**JUICE**

**Radio and Plasma Wave Instrument**

**Low Frequency Receiver digital interface**

**JUI-IAP-RPWI-LF-DIF**

**Issue: 01, Rev. 15**

|  |  |  |  |
| --- | --- | --- | --- |
| **Prepared by:** | **Affiliation:** | **Signature:** | **Date** |
| Jan Soucek | IAP |  | 04/10/2023 |
| Ludek Uhlir | IAP |  |  |
| Radek Lan | IAP |  |  |
| Ondrej Santolik | IAP |  |  |
| **Verified by:** |  |  |  |
|  |  |  |  |
| **Approved by:** |  |  |  |
|  |  |  |  |

**Change Record**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Issue** | **Rev.** | **Date** | **Authors** | **Modifications** |
| 1 | 0 | 17/11/2015 | J. Soucek | Initial release |
| 1 | 1 | 10/5/2016 | J. Soucek, O.Santolik | DPU processing added |
| 1 | 2 | 28/11/2016 | L. Uhlir, J. Soucek | Register interface updated. Spectral processing added. Added FFT data product. Major changes throughout. |
| 1 | 3 | 17/2/2017 | L. Uhlir, J. Soucek | Register interface updated. Spectral processing added. EM2 version. |
| 1 | 4 | 14/3/2017 | L. Uhlir, J. Soucek | - Added SRAM uncorrectable error register 1, 2, 3 (address 0x0013, 0x0014, 0x0015)  - Updated data frame system anomalies (0x0009)  - Add memory read/write registers (address 0x0016, 0x0017) |
| 1 | 5 | 18/10/2017 | L.Uhlir, J. Soucek | - Removed SRAM EDAC error counter  - Added CWF, DWF, WFS buffer overflow  - Added SW configuration and TM packet structure |
| 1 | 6 | 17/05/2018 | L.Uhlir, J. Soucek | - PPS counter value can be read without read flag  - Reorganize cap 12.1 Main configuration structure table  - Rename WFS\_NUM\_SAMPS to WFS\_NUMSAMP\_BLOCKS  - Fixed register 0xe description (FPGA firmware version) – added branch version, fixed bit positions  - scm calibration sequence config moved from PWR reg to STAT register  - added power up FFTbin include mask initialization (all bins included)  - added FFTbin mask reload on the fly feature – setting appropriate bit in status register will inform SMX to latch and use actual FFTbin mask configuration  - change STATUS register: data frame ready bit to read only  - added LFR registers to be used for instrument HK packet chapter |
| 1 | 7 | 21/06/2018 | L. Uhlir, J. Soucek | FPGA version 15 released |
| 1 | 8 |  |  | Added link test pattern  Update chapter 12: SRAM memory organization (removed FFT bin mask)  Register “FFT bins summation mask 1 and mask2” is write only  Added SCM OC 1ms glitch filter  Added SCM OC scm power up 200 ms timeout  Added LF self PPS feature  FPGA version 16 released  Update system anomalies  AMBA performance meter replaced by CRC16  Extended FFT Sum product description  FPGA: version 17 released  FPGA: Removed CWF FIR compensator filter  Added register 0xA: SMX number of accumulation description, time computation formula  Added WFS period computation formula in register 0x8  Added WFS length computation formula in register 0x9  Improved Spectral bins description in register 0xF  Fixed Waveform snapshots (WFS) size description |
| 1 | 9 | 22/11/2019 |  | **Version applicable to final FPGA firmware V22.**  Fix LFR SRAM memory organization – update table SMX bin table addresses.  Clarified the use of the time register 0x00 in section 5.  Added “trigger snapshot bit” in register 0x06 (new in firmware V22)  Changed SCM calibration signal mechanism (new in V22). Added description in section 6.  Updates to TM/TC in sections 14 and 15. |
| 1 | 10 | 28/12/2020 |  | **Version applicable to final PFM/FS FPGA firmware V22 and software 1.1.**  Improve DAC bit behavior description in 0xC register.  Corrected SID values  Improve CHCFGREQ bit description in the data product 0x6 register description  Add invalid command flag set condition in write command status register (0x10)  Various updates to SW config and data products. |
| 1 | 11 | 4/10/2021 |  | Updates of software configuration and TM for software version 2.0. |
| 1 | 12 | 05/10/2022 |  | * Fixed/completed format of some TM packets in section 15. * Added and extended sections 16 and 17, describing data processing, SW configuration, triggering etc. |
| 1 | 13 | 01/02/2023 |  | * Renamed “exclude spectral mask” to “include spectral mask” for consistency with the SW implementation * Numerous updates and fixes in PCE config description, TM description etc. * Added TRIGGER\_INFO description for TSWF * Added appendix 2 with description of configurations * Added section 16.5 on interaction with sequencer |
| 1 | 14 | 04/10/2023 |  | * Updated LF configurations in Section 19 for final SW2.0 implementation * Added more details to config structure description in Section 14.1 |

Table of Contents

[1 Applicable and reference documents 6](#_Toc147348230)

[2 LFR data products (FPGA) 7](#_Toc147348231)

[3 Address map 7](#_Toc147348232)

[4 Description of individual registers 8](#_Toc147348233)

[5 Commanding of LFR 15](#_Toc147348234)

[6 LFR registers to be used for instrument HK packet 17](#_Toc147348235)

[7 Data transfer and format 18](#_Toc147348236)

[8 Introduction 18](#_Toc147348237)

[9 Data frame layout 18](#_Toc147348238)

[10 Description of individual fields 19](#_Toc147348239)

[11 Datarates 21](#_Toc147348240)

[12 Memory organization (FPGA) 22](#_Toc147348241)

[13 FPGA side processing of spectral matrices 22](#_Toc147348242)

[14 LF software configuration 24](#_Toc147348243)

[14.1 Processing compression encoding (PCE) configuration – LF main configuration structure 24](#_Toc147348244)

[14.2 Spectral bin tables (SB\_TABLE) 28](#_Toc147348245)

[14.3 Include frequency mask table (EFM table) 28](#_Toc147348246)

[15 LF TM packets (SW interface to OBC) 29](#_Toc147348247)

[15.1 TM\_LF\_RAW 29](#_Toc147348248)

[15.2 TM\_LF\_RSWF 30](#_Toc147348249)

[15.3 TM\_LF\_TSWF 31](#_Toc147348250)

[15.4 TM\_LF\_DWF 32](#_Toc147348251)

[15.5 TM\_LF\_SM 33](#_Toc147348252)

[15.6 TM\_LF\_BP0 34](#_Toc147348253)

[15.7 TM\_LF\_BP1 36](#_Toc147348254)

[15.8 TM\_LF\_DWFS 38](#_Toc147348255)

[15.9 TM\_LF\_BP2 39](#_Toc147348256)

[15.10 TM\_LF\_STAT 40](#_Toc147348257)

[15.11 Common header data 41](#_Toc147348258)

[16 DPU-side software processing of LF data 42](#_Toc147348259)

[16.1 Basic operation and configuration of LF software 43](#_Toc147348260)

[16.2 High resolution waveform snapshots (TM\_LF\_RSWF and TM\_LF\_TSWF) 43](#_Toc147348261)

[16.3 Decimated Waveform at 763/2^N sps (TM\_LF\_DWF and TM\_LF\_DWFS) 43](#_Toc147348262)

[16.4 Spectral data products (TM\_LF\_SM, TM\_LF\_BP0-2) 44](#_Toc147348263)

[16.4.1 TM\_LF\_SM data product: 44](#_Toc147348264)

[16.4.2 TM\_LF\_BP0 data product: 44](#_Toc147348265)

[16.4.3 TM\_LF\_BP1 data product: 45](#_Toc147348266)

[16.4.4 TM\_LF\_BP2 data product: 45](#_Toc147348267)

[16.4.5 Configuration of spectral bins and include masks 45](#_Toc147348268)

[16.4.6 Handing of reaction wheel data 45](#_Toc147348269)

