memory
DUMP		Dump	read data from the SPU memory
CHECK		Check	check specified SPU memory locations.
Perform Activity	CP_DATA_RAM_EEPROM	Copy Data from RAM to EEPROM	burn upgraded HLSW into EEPROM
	RESET	Warm Reset	performs SPU sub-unit warm reset
	RAW_CHAN_TRAN_MODE	Raw Channel Selection	selects the number and index of raw channel
	STOP_REDUCT_COMPR	Stop	interrupts the application SW
	START_REDUCT_COMPR	Start	runs the application SW
	PEAK_UP	Peak-up	performs the Peak-up routine
	ACT_TEST_PHOT	SPU Test Mode for Photometry	generate synthetic data in photometry, compress it and send it to DPU
	ACT_TEST_SPEC	SPU Test Mode for Spectroscopy	generate synthetic data in spectroscopy, compress it and send it to DPU
	CONNECT_DMC	Connect to DEC/MEC	start the SPU link to DEC/MEC as slave or master
Write	WRT_DXSx	Write Detectors Selection Table	write the tables of the selected detectors (x = 1 .. 7)
	WRT_DET_CST_SPEC	Write Detectors Constants in Spectroscopy	write the table of the photoconductors constants
	WRT_DET_CST_PHOT	Write Detectors Constants in Photometry	write the table of the bolometer constants
	WRT_SIM_DATA	Write Simulated Data Parameters	write the table of parameters for data simulation (SPU Test Mode)

Table 1. Description of the SPU HLSW Commands

4. SPU – DPU Protocol

Four kinds of data packets are foreseen from/to SPU to/from DPU:

- Command data packets
- Response data packets (Acknowledgement of DPU commands)
- SPU HLSW HK data packets
- Science data packets (compressed/uncompressed science data + compressed DEC/MEC header)

The commands, listed in Section 3, are sent from DPU to SPU using five command packet structures (Load, Dump, Check, Perform Activity and Write), as represented below.

4.1. Command Source Packet Structure

The commands are sent from DPU through IEEE 1355/spacewire data links according to the packet structure represented by *Figure 2*. The command size should fit into a TC packet.

Command ID 2 Bytes	Data Nx2 Bytes

Figure 2. Command Packet Structure

4.1.1. Command ID

This field (2 Bytes) will be used to identify the type of DPU command, which is sent to the SPU. The SPU HLSW execution of one of them will be performed after the identification of the requested command. The command IDs are represented by Table 2.

Command	Command ID	Comments
LOAD	0x01	This command is used to write data on the SPU memory
DUMP	0x02	This command is used to read data from the SPU memory
CHECK	0x03	This command is used to check specified SPU memory locations.
Perform Activity	0x04	This command is used for several activities: Peak-Up I/F (PEAK_UP), Copy Data from RAM to EEPROM (CP_DATA_RAM_EEPROM), Warm Reset (RESET), Stop (STOP_REDUCT_COMPR), Start (START_REDUCT_COMPR), SPU Test Mode for Photometry (ACT_TEST_PHOT) and for Spectroscopy (ACT_TEST_SPEC) and Connect to DEC/MEC (DMC_CONNECT).
Write	0x06	This command is used to upgrade the parameter tables in the SPU memory (WRT_DXSx (x = 1…7), WRD_DET_CST_SPEC, WRT_DET_CST_PHOT, WRT_SIM_DATA).

Table 2. DPU Command IDs

4.1.2. Data

This field (Nx2 Bytes) is used to attach the required fields to execute the commands. The attached fields are described in the following subsections.

4.1.2.1. LOAD

Figure 3 depicts the load command data field packet structure.

Address1 2 Bytes	Address2 2 Bytes	Length 2 Bytes	Data <Length>SAUs	Checksum 2 Bytes

Figure 3. Load Command Data Field Packet Structure

- **Address1:** This field as described in *Figure 4*, contains the Memory ID (1 Byte) and the MSB of the start address.

Memory ID			
Subsystem ID (3 bits)	Memory Type (1 bit)	Block inside the Memory Type (4 bits)	MSB of the Address (1 Byte)

Figure 4. Organization of the Address1 Field

Subsystem ID: 010b for the SWL SPU, 011b for the LWL SPU and 100b for both SPU.
Memory type: 0b for the program RAM and 1b for the data RAM
Memory block: 0001b … RAM
0010b … Extended RAM
0100b … SMCS DPRAM

The MSB of the Start Address is read from the LSB of the Address1 field.

- **Address2:** This field contains the 2 LSBs of the start address
- **Length** of the data to be loaded (2 Bytes).

  Its unit is word (4 Bytes for the load to data memory (SAU = 32 bits) and 6 Bytes for the load to program memory (SAU = 48 bits)).

- **Data** to be loaded (<Length>SAUs)

  The maximum length of data that can be written with a single TC is 204 Bytes (51/34 SAUs for DRAM/PRAM) as described in AD002.

- **Checksum** that is used to verify the integrity of the data to be loaded (2 Bytes)

4.1.2.2. DUMP

Figure 5 depicts the dump command data field packet structure.

Address1 (2 Bytes)	Address2 (2 Bytes)	Length (2 Bytes)

Figure 5. Dump Command Data Field Packet Structure

- **Address1:** This field as described in *Figure 4*, contains the Memory ID (1 Byte) and the MSB of the start address
- **Address2:** This field contains the 2 LSBs of the start address
- **Length** of the data to be dumped (2 Bytes)

  Its unit is word (4 Bytes in case of data memory (SAU = 32 bits) and 6 Bytes for the program memory (SAU = 48 bits)). The maximum length of memory that can be dumped with a single TC is 65532 Bytes (16383/10922 SAUs for DRAM/PRAM) as written in AD002.

4.1.2.3. CHECK

Figure 6 depicts the check command data field packet structure.

Address1 (2 Bytes)	Address2 (2 Bytes)	Length (2 Bytes)

Figure 6. Check Command Data Field Packet Structure

- **Address1:** This field as described in *Figure 4*, contains the Memory ID (1 Byte) and the MSB of the start address
- **Address2:** This field contains the 2 LSBs of the start address
- **Length** of the data to be checked (2 Bytes)

  Its unit is word (4 Bytes in case of data memory (SAU = 32 bits) and 6 Bytes for the program memory (SAU = 48 bits)). The maximum length of memory that can be checked with a single TC is 65532 Bytes (16383/10922 SAUs for DRAM/PRAM) as written in AD002.

4.1.2.4. Perform Activity

Several activities are executed via this command. The Perform Activity command data field packet structure follows the service TC (8,4) in AD002. The Structure ID field defines the presence and the number of attached parameters. The coding of the Activity IDs and Structure IDs for the Perform Activity command is listed in Table 3.

Activity	Activity ID	Structure ID	Comments
CP_DATA_RAM_EEPROM	'00000100' (0x04)	'00000101' (0x05)	The SPU HLSW burns the upgraded HLSW into EEPROM.
RESET	'00000101' (0x05)	'00000000' (0x00)	The SPU HLSW will be reset whenever this command is received
RAW_CHAN_TRAN_MODE	'00000110' (0x06)	'00000101' (0x05)	This command is sent in SPU Stopped mode to select number and end index of detectors from which data are send in raw form
STOP_REDUCT_COMPR	'00000111' (0x07)	'00000000' (0x00)	This command is transmitted for the interruption of the application SW.
START_REDUCT_COMPR	'00001000' (0x08)	'00000000' (0x00)	The SPU HLSW will run the application SW after the reception of the start command
PEAK_UP	'00001001' (0x09)	'00000000' (0x00)	After the reception of this command the SPU HLSW will run the Peak-Up SW.
ACT_TEST_PHOT	'00001010' (0x0A)	'00000000' (0x00)	After the reception of this command the SPU HLSW will generate synthetic data (photometry), compress it and send it to DPU.
ACT_TEST_SPEC	'00001011' (0x0B)	'00000000' (0x00)	After the reception of this command the SPU HLSW will generate synthetic data (spectroscopy), compress it and send it to DPU.
CONNECT_DMC	'00010000' (0x10)	'00000001' (0x01)	This command is used to start the SPU link to DEC/MEC as slave or master.

Table 3. Identification of the Activities

4.1.2.4.1. CP_DATA_RAM_EEPROM

Figure 9 represents the Perform Activity command data field packet structure for Copy Data from RAM to EEPROM.

Spare 2 Bytes	Activity ID 2 Bytes	Structure ID 2 Bytes	Parameter1..5 5x4 Bytes

Figure 7 Copy Data from RAM to EEPROM Command Data Field Packet Structure

- **Spare** field (2 Bytes) included for the 4 bytes alignment.
- **Activity ID** field (2 Bytes) for the Copy Data from RAM to EEPROM command is '0000100' (0x04)

- **Structure ID** field (2 Bytes) for the Copy Data from RAM to EEPROM command is '00000101' (0x05)
- **Parameter1: Spare** (4 Bytes).
- **Parameter2: Memory ID for RAM** (0x01 for PRAM, 0x11 DRAM and 0x12 for Extended DRAM)
- **Parameter3: Start Address in RAM** (Start address) (4 Bytes).
- **Parameter4: Start Address in Target Memory** (Start address in EEPROM)(4 Bytes)
- **Parameter5: Length** (length in 16bit words)(4 Bytes).