[16.5 Interaction with RPWI software sequencer 46](#_Toc147348270)

[17 Triggering, dust detection and statistics (TM\_LF\_STAT) 46](#_Toc147348271)

[17.1 Basic triggering using LP detection (LF\_ ALGO\_EXTERNAL) 47](#_Toc147348272)

[17.2 Autonomous triggering with dust detection (LF\_ ALGO\_DUST\_WAVE) 48](#_Toc147348273)

[17.3 Triggering based on BP2 49](#_Toc147348274)

[17.4 Common additional triggering settings (handling of long snapshots, immediate trigger, dump cycling) 49](#_Toc147348275)

[18 Appendix 1: BP2 spectral parameter calculation description 50](#_Toc147348276)

[18.1 Configuration parameters: 50](#_Toc147348277)

[18.2 Calibration tables/matrices: 50](#_Toc147348278)

[18.3 Definitions: 51](#_Toc147348279)

[18.4 Initialization steps (to be done on EV\_CONFIG, before first matrix calculation) 51](#_Toc147348280)

[18.5 Routine processing, for every SM matrix received: 52](#_Toc147348281)

[18.6 Final processing after averaging of all matrices is completed: 54](#_Toc147348282)

[19 Appendix 2: Defaults configurations in SW2.0 56](#_Toc147348283)

[19.1 Built-in LF configurations 56](#_Toc147348284)

[19.2 Standard E-field mux configs used in the above LF configurations 58](#_Toc147348285)

[19.3 Use of LF configurations in RPWI operational sequences 59](#_Toc147348286)

# Applicable and reference documents

**Applicable documents**

This document responds to the requirements of the documents listed in the following table:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Mark** | **Reference** | **Title of the document** | **Version** | **Date** |
| AD1 | JUI-IRFU-RPWI-ICD-059\_i1.11 | Interface Control Document for Low Frequency (LF) | 1.11 | 12/06/2018 |
| AD2 | JUI-IRFU-RPWI-MX-021\_i2.6 | Compliance, Traceability and Verification Matrix for RPWI | 2.6 | 30/10/2015 |
| AD3 | JUI-IRFU-RPWI-TN-100\_i1.0 | Technical note - Serial link protocol | 1.0 | 14/6/2013 |
|  |  |  |  |  |
|  |  |  |  |  |

**Reference documents**

The present document refers to the documents listed in the following table:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Mark** | **Reference** | **Title of the document** | **Version** | **Date** |
|  |  |  |  |  |
|  |  |  |  |  |

# LFR data products (FPGA)

LFR FPGA shall produce a combination of the following data products as configured by software in the Data product configuration register (0x0006). Any combination of the data product is allowed with the exception that CWF and WFS cannot be enabled simultaneously.

**Waveform snapshots (WFS):** Waveform snapshots are blocks of 8 x 128 x N\_CWF\_frames (see register WFS length 0x9) digitized at 48.828 ksps or 24.414 ksps buffered internally in the LF and sent after the collection has ended.

**Continuous waveform (CWF):** A continuous stream of samples digitized at 48.828 ksps or 24.414 ksps (8 channels).

**Decimated waveform (DWF):** A continuous stream of decimated data digitized at 762 sps.

**Spectral matrices (SM):** Blocks of 8 x 8 x (number of frequency bins) 64-bit numbers sent at a low cadence (at most once per second).

**Raw FFT (FFT):** A diagnostic product allowing to transmit complex FFTs for al channels continuously. For test only, not to be used in flight.

**FFT Sum (FSUM):** Component sum power spectra calculated at full time resolution (every 1024 samples). One spectrum of 1024 frequencies transmitted per packet. Two spectra can be configured obtained by summing components specified in register 0x18 (register name: FFT bins summation mask 1 and mask2). The output 16 bits samples are custom float numbers composed of 6 bits exponent of base2 (MSbits) and 10 bits of mantissa (LSbits). This number is sum of selected components. Each component have their real part squared summed with imaginary part squared (RE^2+IM^2). The sample decimal value is decoded from sample bits as:

* exponent = sample\_bits(15 downto 10)
* mantissa = sample\_bits (9 downto 0)
* sample\_value = mantissa << exponent

**DCFG:** A special diagnostic packet containing current full configuration and status of LF (addresses 0x0000 to 0x01FF) of the LF configuration memory, with actual time, except 0x12 register – Parity error counter register. The space between the end of LF configuration map (0x01FF) and the end of the packet is filled by a testpattern (0x0, 0x55555555, 0xAAAAAAAA, 0xFFFFFFFF) varying every 32 bits. The rest of the packet Transmitted as a single frame on request (when register 0x0006 is written with bit 15 set to 1).

# Address map

All commanding of LF and readout of status information by the DPU is performed by reading and writing the address space exported by LFR. Map of the LF address space visible to DPU over link is detailed. Only the first 0x0200 words are used.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Address | Function/content |  | R/W | Size | Power on value |
| 0x0000 | Time (21 bits skew + PPS + R/W bits) |  | RW | 32 bit | N/A |
| 0x0002 | HW switch configuration 1 |  | RW | 16 bit | 0 |
| 0x0003 | HW switch configuration 2 |  | RW | 16 bit | 0 |
| 0x0004 | HW switch configuration 3 |  | RW | 16 bit | 0 |
| 0x0005 | HW switch configuration 4 |  | RW | 16 bit | 0 |
| 0x0006 | Data product configuration |  | RW | 16 bit | 0 |
| 0x0007 | PPS to start acquisition | 0-15 | RW | 16 bit | 0 |
| 0x0008 | WFS period [in data frames]  [0 = only one, > 0 periodic acquisition, minimum WFS length+1] |  | RW | 16 bit | 0 |
| 0x0009 | WFS length-1 in blocks of 1024 samples | 0-3071 | RW | 16 bit | 0 |
| 0x000A | NAS: Number of averaged spectra in SM – 1 (e.g. 7 corresponds to 8 averaged spectra) | 0-4095 | RW | 16 bit | 0 |
| 0x000B | PWR control |  |  |  | 0 |
| 0x000C | Status register |  |  |  | 0 |
| 0x000D | Test pattern configuration |  | RW | 16 bit | 0 |
| 0x000E | Firmware version |  | R | 16 bit | Actual version |
| 0x000F | Number of spectral bins (NFB-1). Must be a multiple of 4. | 3-127 | RW | 16-bits | 255 = unconfigured |
| 0x0010 | LFR address map write commands counter | 0-511 | RW |  | 0 |
| 0x0011 | Instrument anomalies (same as data frame @ address 0x1009) |  | RW |  | 0 |
| 0x0012 | 8 bit Parity error counter | 0-256 | R |  | 0 |
| 0x0013 | SRAM uncorrectable error register 1 |  |  |  |  |
| 0x0014 | SRAM uncorrectable error register 2 |  |  |  |  |
| 0x0015 | SRAM uncorrectable error register 3 |  |  |  |  |
| 0x0016 | Memory read/write registers |  | R/W |  |  |
| 0x0018 | FFT bins summation mask 1 and mask2 |  | Write only |  |  |
| 0x0019 | reserved |  |  |  |  |
| 0x0020 – 0x005F | Include mask. A bitmask of FFT bins to be included in spectral matrix averaging. |  | Write only | 1024 bits (64\*16bit) | All bits set to 1 |
| 0x0100-0x01FF | Edges of spectral bins (2\*NFB values). An array of pairs (first bin index, last bin index). | 0-1023 | Write only | 2\*NFB 16bit numbers | Undefined. |

# Description of individual registers

|  |  |  |
| --- | --- | --- |
| **Time0 register (two 16 bit words, addresses 0x0000 and 0x0001): A pair of registers used to read out and set the current local time of LF.** | | |
| **Bits** | **Function/content** | **Allowed values** |
| **Bits 1 (addresses offset 0x0000)** | | |
| 0-7 (LSB) | 8 most significant bits of skew |  |
| 8-11 | PPS counter |  |
| 12 | reserved | 0 |
| 13 | Enable self-PPS mode (incoming PPS is ignored) | 0,1 |
| 14 | Set PPS value bit (write to set PPS value). Reset by instrument | 0,1 |
| 15 | Read instrument skew time. Reset by instrument | 0,1 |
| **Bits 1 (addresses offset 0x0001)** | | |
| 0-15 | 16 least significant bits of skew |  |

The PPS counter is a 4-bit counter incremented by LF with every PPS pulse. The skew is incremented internally with the clock frequency of 781.25 kHz (50 MHz / 64). The skew is reset to 0 with every PPS pulse, when PPS counter is incremented.

It is possible to read the PPS counter value from register 0x0000, as long as it is least 1us after PPS signal. To read PPS it is not necessary to write bit 15.

To read the skew time, a two-step procedure is needed:

* Write the register 0x0000 with a value of 0x8000.
* Read the content of registers 0x0000 and 0x0001.

Register 0x0000 can also be used to set the PPS value. PPS value is set by writing this register, with bit 14 set to 1 and bit 15 set to 0. The value at bits 2-5 is then set as the new PPS value. Other bits are ignored. **The new value is applied when LF receives the next PPS pulse, not immediately.**

The following 4 registers are used to configure hardware switches on LFR. Each bit corresponds to one switch.

|  |  |  |
| --- | --- | --- |
| **HW switch configuration register (four 16 bit words, addresses offset 0x0002-0x0004)** | | |
| **Bits** | **Hardware signal** | **Function** |
| **HW switch configuration register 1 (addresses offset 0x0002)** | | |
| 0 (LSB) | AD1 SCM X EN | Enable opamp in AD1 channel (writeable only when AD1234 REF EN is set) |
| 1 | AD2 SCM Y EN | Enable opamp in AD2 channel (writeable only when AD1234 REF EN is set) |
| 2 | AD3 SCM Z EN | Enable opamp in AD3 channel (writeable only when AD1234 REF EN is set) |
| 3 | AD4 ESUMED EN | Enable opamp in AD3 channel (writeable only when AD1234 REF EN is set) |
| 4 | AD1234 REF EN | Enable references for AD1234 |
| 5-7 | Reserved |  |
| 8 | AD4 ESUMED MUX A |  |
| 9 | AD4 ESUMED MUX B |  |
| 10 | AD4 ESUMED MUX C |  |
| 11 - 15 | Reserved |  |
| **HW switch configuration register 2 (addresses offset 0x0003)** | | |
| 0 (LSB) | E SUM G0 | SC potential sum gain setting 1 |
| 1 | E SUM G1 | SC potential sum gain setting 2 |
| 2 | E SUM E1 EN | Enable probe E1 in the sum |
| 3 | E SUM E2 EN | Enable probe E2 in the sum |
| 4 | E SUM E3 EN | Enable probe E3 in the sum |
| 5 | E SUM E4 EN | Enable probe E4 in the sum |
| 6-14 | Reserved |  |
| 15 | SCME RLD | Reload part 1 of HW switches. Configures E SUM multiplexers, AD1, AD2, AD3 and AD4 |
| **HW switch configuration register 3 (addresses offset 0x0004)** | | |
| 0 (LSB) | AD5 EDINMX1 EN | (writeable only when AD5678 REF EN is set) |
| 1 | AD6 EDINMX2 EN | (writeable only when AD5678 REF EN is set) |
| 2 | AD7 EDINMX3 EN | (writeable only when AD5678 REF EN is set) |
| 3 | AD8 ESUMED EN | (writeable only when AD5678 REF EN is set) |
| 4 | AD5678 REF EN |  |
| 5-7 | Reserved |  |
| 8 | AD8 ESUMED MX A |  |
| 9 | AD8 ESUMED MX B |  |
| 10 | AD8 ESUMED MX C |  |
| 11 - 15 | Reserved |  |
| **HW switch configuration register 4 (addresses offset 0x0005)** | | |
| 0 (LSB) | ED INMX1P A |  |
| 1 | ED INMX1P B |  |
| 2 | ED INMX1N A |  |
| 3 | ED INMX1N B |  |
| 4 | ED INMX2P A |  |
| 5 | ED INMX2P B |  |
| 6 | ED INMX2N A |  |
| 7 | ED INMX2N B |  |
| 8 | ED INMX3P A |  |
| 9 | ED INMX3P B |  |
| 10 | ED INMX3N A |  |
| 11 | ED INMX3N B |  |
| 12 | ED INMXS ENN |  |
| 13-14 | Reserved |  |
| 15 (MSB) | ED RLD | reload part 2 of HW switches. Configures ED multiplexers, AD5, AD6, AD7 and AD8 |

|  |  |  |
| --- | --- | --- |
| **Data product configuration register (address 0x0006): Specified which data products are to be collected and transmitted.** | | |
| **Bit** | **Function/content** | **Allowed values** |
| 0 (LSB) | DWF - Enable decimated WF data |  |
| 1 | SMX - Enable spectral matrices |  |
| 2 | WFS - Waveform snapshots enabled | Only one bit of 2-3 can be set |
| 3 | CWF - Continuous waveform (CWF) produced. | Only one bit of 2-3 can be set |
| 4 | FFT - Enable raw FFT data product. |  |
| 5 | FFT summed |  |
| 6 | Trigger snapshot immediately. To be used without bit 14 set. | 0 |
| 7 | Spare | 0 |
| 8 | If set, sampling of 24.4 kHz is used instead of 48.8  (this affects CWF, SMX and WFS data, not DWF). |  |
| 9-13 | Spares | 0 |
| 14 | CHCFGREQ - Change configuration request – must be set to apply new product configuration in bits (0-8). | When set the instrument is reinitialized and the bit is cleared with a PPS match |
| 15 (MSB) | DCFG - Dump parameters. Send a packet with complete configuration. |  |

Note: if the data products bits are changed before the CHCFGREQ bit is cleared then the invalid command flag (register 0x10) is set

|  |  |  |
| --- | --- | --- |
| **WFS start configuration register (address 0x0007):** | | |
| **Bit** | **Function/content** | **Allowed values** |
| 0-3 | PPS start time of snapshot acquisition data CWF | 0-15 |
| 4-15 | Spares | 0 |

|  |  |  |
| --- | --- | --- |
| **WFS period configuration register (address 0x0008):** | | |
| **Bit** | **Function/content** | **Allowed values** |
| 0-15 | Value = Period-1  Unit is a CWF data frame, one CWF data frame (8channels\*128samples) is collected in 128/Fs ≈ 2.62 ms for Fs ≈ 48kHz  Example for Fs ≈ 48kHz: Value for ~1sec period: 1/0.00262-1 ≈ 380 | - If Value = 0 then only one snapshot is generated  - minimum value = wfs\_length\_value+1 (if less then adjusted by FPGA)  - maximum value = 2^16-1 |

**Note:** LFR accommodates only one buffer for WFS data. If the buffer is filled up then it must be sent out before a new data acquisition begins – thus the configured snapshot period can be prolonged.