4.1.2.4.2. RESET

Figure 8 represents the Perform Activity command data field packet structure for Warm Reset.

Spare (2 Bytes)	Activity ID (2 Bytes)	Structure ID (2 Bytes)

Figure 8. Warm Reset Command Data Field Packet Structure

- **Spare** field (2 Bytes) included for the 4 bytes alignment.
- **Activity ID** field (2 Bytes) for the warm reset command is '00000101' (0x05)
- **Structure ID** field (2 Bytes) for the warm reset command is '00000000' (0x00)

4.1.2.4.3. RAW_CHAN_TRAN_MODE

Figure 9 represents the Perform Activity command data field packet structure for the Raw Channel Selection.

Spare 2 Bytes	Activity ID 2 Bytes	Structure ID 2 Bytes	Parameter1..5 5x4 Bytes

Figure 9. Raw Channel Selection Command Data Field Packet Structure

- **Spare** field (2 Bytes) included for the 4 bytes alignment.
- **Activity ID** field (2 Bytes) for the Raw Channel Selection command is '0000110' (0x06)
- **Structure ID** field (2 Bytes) for the Raw Channel Selection command is '00000101' (0x05)
- **Parameter1: Spare** field (4 Bytes).
- **Parameter2: Observing Mode** (0x01 Spectroscopy, 0x02: Photometry) (4 Bytes).
- **Parameter3: RCNB** Number of Raw Channels (Range:0-31) (4 Bytes).
- **Parameter4: RCX**: Index of Raw Channels until which RCNB pixels are transmitted (4 Bytes).

- **Parameter5: Spare** field (4 Bytes).

4.1.2.4.4. STOP_REDUCT_COMPR

Figure 10 represents the Perform Activity command data field packet structure for the Stop.

Spare (2 Bytes)	Activity ID (2 Bytes)	Structure ID (2 Bytes)

Figure 10. Stop Command Data Field Packet Structure

- **Spare** field (2 Bytes) included for the 4 bytes alignment.
- **Activity ID** field (2 Bytes) for the stop command is '00000111' (0x07)
- **Structure ID** field (2 Bytes) for the stop command is '00000000' (0x00)

4.1.2.4.5. START_REDUCT_COMPR

Figure 11 represents the Perform Activity command data field packet structure for the Start.

Spare (2 Bytes)	Activity ID (2 Bytes)	Structure ID (2 Bytes)

Figure 11. Start Command Data Field Packet Structure

- **Spare** field (2 Bytes) included for the 4 bytes alignment.
- **Activity ID** field (2 Bytes) for the start command is '00001000' (0x08)
- **Structure ID** field (2 Bytes) for the start command is '00000000' (0x00)

4.1.2.4.6. PEAK_UP

Figure 12 represents the Perform Activity command data field packet structure for the Peak-up.

Spare (2 Bytes)	Activity ID (2 Bytes)	Structure ID (2 Bytes)

Figure 12. Peak-up Command Data Field Packet Structure

- **Spare** field (2 Bytes) included for the 4 bytes alignment.
- **Activity ID** field (2 Bytes) for the peak-up command is '00001001' (0x09)
- **Structure ID** field (2 Bytes) for the peak-up command is '00000000' (0x00)

4.1.2.4.7. ACT_TEST_PHOT

Figure 13 represents the Perform Activity command data field packet structure for the SPU Test for Photometry.

Spare (2 Bytes)	Activity ID (2 Bytes)	Structure ID (2 Bytes)

Figure 13. SPU Test Command for Photometry Data Field Packet Structure

- **Spare** field (2 Bytes) included for the 4 bytes alignment.
- **Activity ID** field (2 Bytes) for the SPU test command for photometry is '00001010' (0x0A)
- **Structure ID** field (2 Bytes) for the SPU test command for photometry is '00000000' (0x00)

4.1.2.4.8. ACT_TEST_SPEC

Figure 14 represents the Perform Activity command data field packet structure for the SPU Test for Spectroscopy.