|  |  |  |
| --- | --- | --- |
| **WFS length configuration register (address 0x0009):** | | |
| **Bit** | **Function/content** | **Allowed values** |
| 0-15 | Value = WFS snapshot length-1  Unit is a CWF data frame (8channels\*128samples)  Example: Value for ~4ksamps/channel: 4096/128-1 ≈ 31 | Minimum value = 0  Maximum value = 3071 |

|  |  |  |
| --- | --- | --- |
| **SMX number of accumulated spectra register (address 0x000A):** | | |
| **Bit** | **Function/content** | **Allowed values** |
| 0-15 | Value = Number of SMX accumulations-1  single SMX is composed of 8x8 of 64bit numbers (512bytes), one SMX is accumulated in 1024/Fs ≈ 20.97 ms for Fs ≈ 48kHz  Example for Fs ≈ 48kHz: Value for ~10sec period: 10/0.02097-1 ≈ 475 | Minimum value = 0 (nonaccumulated SMXs)  Maximum value = 4095 (4096 accumulations) |

|  |  |
| --- | --- |
| **PWR control register (address 0x000B): Register controlling powering on/off the ADC blocks of LF and generation of SCM calibration signal.** | |
| **Bit** | **Function/content** |
| 0 | Power on SCM part |
| 1 | Power on ED part |
| 2-4 | Reserved |
| 5 | Disable SCM OC protection (10ms) |
| 6 | Power on SCM sensor |
| 7 | Force power supply units to synchronize with LFR clock |
| 8-10 | LFR link push data delay prescaler (‘111’ - > ~2620us (11% duty cycle), ‘000’ -> ~330us (50% duty cycle), default ‘001’ -> ~660us (33% duty cycle) ) |
| 11 | Spares (must be 0) |
| 12-15 | Reset by magic number 11 (0xB) |

|  |  |
| --- | --- |
| **STATUS register (address 0x000C):** | |
| **Bit** | **Function/content** |
| 0 | data frame ready (read only) |
| 1 | Set to inform SMXP to reload FFT bin mask. Bit is self cleared. |
| 2 | Enable the calibration sequence. The calibration signal is triggered and this bit is cleared by the PPS signal.  Note: Once it is set the bit cannot be cleared. In this case an invalid command report is generated.  Note: This bit can be set only when SCM sensor is on. |
| 3-15 | Spares (must be 0) |

|  |  |
| --- | --- |
| **Artificial input register (address 0x000D): Register allowing to configure a test mode, where ADC inputs are replaced by test patterns. For testing only.** | |
| **Bit** | **Function/content** |
| 0-1 | CWF data processing output   * 0: processed data (default) * 1: constant per channel (constant = 256\*channel\_number[0-7]) * 2: sawtooth per channel (channel X output incremented by 1 every 2X sample, where X=[0,..,7]) |
| 2-3 | SCM ADCs output configuration   * 0: ADC samples (default) * 1: constant per channel (constant = 256\*channel\_number[0-3]) * 2: sawtooth per channel (channel X output incremented by 1 every 2X sample, where X=[0,..,3]) |
| 5-6 | Reserved |
| 6-7 | ED ADCs output configuration   * 0: ADC samples (default) * 1: constant per channel (constant = 256\*channel\_number[0-3]) * 2: sawtooth per channel (channel X output incremented by 1 every 2X sample, where X=[0,..,3]) |
| 8 | ADCs samples act as constants in CWF buffer per channel and cwfbuffer number:  sample\_value = channel\_number\*16 + cwf\_buffer\_number\*256  sample\_value is always positive |
| 9 | FFT constants per channel |
| 10 | Link test pattern (per 4 bytes) 0x00000000, 0xF.., 0x5.., 0xA..,… . Data frame headers stay with the actual data product configuration. |

|  |  |
| --- | --- |
| **Firmware version register (address 0x000E): Read only register containing firmware version information.** | |
| **Bit** | **Function/content** |
| 0-3 | serial link interface IP core version (2) |
| 4-7 | LFR FPGA design branch version (0) |
| 8-14 | LFR FPGA design version (13) |
| 15 | LFR FPGA develop design flag (0) |

|  |  |
| --- | --- |
| **Number of spectral bins (address 0x000F):** Number of spectral bins register | |
| **Bit** | **Function/content** |
| 0-6 | Number of SMX bins (NFB-1). Must be a multiple of 4. Read-write.  single SMX is composed of 8x8 of 64bit numbers (512bytes), one SMX is accumulated in 1024/Fs ≈ 20.97 ms for Fs ≈ 48kHz |
| 7 | Read only bit indicating that the register contains an undefined power-on value. Bit is set to zero after first write to register. |
| 8-15 | Reserved (0) |

|  |  |
| --- | --- |
| **Write commands status (address 0x0010):** Register indicating the status of last write operation. | |
| **Bit** | **Function/content** |
| 0-8 | Write counter: 9-bit wrapping counter incremented by one with each write command received. Read only. **Reset on LF soft reset.** |
| 9 | Invalid command flag is set to 1 if :   * SMX product is enabled before number of spectral bins has been set. * Data product configuration changed without change-request bit set * If DAC is enabled when DAC is not powered-on * If DAC is disabled when already enabled (disabled only by instrument) * If changing data products bits of register 0x6 while the CHCFGREQ is set   Can be set to zero by writing zero to this register. |
| 10-15 | Reserved |

|  |  |
| --- | --- |
| **Instrument anomalies register (address 0x0011):** Instrument anomalies (see data frame @ address 0x1009). Can be written to reset the flags. | |
| **Bit** | **Function/content** |
| 0-15 | See data frame @ address 0x1009 |

|  |  |
| --- | --- |
| **Parity error counter (address 0x0012):** Serial link parity error counter. Read only register | |
| **Bit** | **Function/content** |
| 0-7 | Link Parity error counter, ceiling to 255. **Reset with LF reset or configuration change.** |
| 8-15 | Spares (must be 0) |

|  |  |
| --- | --- |
| **SRAM uncorrectable error register 1 (address 0x0013):** SRAM multibit error information register | |
| **Bit** | **Function/content** |
| 0-15 | MSbits of address (16 to 31) of the first error – read only |

|  |  |
| --- | --- |
| **SRAM uncorrectable error register 2 (address 0x0014):** SRAM multibit error information register | |
| **Bit** | **Function/content** |
| 0 | Clear error – when set bits “new error” is cleared. Bit clears itself |
| 1 | New error – read only |
| 2-15 | LSbits of address (2 to 15) of the first error – read only |

|  |  |
| --- | --- |
| **SRAM uncorrectable error register 3 (address 0x0015):** SRAM multibit error information register | |
| **Bit** | **Function/content** |
| 0-7 | reserved |
| 8-11 | Number of amba master issuing the transaction – read only |
| 12-14 | Size of the transaction – read only |
| 15 | Master write signal – read only |

|  |  |
| --- | --- |
| **Memory read/write register 1 (address 0x0016):** memory read/write register 1 | |
| **Bit** | **Function/content** |
| 0-15 | MSbits of read/write address (16 to 31) or MSbits of read data (16 to 31) |

|  |  |
| --- | --- |
| **Memory read/write register 2 (address 0x0017):** memory read/write register 2 | |
| **Bit** | **Function/content** |
| 0 | Read from memory trigger if it is changed from 0 -> 1 (lower priority) |
| 1 | Write to memory trigger if it is changed from 0 -> 1 (higher priority) – if bit zero (read bit) is ‘1’ then data value of ‘111..’ is written, else data value of ‘000..’ is written. |
| 2-15 | LSbits of read/write address (2 to 15) or LSbits of read data (2 to 15) |

Note: useful for SRAM read/write test. RAM address starts at 0xa0000000 and occupies 8MB (up to 0xa07fffff).

|  |  |
| --- | --- |
| **FFT bins summation mask 1 and mask2 (address 0x0018):** FSUM product configuration | |
| **Bit** | **Function/content** |
| 0-7 | FFT sum packet 1 bin mask, default ‘11100000’ (SCM X,Y,Z) |
| 8-15 | FFT sum packet 1 bin mask, default ‘00001110’ (ADC 5,6,7) |

|  |  |
| --- | --- |
| **Spectral matrix FFT bins include bit mask register (address 0x0020 - 0x005f): example of 0x0020** | |
| **Bit** | **Function/content** |
| 0 | FFT bin number 0 flag (1 to include bin, 0 to skip the bin), default ‘1’ |
| 1 – 14 | FFT bin number 1 - 14 flag (1 to include bin, 0 to skip the bin), default ‘1’ |
| 15 | FFT bin number 15 flag (1 to include bin, 0 to skip the bin), default ‘1’ |

|  |  |
| --- | --- |
| **Spectral matrix FFT bins include bit mask register (address 0x0020 - 0x005f): example of 0x005f** | |
| **Bit** | **Function/content** |
| 0 | FFT bin number 1008 flag (1 to include bin, 0 to skip the bin), default ‘1’ |
| 1 – 14 | FFT bin number 1009 - 1022 flag (1 to include bin, 0 to skip the bin), default ‘1’ |
| 15 | FFT bin number 1023 flag (1 to include bin, 0 to skip the bin), default ‘1’ |

# Commanding of LFR

**When powered on LFR will start up in standby mode, where:**

* Analog sections are powered down
* No packets are transmitted

**To configure LFR into science mode, the software has to perform the following steps:**

1. Configure hardware switches (write words at address 0x0002, 0x0003,0x0004, 0x0005, 0x000B)
2. If SMX product is desired (corresponding bit will be set in register 0x0006):

* Set number of spectral bins (NFB) in register 0x000F.
* Upload and configure NFB spectral bins (0x0100 to 0x01FF).
* Configure mask of bins to include (0x0020 to 0x005F) if desired. Otherwise a default is used (all bins will be included in averaging).
* Configure number of averaged spectra in register 0x000A

1. Write the PPS value to start data acquisition in register 0x0007
2. If WFS snapshot acquisition is required, set the WFS period in register 0x0008.
3. Write register 0x0006, setting the bits corresponding to the requested data products. This actually enables the science mode. The acquisition starts on the next PPS signal.

**Changing configuration / re-synchronizing:** Every time register 0x0006 is written (with bit CHCFGREQ = 1). New configuration is applied and data acquisition is stopped and re-started on the PPS pulse specified in register 0x0007. This has to be done for every change in instrument configuration. If no configuration changes are made before writing to 0x0006, the same configuration applies, but data acquisition is re-synchronized to the specified PPS pulse.

**Generating SCM calibration signal:** LF can transmit a synthetic signal to SCM with the following configuration:

* Upload the waveform to be transmitted as 1024 samples (unsigned 16-bit integers) by writing to addresses 0x200-0x5FF. This waveform will later be played back through the DAC (DAC range is 12 bits, so full useful range is between ~100 and 4000. The full range is translated to an output voltage range of about 0 to +5V).
* Configure all LF registers in the desired science configuration same as above.
* Before enabling the configuration by register 0x0006, write register 0x0C to set bit 2 to 1.
* Enable science mode as normal. When the data acquisition starts on the next PPS pulse, the SCM signal is generated immediately.
* The Uploaded waveform is played back in the following manner:
  + It is played to the DAC at a frequency of 524 kHz (2^19 Hz)
  + Initially every 64th sample is sent to output and this is repeated 4 times.
  + Next every 32th sample is sent to output and this is repeated 4 times.
  + Next every 16th sample is sent to output and this is repeated 4 times.
  + ….
  + Full waveform is played at DAC frequency of 524 kHz (four times).
  + Next the full waveform is played (4 times) at half of the initial sampling frequency (at 262 kHz)
  + Then the sampling frequency is halved at every step until the sampling frequency gets to 256 Hz.
* This way, if one uploads to LF one period of a sine wave (centered around 2048 integer value), we get a logarithmic frequency sweep starting at 32768 Hz and ending at 0.25 Hz. At each step, four wave periods are transmitted and frequency is halved at every step.

# LFR registers to be used for instrument HK packet

|  |  |  |  |
| --- | --- | --- | --- |
| **LFR register for instrument HK packet** | | | |
| **Length [bytes]** | **LFR register address** | **name** | **Register description** |
| 4 | NA | SID | Fixed 0x10 |
| 4 | 0x0000 | TIME | Subunit time |
| 2 | 0x0002 | HW\_SW\_CFG1 | LFR switches configuration 1 |
| 2 | 0x0003 | HW\_SW\_CFG2 | LFR switches configuration 2 |
| 2 | 0x0004 | HW\_SW\_CFG3 | LFR switches configuration 3 |
| 2 | 0x0005 | HW\_SW\_CFG4 | LFR switches configuration 4 |
| 2 | 0x0006 | DATA\_PRODUCT\_CFG | Data product configuration |
| 2 | 0x0007 | PPS\_TO\_START | Start data acquisition at defined PPS value |
| 2 | 0x0008 | WFS\_CFG \_PER | WFS product configuration: period value |
| 2 | 0x0009 | WFS\_CFG\_LNG | WFS product configuration: length value |
| 2 | 0x000a | SMX\_CFG\_NAS | SMX product configuration: number of averaged spectras |
| 2 | 0x000b | PWR\_CFG | Power configuration |
| 2 | 0x000c | STATUS | LFR status |
| 2 | 0x000d | TEST\_PATTERNS\_CFG | Test patterns configuration |
| 2 | 0x000e | FPGA\_FW\_VERSION | FPGA firmware version |
| 2 | 0x000f | SMX\_CFG\_NFB | SMX product configuration: number of spectral bins |
| 2 | 0x0010 | WRITE\_REGS\_STAT | Status of written values to the LFR registers |
| 2 | 0x0011 | ANOMALIES | Subunit anomalies detected |
| 2 | 0x0012 | LINK\_ ERR\_CNT | Interface link parity counter |
| 4 | 0x0013 | SRAM\_UC\_ERR | SRAM uncorrectable error |

# Data transfer and format

All data is transmitted as data frames of 1024 16-bit words + header of 10 16-bit words using the push transfer type of the serial link. Each data frame has an identical format described below (section Data frame layout). The data product contained in this frame is determined by the content of the first 16-bit word of the header.

* Waveform snapshots are always sent as frames of 8x128 16-bit words. Unused channels contain random data. The Acquisition time correspond to the first sample of snapshot and only the PPS value is valid (skew should be 0)
* DWF and CWF sent as frames of 8x128 16-bit words. The Acquisition time corresponds to the first sample in the frame.
* Spectral matrices are sent as 8x8 (64) sets of 64-bit numbers corresponding to a single frequency (512 bytes per frequency). NFB frequency bins are transmitted sequentially, 2 frequencies per frame. Note: for SM, the Acquisition time correspond to the first **sample** of the last FFT frame in the matrix.

Special **DCFG** diagnostic packet (data id == 0x8000) is sent by LF on request, when 1 is written to DCFG bit in register 0x0006. This packet contains the dump of the configuration memory of LF (addresses 0x0000 to 0x01FF). This packet has a standard dateframe layout with a standard header and length.

# Introduction

This document provides a specification of communication protocol for the Low Frequency receiver board of the RPWI instrument for JUICE. The document describes both the interface between the FPGA and DPU as well as the configuration and structure of the data products transmitted by RPWI to the spacecraft.

# Data frame layout

This table specified the format of the data frame. All data products are transmitted in frames formatted according to this specification using the push operation. The length of the data frame shall probably be fixed to 1024 + 12 = 1036 16-bit words (2072 bytes).

|  |  |  |  |
| --- | --- | --- | --- |
| Address | Function/content | R/W | Size |
| 0x0000 | Data product (bit mask) | R | 16 bit |
| 0x0001 | System information register | R | 16 bit |
| 0x0002 | Acquisition time (21 bits skew + PPS) | R | 32 bit |
| 0x0004 | Sequential number of frame (wrapped) | R | 16 bit |
| 0x0005 | Artefacts (overflows etc.) | R | 16 bit |
| 0x0006 | HW switch configuration word 1 & 2 | R | 16 bit |
| 0x0007 | HW switch configuration word 3 | R | 16 bit |
| 0x0008 | HW switch configuration word 4 | R | 16 bit |
| 0x0009 | System anomalies | R | 16 bit |
| 0x000A | Sequential number of this frame in one data product (e.g. SM or snapshot) | R | 16 bit |
| 0x000B | Total number of frames in one data product (e.g. SM or snapshot) | R | 16 bit |
| 0x000C | Start of data | R | N x 16 bit |