Spare (2 Bytes)	Activity ID (2 Bytes)	Structure ID (2 Bytes)

Figure 14. SPU Test Command for Spectroscopy Data Field Packet Structure

- **Spare** field (2 Bytes) included for the 4 bytes alignment.
- **Activity ID** field (2 Bytes) for the SPU test command for spectroscopy is '00001011' (0x0B)
- **Structure ID** field (2 Bytes) for the SPU test command for spectroscopy is '00000000' (0x00)

4.1.2.4.9. CONNECT_DMC

Error! Reference source not found. represents the Perform Activity command data field packet structure for the Connect to DEC/MEC command.

Spare 2 Bytes	Activity ID 2 Bytes	Structure ID 2 Bytes	Parameter1 4 Bytes

Figure 15. Connect to DEC/MEC Command Data Field Packet Structure

- **Spare** field (2 Bytes) included for the 4 bytes alignment.
- **Activity ID** field (2 Bytes) for Connect to DEC/MEC command is '00010000(0x10)
- **Structure ID** field (2 Bytes) for Connect to DEC/MEC command is '00000001' (0x01)
- **Parameter1** field (4 Bytes) contains the SPU link start-up protocol to DEC/MEC (0x11: Master, 0x22: Slave).

4.1.2.5. Write

Figure 16 represents the Write command data field packet structure.

Spare 2 Bytes	Parameter ID 2 Bytes	Length 2 Bytes	Data <Length>4 Bytes	Checksum 2 Bytes	Spare 2 Bytes

Figure 16. Write Command Data Field Packet Structure

- **Spare** field (2 Bytes) included for the 4 bytes alignment.
- **Parameter ID** field (2 Bytes) represents the Memory/Address identifier. The Parameter ID of the write commands is presented in Table 4.

Write Command	Parameter ID	Description
WRT_DXS1	'10000001' (0x81)	This command is used to write the tables of the selected detectors (from which data are requested) on the SPU memory. WRT_DXS1-5 are reserved for the bolometers while WRT_DXS6-7 are reserved for the photoconductors.
WRT_DXS2	0x82	
WRT_DXS3	0x83	
WRT_DXS4	0x84	
WRT_DXS5	0x85	
WRT_DXS6	0x86	
WRT_DXS7	0x87	
WRT_DET_CST_SPEC	'01000010' (0x42)	This command is used to write the table of the photoconductors constants, which are relevant for the pre-processing step.

WRT_DET_CST_PHOT	'00100100' (0x24)	This command is used to write the table of the bolometer constants, which are relevant for the pre-processing step.
WRT_SIM_DATA	'00011000' (0x18)	This command is used to write the table of parameters, which will be used, for the SPU test mode.

Table 4. Parameter IDs for the Write Commands

- **Length** of the data to be written (2 Bytes).

  Its unit is 4 Bytes for the write to the SPU memory.
- **Data** to be written (<Length>4 Bytes).

  The maximum length of data that can be written with a single TC is 204 Bytes as written in AD002.
- **Checksum** that is used to verify the integrity of the data to be written (2 Bytes)
- **Spare** field (2 Bytes) included for the 4 bytes alignment.

4.2. Response Source Packet Structure

The SPU HLSW acknowledges the command reception and/or execution by sending the command response within a maximum timeout of 200 milliseconds[1]. It should be conforming to the structure defined in Figure 17.

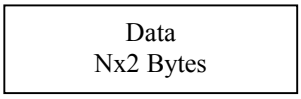

Figure 17. Response Packet Structure

The data field (Nx2 Bytes) is used to acknowledge the DPU command. Its coding is proposed in the following subsections.

4.2.1. Acknowledgement of the LOAD Command

Two kinds of acknowledgement are proposed: positive and negative. Their packet structures are represented respectively in *Figure 18* and *Figure 19*.

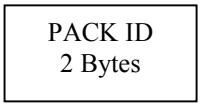

Figure 18. Packet Structure of the Positive Acknowledgement (PACK) of the Load Command

PACK ID: '10000001' (0x81)

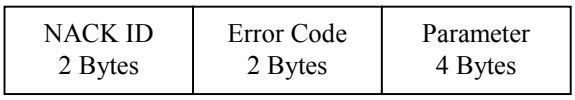

Figure 19. Packet Structure of the Negative Acknowledgement (NACK) of the Load Command

NACK ID: '11110001' (0xF1)

Error Code	Error Definition	Parameter Attached …
0x72	invalid Memory ID	Spare (3 Bytes) and Memory ID (1 Byte)
0x73	invalid Address	Spare (1Byte) and Address (3 Bytes)
0x75	invalid Checksum	Spare (2 Bytes) and Checksum (2 Bytes)
0x7B	invalid Checksum after re-reading	Spare (2 Bytes) and Checksum after re-reading (2 Bytes)

Table 5. Error Codes and Attached Parameters for the Load Command

[1] In this case, the maximum data size, which could be dumped, is 10 Kbytes, while only 9 Kbytes could be checked.

4.2.2. Acknowledgement of the DUMP Command

Two kinds of acknowledgement are proposed: positive and negative. Their packet structures are represented respectively in *Figure 20* and *Figure 21*.

The dumped data packet will be split into blocks of 996 Bytes whenever it exceeds the telemetry packet size. Therefore, the last block will have a size less or equal to 996 Bytes.

PACK ID 2 Bytes	Address1 2 Bytes	Address2 2 Bytes	Length 2 Bytes	Data <Length>SAUs	Checksum 2 Bytes

Figure 20. Packet Structure of the Positive Acknowledgement (PACK) of the Dump Command

PACK ID: '000110000010' (0x182).....for the intermediate packets (packets previous to the last)

'10000010' (0x82).....for the last packet

Address1: represents the memory ID and the MSB (1 byte) of the start address of the dumped data block.

Address2: represents the LSB (2 bytes) of the start address of the dumped data block.

Length: is the size of the dumped data block in words.

Data: maximum 996 Bytes per dumped data block. SAU is 6 Bytes for program memory and 4 Bytes for data memory.

Checksum: the resulting checksum of the dumped data block.

NACK ID 2 Bytes	Error Code 2 Bytes	Parameter 4 Bytes

Figure 21. Packet Structure of the Negative Acknowledgement (NACK) of the Dump Command

NACK ID: '11110010' (0xF2)

Error Code	Error Definition	Parameter Attached …
0x72	invalid Memory ID	Memory ID (1 Byte) and Address (3 Bytes)
0x73	invalid Address field	Memory ID (1 Byte) and Address (3 Bytes)

Table 6. Error Codes and Attached Parameters for the Dump Command

4.2.3. Acknowledgement of the CHECK Command

Two kinds of acknowledgement are proposed: positive and negative. Their packet structures are represented respectively in *Figure 22* and *Figure 23*.

PACK ID 2 Bytes	Address1 2 Bytes	Address2 2 Bytes	Length 2 Bytes	Checksum 2 Bytes

Figure 22. Packet Structure of the Positive Acknowledgement (PACK) of the Check Command

PACK ID: '10000011' (0x83)

The **Address1**, the **Address2** and the **Length** are represented as in the original command.

Checksum: the resulting checksum of the data.

NACK ID 2 Bytes	Error Code 2 Bytes	Parameter 4 Bytes

Figure 23. Packet Structure of the Negative Acknowledgement (NACK) of the Check Command

NACK ID: '11110011' (0xF3)

Error Code	Error Definition	Parameter Attached …
0x72	Invalid Memory ID	Memory ID (1 Byte) and Address (3 Bytes)
0x73	Invalid Address field	Memory ID (1 Byte) and Address (3 Bytes)

Table 7. Error Codes and Attached Parameters for the Check Command

4.2.4. Acknowledgement of the Perform Activity Command

Two kinds of acknowledgement are proposed: positive and negative. Their packet structures are represented respectively in *Figure 24* and *Figure 25*.

PACK ID 2 Bytes

Figure 24. Packet Structure of the Positive Acknowledgement (PACK) of the Perform Activity Command

PACK ID: '10000100' (0x84)

The positive acknowledgement informs that the SPU HLSW has received and understood the Perform Activity command (it is not sent after the execution of the command).

NACK ID 2 Bytes	Error Code 2 Bytes	Parameter 4 Bytes

Figure 25. Packet Structure of the Negative Acknowledgement (NACK) of the Perform Activity Command

NACK ID: '11110100' (0xF4)

Error Code	Error Definition	Parameter Attached …
0x77	Invalid Activity ID	Spare (2 Bytes) and Activity ID (2 Bytes)
0x78	Invalid Structure ID	Spare (2 Bytes) and Structure ID (2 Bytes)
0x79	Invalid SID Parameter	SID Parameter (4 Bytes)

Table 8. Error Codes and Attached Parameters for the Perform Activity Command

4.2.5. Acknowledgement of the Write Command

Two kinds of acknowledgement are proposed: positive and negative. Their packet structures are represented respectively in *Figure 26* and *Figure 27*.