# Description of individual fields

|  |  |  |
| --- | --- | --- |
| **Data product (data frame, address 0x1000):** A bit mask identifying the type of data in this packet. | | |
| **Bit** | **Function/content** | |
| 0 (LSB) | Decimated WF (DWF) data | Only one of bits 0-4 can be set. |
| 1 | Spectral Matrices |
| 2 | Waveform snapshots |
| 3 | Continuous waveform (CWF) |
| 4 | FFT of CWF |
| 5 | FFT sum |  |
| 6-7 | Spares | 0 |
| 8 | Decimation (0 – 4x, 1 – 8x) |  |
| 9-14 | Spares | 0 |
| 15 (MSB) | LFR register address map |  |

|  |  |  |
| --- | --- | --- |
| **System information register (data frame, address 0x1001):** Contains info about LF system status. | | |
| **Bit** | **Function/content** | **Severity** |
| 0-15 | CRC16 CCITT (version 0xFFFF) computed from 0x0002 | failure |

The acquisition time (time when data was measured) is encoded in 2 16-bit words at 0x1002 and 0x1003 (see the format below). Usually, for DWF, CWF and WFS waveform products, the time corresponds to the first sample in the given data frame. A special case are Spectral Matrices, where the time corresponds to the first sample of the last FFT accumulated in the matrix. So the time corresponding to the beginning of the averaging has to be reconstructed by subtracting NAS\*1024\*cwf\_sampling\_period from the acquisition time (where NAS is the content of register 0x000A, number of averaged FFT spectra – 1, and the cwf\_sampling\_period is either 1/48828.125 or 1/24414.063 sec, depending on decimation setting).

|  |  |  |
| --- | --- | --- |
| **Acquisition time (data frame, address 0x1002):** higher part | | |
| **Bit** | **Function/content** | |
| 0-7 | PPS skew MSbits |  |
| 8-9 | spares |  |
| 10-13 | PPS |  |
| 14-15 | spares |  |
| **Acquisition time (data frame, address 0x1003):** lower part | | |
| **Bit** | **Function/content** | |
| 0-15 | PPS skew LSbits |  |

|  |  |  |
| --- | --- | --- |
| **Artefacts (data frame, address 0x1005):** A bit mask identifying ADC overranges. | | |
| **Bit** | **Function/content** | |
| 0 | ADC n.1 overrange detected |  |
| 1 | ADC n.2 overrange detected |  |
| 2 | ADC n.3 overrange detected |  |
| 3 | ADC n.4 overrange detected |  |
| 4 | ADC n.5 overrange detected |  |
| 5 | ADC n.6 overrange detected |  |
| 6 | ADC n.7 overrange detected |  |
| 7 | ADC n.8 overrange detected |  |
| 8-13 | Reserved |  |
| 14 | SMX: new FFT bins mask applied |  |
| 15 | FFT sum packet generated from FFT summation bitmask 1 (value ‘0’), or from bitmask 2 (value ‘1’) |  |

|  |  |  |
| --- | --- | --- |
| **Data product (data frame, address 0x1006):** HW switch configuration word 1&2. | | |
| **Bit** | **Function/content** | |
| **HW switch configuration register 1** | | |
| 0 | AD1 SCM X EN |  |
| 1 | AD2 SCM Y EN |  |
| 2 | AD3 SCM Z EN |  |
| 3 | AD4 ESUMED EN |  |
| 4 | AD1234 REF EN |  |
| 5 | AD4 ESUMED MUX A |  |
| 6 | AD4 ESUMED MUX B |  |
| 7 | AD4 ESUMED MUX C |  |
| **HW switch configuration register 2** | | |
| 8 | E SUM G0 |  |
| 9 | E SUM G1 |  |
| 10 | E SUM E1 EN |  |
| 11 | E SUM E2 EN |  |
| 12 | E SUM E3 EN |  |
| 13 | E SUM E4 EN |  |
| 14-15 | reserved |  |

|  |  |  |
| --- | --- | --- |
| **System anomalies (data frame, address 0x1009):** A bit mask identifying instrument anomalies in this packet. Flags are cleared by an instrument reset. | | |
| **Bit** | **Function/content** | **Severity** |
| 0 | DMAF amba CWF buffer half-full detected | Warning |
| 1 | DMAF amba CWF buffer full detected | Failure |
| 2 | DMAF amba DWF buffer half-full detected | Warning |
| 3 | DMAF amba DWF buffer full detected | Failure |
| 4 | FFTP input CWF buffer overflow | Failure |
| 5 | SMXP input FFT buffer overflow | Failure |
| 6 | FSUM input FFT buffer overflow | Failure |
| 7 | VIDA input CWF buffer overflow | Failure |
| 8 | VIDA input DWF buffer overflow | Failure |
| 9 | VIDA input WFS buffer overflow | Failure |
| 10 | VIDA input FFT buffer overflow | Failure |
| 11 | VIDA input SMX buffer overflow | Failure |
| 12 | VIDA spectral bin write failure | Warning |
| 13 | DAC write sample failure | Failure |
| 14 | SRAM EDAC multi-bit error detected | Warning |
| 15 | SCM overcurrent flag | Warning |

# Datarates

This section describes the data products generated by the LF receiver toward the DPU (not the actual RPWI products, which are further processed in software). The LF receiver will produce the telemetry at the following rates:

**CWF (Continuous waveform):** One frame of 8x128 samples + header transmitted every 2.6 milliseconds for 48.8 ksps sampling rate (~382 packets per second) continuously. For the reduced sampling rate of 24.4 ksps, the datarate is decrease by half to one frame per 24.4 ksps. When CWF is produced, waveform snapshots are not collected.

**WFS (Waveform snapshots):** This product is sent in the same manner as the CWF product, with the difference that it is not transmitted continuously. LF will transmit a series of frames corresponding to one snapshots (at most 6 MB of data = 8x393216 samples) followed by a gap when no data is transmitted. The rate of packet transmission can be decreased if necessary.

**DWF:** One frame of 8x128 samples + header transmitted every 168 milliseconds (6 packets per second). If enabled, this product is transmitted continuously.

**SM (spectral matrices):**

* The rate of spectral matrices depends on the configuration parameters NFB (number of frequency bins) and NAS (number of averages spectra -1).
* Spectral matrices are sent as 8x8 (64) sets of 64-bit numbers corresponding to a single frequency (512 bytes per frequency). NFB frequency bins are transmitted sequentially, 4 frequencies per frame. The size of each frame is 2048 bytes (or 1024 16-bit words) + header, same as for the waveform products.
* One full matrix (512\*NFB bytes = NFB/4 packets) is sent every 20.9\*(NAS+1) milliseconds.
* Minimum value for NAS is 2, corresponding to a rate SM\_RATE of one matrix every 63 milliseconds (~16 matrices per second).
* A limit is imposed on the product SM\_RATE \* NFB < 512. This implies that for NAS=2 (SM\_RATE =~16 matrices per second), the maximum number of bins is 32 and for NFB = 128 at most 4 matrices can be produced every second. This worst case rate corresponds to 128 SM packets every second.
* Typical SM rate will be one matrix of 32 frequencies every 4 seconds (2 packets per second).

**FFT Sum (FSUM):** When enabled the component sum power spectra is calculated every 1024 CWF samples that takes 21ms and 2072 bytes of the LFR packet. If two FFT sum power spectra are configured then two LFR packets are generated every 21 ms. For the datarate one FFT sum power spectra consumes ~98,7 kBps (48.8 ksps), for two FFT sum power spectra it consumes ~197,3 kBps (97.5 ksps).