PACK ID 2 Bytes

Figure 26. Packet Structure of the Positive Acknowledgement (PACK) of the Write Command

PACK ID: '10000110' (0x86)

NACK ID 2 Bytes	Error Code 2 Bytes	Parameter 4 Bytes

Figure 27. Packet Structure of the Negative Acknowledgement (NACK) of the Write Command

NACK ID: '11110110' (0xF6)

Error Code	Error Definition	Parameter Attached …
0x74	Length not compatible to Parameter ID	Spare (2 Bytes) and Length (2 Bytes)
0x75	Invalid Checksum	Spare (2 Bytes) and Checksum (2 Bytes)
0x76	Invalid Parameter ID	Spare (2Byte) and Parameter ID (2 Bytes)
0x79	Invalid Data Parameter	Data Parameter (4 Bytes)
0x7B	Invalid Checksum after re-reading	Spare (2 Bytes) and Checksum after re-reading (2 Bytes)

Table 9. Error Codes and Attached Parameters for the Write Command

4.2.6. Response to an Invalid Command ID

This acknowledgement is send in case of receiving an unknown command ID. *Figure 28* represents the packet structure.

NACK ID 2 Bytes	Error Code 2 Bytes	Command ID 4 Bytes

Figure 28. Packet Structure of the Negative Acknowledgement (NACK) of an Invalid Command ID

NACK ID: ‘000111111111’ (0x1FF)

Error Code: ‘01110001’ (0x71) … invalid command ID: unknown command

Command ID is represented as in the original command.

4.3. SPU HLSW HK

4.3.1. Definition and Packet Structure

The SPU HLSW HK packet is sent from the SPU to the DPU. It consists either of compression results and status of the HLSW whenever raw data are received from DEC/MEC, or the SPU HLSW status while no observation is performed. The SPU HK parameters are listed in *Table 10*. Their associated field is depicted in Figure 29. SPU HLSW HK are transmitted at a data rate compatible with the overall PACS HK (see AD001 for details).

The Size of SPU HK data field is 72 Bytes per sub-unit. Every HK parameter is put in a 4-byte Word. In SPU idle mode (stopped data compression), only PIX, CI, CPU_WORKLOAD, DMC_LINK_STATUS, MEM_STATUS_CNTS and PAR_MONITOR fields are filled. The other fields are set to default value.

HK_HEADER 4 B	OBSID 4 B	PIX 4 B	CI 2 B	REAL 2 B	SATUR_FLAG 1 B	SAMP_CORR 3 B	MAINT_RAMPS 2 B	CPU_WORK LOAD 2 B	DMC_LINK_ STATUS 2B

INTEG_RAMPS 1 B	VID 1 B	RCX 2 B	DMC_ERROR 1 B	MEM_STATUS_CNTS 2 B	SPARE_1 2 B	LLC_ERROR 2 B	PAR_MONITOR 2 B

Figure 29. SPU HK Data Packet Structure for Blue SPU/Red SPU

4.3.2. SPU HLSW HK List

The SPU HLSW HK are listed in Table 10.

Parameter	Size (B)	Variable	Default Value	Description
SPUS_HK_HEADER SPUL_HK_HEADER	4	HK Header	0x00870000	This field is added to the HK packet sent from SPU to DPU. It distinguishes the HK from the science packet and the response packet
SPUS_OBSID SPUL_OBSID	4	Observation Identification	0	This field is used to identify the observation the HK belongs to. It should be the same OBSID for the science data.

SPUS_PIX SPUL_PIX	4	Packet Index	0xFFFFXXXX	- The two MSB of the PIX are set to 0, when the HK packet contains the compression results (compression started). In this case, the two LSB are counting the number of downlinked compressed entities since the HLSW handover of the SPU or the last warm reset command. Moreover, the 2 LSB provide the relationship between the science packet and the related HK. The concordance is determined on ground. - The two MSB of the PIX are set to: 1.) 0xFFFF for stopped compression, 2.) 0xDDDD for started compression or 3.) 0xAAAA for auto-stopped compression (Avoiding a SW crash). In all three cases, the two LSB are stalled to the last updated value
SPUS_CI SPUL_CI	2	Counter Increments	0xXXXX	This field is used to attest that HLSW is not hanging. It is periodically incremented from 0 to 65535.
SPUS_REAL SPUL_REAL	2	Reduction Algorithm Used	0xFF	LSB is used to signal the used reduction algorithm if compression started. It is set to 0xFF if on-board data reduction is not used. MSB is used to signal the number of samples per sub-ramp (range: 2-128 in started compression). It is set to 0 if on-board data reduction is not used.
SPUS_SATUR_FLAG SPUL_SATUR_FLAG	1	Saturation Flag	0	0x11: Saturated, 0: Non-Saturated
SPUS_SAMP_CORR SPUL_SAMP_CORR	3	Glitch Counter Information	0	Number of samples affected by glitches for the blue SPU.
SPUS_MAINT_RAMPS SPUL_MAINT_RAMPS	2	Number of Maintained Ramps/ Averages	0	Number of maintained slopes or sub-slopes (Spectroscopy) or averages (Photometry).
SPUS_CPU_WORKLOAD SPUL_CPU_WORKLOAD	2	CPU Workload	0xXXXX	CPU workload for the past second.