# Memory organization (FPGA)

The LFR SRAM memory of 8 Mbytes (8388608 bytes) is organized as follows:

Table ‑ LFR SRAM organization (values are in HEX format). Note that the address used by the SRAM controller is prefixed with 0xA0 (Address 0 = 0xA0000000)



# FPGA side processing of spectral matrices

**Notation used:**

* NFB is number of frequency bins between 4 and 128. Uploaded via TC.
* n = index of output frequency bin (n=1…NFB).
* i = row index of spectral matrix (i=1…8)
* j = column index of spectral matrix (j=1…8)
* f = FFT frequency index. Index to the raw FFT (f = 1…1024)
* Fc(f) = FFT of channel “c”, frequency bin f. Complex number 16-bits. Stored in SRAM.
* bs(n) first frequency in output bin number n. bs(n) is an index to Fc(f) and is between 1..1024. bs(n) is uploaded as a table via TC at LFR initialization.
* be(n) last frequency in output bin number n. be(n) is an index to Fc(f) and is between 1..1024. be(n) is uploaded as a table via TC at LFR initialization.
* ASMij(n) = output accumulated spectral matrix. For each output bin “n” it is 8 x 8 x 64 bits. Stored in external RAM. Real numbers. Real part of cross spectrum for j > i, imaginary for i < j and auto spectrum (real) for i == j.
* include\_mask = bitmask of 1024 bits. If a bit is set to 1, the corresponding FFT bin is included in averaging in frequency.

**Algorithm for SM calculation:**

for n = 1 to NFB

if (first\_accumulated\_fft)

ASMij(n) = 0 for i=1..8,j=1..8

else

Load ASMij(n) from external RAM for i=1..8,j=1..8

end

for f = bs(n) to be(n)

if (include\_mask(f) == 0)

continue

end

load Fc(f) for all c=1..8 from RAM to internal buffer.

for i=1..8

for j=1..8

if (j >= i)

ASMij(n) += Re[Fi(f)]\*Re[Fj(f)] + Im[Fi(f)]\*Im[Fj(f)]

else

ASMij(n) += Im[Fi(f)]\*Re[Fj(f)] - Re[Fi(f)]\*Im[Fj(f)]

end

end

end

end

Write ASMij(n) back to external RAM for i=1..8, j=1..8

end

# LF software configuration

## Processing compression encoding (PCE) configuration – LF main configuration structure

This structure contains a complete configuration of LF, excluding the spectral bin configuration tables described in next section.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Offset [byte]** | **ID** | **Size [bits]** | **Range** | **Description** | |
| **Hardware and general configuration** | | | | | |
| 0 | DATA\_PRODUCTS | 16 |  | Bitmask specifying which data products to send (see below) | |
| **Decimated waveform (DWF) configuration** | | | | | |
| 2 | DWF\_DECIMATION | 8 | 0 to 5 | Bits 0-2: Decimation rate (0 = none, 1 = 2x, …, 5 = 32x)  Bits 3-6: Spare  Bit 7: If set to 1, use the floating point FIR filter with the same kernel as LP. This is only valid with decimation 32x (DWF\_DECIMATION = 0x85) | |
| 3 | DWF\_CHANNEL\_MASK | 8 |  | Bitmask of DWF components to transmit. | |
| **Periodic waveform snapshot (WFS) configuration** | | | | | |
| 4 | WFS\_LENGTH | 16 |  | Length of WFS snapshots in (in blocks of 128 samples) | |
| 6 | WFS\_PERIOD | 16 |  | Period of WFS snapshots (in blocks of 128 samples) | |
| 8 | WFS\_CHANNEL\_MASK | 8 |  | Bitmask of WFS components to transmit. | |
| 9 | ULTRA\_SETTINGS | 8 |  | Additional settings.  Bit 7 (MSB): If set, disable trigger during interference intervals.  Bit 6: If set, use trivial calibration tables (all = 1) for BP1/BP2 calculations.  Bits 0-5: not used, spare | |
| **Decimated Waveform snapshot (DWFS) configuration** | | | | |
| 10 | DWFS\_LENGTH | 16 |  | Length of DWFS snapshots in (in blocks of 128 samples) | |
| 12 | DWFS\_PERIOD | 16 |  | Period of DWFS snapshots (in blocks of 128 samples) | |
| 14 | DWFS\_CHANNEL\_MASK | 8 |  | Bitmask of DWFS components to transmit. | |
| **Triggered waveform snapshot configuration** | | | | |
| 15 | TRIG\_NUM\_SNAP\_STAT | 8 | 0-255 | Number of snapshots to process for one STAT- 1 | |
| 16 | TRIG\_DUMP\_CYCLE | 16 |  | Number of snapshots to process before autonomous trigger dump (if bit 7 of TRIG\_ALGO not set) | |
| 18 | TRIG\_ALGO | 8 |  | Trigger algorithm configuration (see below) | |
| 19 | TRIG\_CHANNEL | 8 |  | Bits 0-2 (low): ADC channel to use for detection  Bit 3: not used  Bits 4-7 (high): segment to trigger on in case of extra long snapshots | |
|  | **BP0/BP1/BP2 common configuration** | | | |
| 20 | BP\_MASK\_EB | 8 |  | Bitmask of B and E to include in averaging. For BP1, this has to include 3xB + 3\*E. For BP2 this indicates the E-components to be used in Poynting calculation (B is ignored for BP2). | |
| 21 | BP\_AVG\_TIME | 8 | 1-15 | Number of SM to average in time to produce one BP - 1 | |
| 22 | BP\_AVG\_FREQ\_LOG2 | 8 | 0-3 | Log2 of number of adjacent frequency bins to average. | |
|  | **SM config** | | | |
| 23 | SM\_NUM\_FREQ\_BINS | 8 | 4-128 | Number of frequency bins. Must be a multiple of 4. | |
| 24 | SM\_NUM\_AVG | 16 | 1-4095 | Number of averaged spectra | |
| 26 | SM\_BINEDGES\_TABLE\_INDEX | 8 | 0-15 | (index to SB\_TABLE table) | |
| 27 | SM\_COMPS | 8 |  | Bitmask of components to include in the matrix. | |
| 28 | SM\_INCLUDE\_MASK\_INDEX | 8 | 0-7 | Index of the include mask (index to EFM\_TABLE) | |
| 29 | RW\_MASKING | 8 |  | Reaction wheel masking configuration. See Sect. 16.4.6. Set to 0 to disable RW masking. | |
| 30 | EXTRA\_SETTINGS | 8 |  | A bitmask of additional settings. See below. | |
| 31 | SM\_CAL\_INDEX | 8 |  | **Low 4 bits:** Index of SM calibration table. SW2.0 values are:  - 0 for Dipole\_LP 49 kHz config  - 1 for Monopole 49 kHz  - 2 for Dipole\_LP 24 kHz  - 3 for Monopole 24 kHz  **High 4 bits:** Index of SM transform matrix. Used in BP1 Poynting and BP2 trace calculations. SW2.0 values are:  - 0 for Dipole\_LP config (AD5,6,7)  - 1 for Monopole (AD5,6,7)  - 2 for Dipole\_uber  -3 for Identity matrix | |
| 32 | JMAG\_MIN\_SAMPLES | 8 |  | Minimum number of samples to consider JMAG data valid for BP1/2 | |
| 33 | BP2\_CONFIG | 8 |  | Bitmask of BP2 settings:  **bits6-7:** transformation matrix index. Used for Poynting vector projections in BP2:  0 = Dipole\_LP, 1 = Monopole, 2 = Dipole\_uber, 3= Identity matrix for AD567  **bit5**: Force fixed JMAG value ([0,0,1])  **bits0-4**: Antennas used in BP2 E trace averaging | |
| 34 | BP2\_SZ\_TRESHOLD | 8 |  | Threshold of BP2 normalized Poynting flux | |
| 35 | STAT\_BLOCKS\_PER\_PACK | 8 | 1-64 | Number of statistical blocks per STAT packet | |
| 36 | TRIG\_THR\_RATIO\_DUST | 16 |  | Peak-med ratio threshold for dust | |
| 38 | TRIG\_THR\_ZX\_DUST | 16 |  | Zero cross threshold for dust | |
| 40 | TRIG\_THR\_RATIO\_WAVE | 16 |  | Peak-med ratio threshold for waves | |
| 42 | TRIG\_THR\_ZX\_WAVE | 16 |  | Zero cross threshold for waves | |
| 44 | TRIG\_THR\_MIN\_AMP | 16 |  | Minimum peak amplitude to consider for trigger | |
| 46 | TRIG\_THRESH\_PAR\_A | 16 |  | Extra parameter A for detection. Maximum RMS value for ALT\_RMS. | |
| 48 | TRIG\_THRESH\_PAR\_B | 16 |  | Extra parameter B for detection | |
| 50 | TRIG\_THRESH\_PAR\_C | 16 |  | Extra parameter C for detection | |
| 52 | TRIG\_THRESH\_PAR\_D | 16 |  | Extra parameter D for detection | |
| 54 | TRIX\_ZX\_OFFSET | 16 |  | A signed offset value used for zero crossings | |
| 56 | trig\_quality | 8 |  | Quality param – what parameter to use for quality factor. One of:  *LF\_QUAL\_RMS\_AMP* = 0, *LF\_QUAL\_PEAK\_AMP* = 1, *LF\_QUAL\_RATIO* = 2, *LF\_QUAL\_ZEROX* = 3, *LF\_QUAL\_WAVES\_PEAK* = 4, *LF\_QUAL\_WAVES\_RMS* = 5, *LF\_QUAL\_DUST\_PEAK* = 6, *LF\_QUAL\_DUST\_RATIO* = 7, *LF\_QUAL\_RMS\_ALT\_CH* = 8,  See triggering section for description. | |
|  | **BP2 triggering config** |  |  |  | |
| 57 | trig\_bp2\_bitmask | 8 |  | A bitmask of parameters to use for BP2 trigger | |
| 58 | trig\_alt\_channel\_mask | 8 |  | Bitmask of channels to include in the Alt RMS value (usually 3 magnetic components). | |
| 59 | trig\_snap\_limit | 8 |  | Maximum number of triggered snapshots to send over dump cycle (for immediate trigger) | |
| 60 | trig\_bp2\_index\_low | 16 |  | The lowest bin to be included in Q calculation | |
| 62 | trig\_bp2\_index\_high | 16 |  | The highest bin to be included in Q calculation | |
| 64 | trig\_bp2\_b\_low | float32 |  | low limit for BP2 B trace | |
| 68 | trig\_bp2\_b\_high | float32 |  | high limit for BP2 B trace | |
| 72 | trig\_bp2\_e\_low | float32 |  | low limit for BP2 E trace | |
| 76 | trig\_bp2\_e\_high | float32 |  | high limit for BP2 E trace | |
| 80 | trig\_bp2\_ellipticity\_low | float32 |  | low limit for BP2 ellipticity | |
| 84 | trig\_bp2\_ellipticity\_high | float32 |  | high limit for BP2 ellipticity | |
| 88 | trig\_bp2\_planarity\_low | float32 |  | low limit for BP2 planarity | |
| 92 | trig\_bp2\_planarity\_high | float32 |  | high limit for BP2 planarity | |
| 96 | trig\_bp2\_theta\_low | float32 |  | low limit for BP2 ellipticity | |
| 100 | trig\_bp2\_theta\_high | float32 |  | high limit for BP2 ellipticity | |
| 104 | trig\_bp2\_phi\_low | float32 |  | low limit for BP2 planarity | |
| 108 | trig\_bp2\_phi\_high | float32 |  | high limit for BP2 planarity | |
| 112 | trig\_bp2\_sz\_low | float32 |  | low limit for BP2 Sz (normalized Poynting) | |
| 116 | trig\_bp2\_sz\_high | float32 |  | high limit for BP2 Sz (normalized Poynting) | |
| 120 | su\_comp\_algo | 8 |  | RPWI compression algorithm (0 = algoNULL, 2 = algoUD, 3 = algoLPC) | |
| 121 | su\_lpc\_algo | 8 |  | Number of coefficients for LPC compression. | |