SPUS_DMC_LINK_STATUS SPUL_DMC_LINK_STATUS	2	DEC/MEC Link Status	0xFF	DEC/MEC Link Status: 0x00: Link is OFF, 0xFF: Link is ON and 0xAA: Link connection is in progress.
SPUS_INTEG_RAMPS SPUL_INTEG_RAMPS	1	Number of Integrated Ramps/ Averages	0	Number of integrated ramps (Spectroscopy) or averages (Photometry).
SPUS_VID SPUL_VID	1	Version ID	0xXX	This field is used to represent the software version
SPUS_RCX SPUL_RCX	2	Raw Channel Index	0	The most significant 6 bits contain the Nb. of selected raw channels (RCNB) The least significant 10 bits contains the Index of Raw Channels (RCX) until which RCNB pixels data are transmitted with the compressed science data.
SPUS_DMC_ERROR SPUL_DMC_ERROR	1	DEC/MEC Header Error	0	This field is set to 0xFF if an error is detected in the DEC/MEC header.
SPUS_MEM_STATUS_CNTS SPUL_MEM_STATUS_CNTS	2	Memory Status Counters	0xXXXX	LSB: EDAC memory check counter for Single Error Failure MSB: EDAC memory check counter for Double Error Failure
SPUS_SPARE_1 SPUL_SPARE_1	2	Spare	0	Spare HK field
SPUS_LLC_ERROR SPUL_LLC_ERROR	2	Invalid LLC Parameter	0	This HK parameter reports an invalid write parameter for LLC. Default = 0. See the actual HLSW user manual for more details
SPUS_PAR_MONITOR SPUL_PAR_MONITOR	2		0	This HK parameter can be exploited by the user in monitoring a dedicated parameter in the SPU memory. The address of this parameter can be set via a write command. The default value is 0. See the actual HLSW user manual for more details.

Table 10. SPU HLSW HK List

4.4. Science Source Data

SPU HLSW may transmit a maximum number of 75 science data packets to DPU in one buffer (in one go). The science data packet structure is represented in the following subsections.

4.4.1. Packet Structure

Figure 30 depicts the compressed entity packet structure.

CEH 28 Bytes	CDH <CDHS> 4 Bytes	CSD <SCIS> 4 Bytes

Figure 30. Compressed Entity Packet Structure

CEH: Compressed Entity Header.

CDH: Compressed DEC/MEC Header.

CSD: Compressed Science Data.

The compressed entity is split by the SPU into blocks of 1000 Bytes (see AD002) to fit into the telemetry packets (
Figure *31*). A header of 12 Bytes (Block Header) is added to each block before it is sent to the DPU. 8 Bytes of this header will be ignored in the DPU.

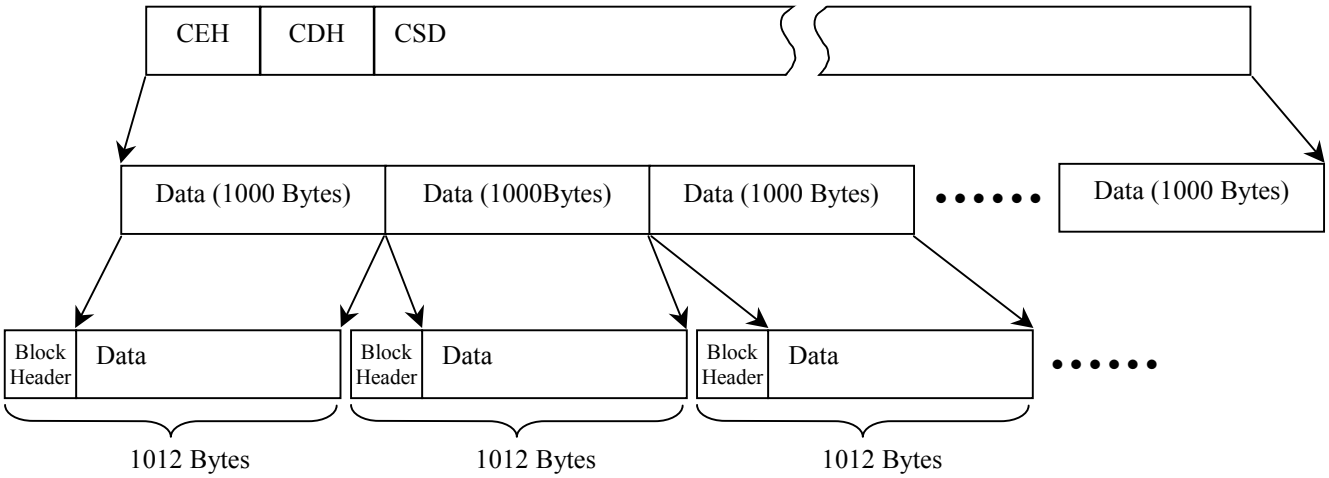

Figure 31. Splitting of the Compressed Entity and the Resulting Science Data Blocks

4.4.1.1. Compressed Entity Header

Figure 32 represents the compressed entity header structure received on ground.

Type 4 Byte	PIX 4 Bytes	DECID 12 Bytes	CRCS 2 Bytes	CDHS 2 Bytes	SCIS 4 Bytes

Figure 32. Compressed Entity Header Structure

Type: Type of the observing Mode (0x01 for spectroscopy and 0x02 for photometry)

PIX: The **PIX field** provides the relationship between the HK packet and the science packet using the same index.

DECID: The **DECID field** (DEcompression Code ID) is analogous to the **CMP/CMM** (Compression Mode) fields in the DEC/MEC Header. This field is used to enter the proper decompression mode.

CRCS: The **CRCS field** represents the size of the Compressed Raw Channel data Size (in 4 Byte words).

CDHS: The **CDHS field** represents the size of the Compressed DEC/MEC Header (in 4 Byte words).

SCIS: The **SCIS field** represents the size of the compressed science data (in 4 Byte words).

4.4.1.2. Compressed DEC/MEC Header

This field contains the lossless compressed DEC/MEC header (see AD001).

4.4.1.3. Compressed Science Data

This field contains the compressed science data (see AD001).

4.4.2. Compressed Entity Transmission Protocol

4.4.2.1. Block Header

Figure 33 represents the block header of the science source packet sent from SPU to DPU.

Header 4 Byte	Counter 4 Bytes	Blocks 4 Bytes

Figure 33. Structure of the Science Data Block sent from SPU to DPU

4.4.2.1.1. Header

The header field (4 Bytes) will be used to identify the packet and data type. This field will not be transmitted to ground. It will be skipped in the DPU level after the collection of the whole blocks of one compressed entity.

Header for science data block is:

0x008A0000 in spectroscopy mode

0x008B0000 in photometry mode

4.4.2.1.2. Counter

The counter field (4 Bytes) specifies the index of the actual transmitted block. Its range is from 1 to 'blocks'. The two upper bytes are not transmitted to ground. They are skipped by DPU.

4.4.2.1.3. Blocks

This field (4 Bytes) specifies the total number of blocks the compressed entity packet has been split into (a block unit is up to 1000 Bytes) to fit into one telemetry packet and sent to the DPU. The two upper bytes are not transmitted to ground. They are skipped by DPU.

4.4.2.2. Data

This field contains the telemetry data block of 1000 Bytes. The collection of these blocks constitutes the compressed entity packet.

5. Are You Alive SPU

The functionality of the SPU HLSW is continuously checked by the DPU monitoring some SPU HLSW HK data. The "Are you alive SPU" could be tested by the DPU using the SPU parameter CI (Table 10) and a time out. Both SPU boards will independently produce HK at regular time interval compatible with the PACS overall HK rate. This time interval is described in AD001. The DPU will always store the HK packets, and checks the parameter CI of the latest received SPU HK packet, at rate of 0.5 Hz, whether they are updated or not, according to the logic described in Table 11. If no update of this entry happens within a maximum time of 8 sec, the DPU will change accordingly its internal HK status related to the corresponding SPU.

CI	DPU Check Rate (Hz)
Incrementing	0.5

Table 11. SPU HK Check by DPU