**Total size = 120 bytes.**

Description of individual fields follows:

|  |  |  |
| --- | --- | --- |
| **DATA\_PRODUCTS: Specified which data products are to be collected and transmitted.** | | |
| **Bit** | **Function/content** | **Allowed values** |
| 0 (LSB) | DWF - Enable decimated WF data |  |
| 1 | SMX - Enable spectral matrices |  |
| 2 | RSWF – Periodic Waveform snapshots enabled | Only one bit of 2-3 can be set |
| 3 | TSWF – Triggered waveform snapshots | Only one bit of 2-3 can be set |
| 4 | BP0 – Enable simple basic parameters | Only one bit of 4,5,7 can be set |
| 5 | BP1 – Enable extended basic parameters | Only one bit of 4,5,7 can be set |
| 6 | DWFS – decimated waveform snapshot |  |
| 7 | BP2 – Special highly compressed spectral parameters | Only one bit of 4,5,7 can be set |
| 8 | STAT – Statistical data packet |  |
| 9-14 | Spares | 0 |
| 15 | If set, sampling of 24.4 kHz is used instead of 48.8  (this affects SMX and WFS data, not DWF). | 0 |

|  |  |
| --- | --- |
| **EXTRA\_SETTINGS: A bitmask of additional LF processing settings** | |
| **Bit** | **Function/content** |
| 0 (LSB) | SCM\_CALIB – if set, a SCM calibration sequence is emitted at the start of the operation. |
| 1-2 | SCM\_AMP – A 2-bit value indicating amplitude of calibration |
| 3 | CFG\_EXTRAS\_BP1\_EDIAG\_TRANS: Transform the electric field data before calculating E-field trace in BP1 data. |
| 4 | CFG\_EXTRAS\_DWFS\_NO\_SEQ: if set, DWFS is generated autonomously, ignoring triggers. |
| 5 | CFG\_EXTRAS\_WFS\_EX\_LONG: enables a special WFS mode is enabled, where WFS snapshots longer than 32k are allowed and they are broken into 32k sample blocks when transmitting. |
| 6 | CFG\_EXTRAS\_JMAG\_BP1\_HDR – JMAG data are put in BP1 header |
| 7 | CFG\_EXTRAS\_JMAG\_BP1\_IBS – IBS JMAG data are used instead of OBS |

|  |  |  |
| --- | --- | --- |
| **TRIG\_ALGO: Configuration of LF snapshot triggering** | | |
| **Bit** | **Function/content** | **Allowed values** |
| 0 (LSB)-1 | ALGO\_CODE: Enum specifying the triggering algorithm:  0 - LF\_ALGO\_EXTERNAL: External trigger (use LP trigger)  1 - LF\_ ALGO\_DUST\_WAVE: Trigger from WFS, detect dust  2 - LF\_ ALGO\_BP2: Trigger from BP2 | 0-2 |
| 2-4 | Spare | 0 |
| 5 | TRIG\_ALGO\_LIMIT – if set, limit the number of triggered snapshots to be sent to TRIG\_DUMP\_CYCLE. Only useful when TRIG\_ALGO\_IMMEDIATE is set. |  |
| 6 | TRIG\_ALGO\_IMMEDIATE – If set, do not store a snapshot in temp buffer, but dump any snapshot fulfilling trigger condition |  |
| 7 | TRIG\_ALGO\_CONCLUDE - If set, wait for conclude trigger event instead of autonomous periodic trigger dump (relevant for ALGO\_CODE = LF\_ ALGO\_DUST\_WAVE and LF\_ ALGO\_BP2) |  |

## Spectral bin tables (SB\_TABLE)

The spectral bin tables specify the edges of frequency bins for averaging of spectral products. These shall be stored in DPU MRAM (changeable by TC) and referred to by an index (SB\_INDEX) from the main configuration structure. When LF board is configured, the table corresponding the specified index shall be uploaded in the LF board using a sequence of write commands.

Each table is 516 bytes long. The DPU shall be able to store up to 16 such tables, each with the following structure:

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Bit size** | **Range** | **Description** |
| SB\_INDEX | 16 | 0-63 |  |
| SB\_NUM\_BINS | 16 | 4-128 |  |
| SB\_BIN\_EDGES | 256\*16 bits |  |  |

**Total size: 516 bytes.**

## Include frequency mask table (EFM table)

The spectral bin tables shall be stored in DPU MRAM (changeable by TC) and referred to by an index (EFM\_INDEX) from the main configuration structure. Each table is 132 bytes long. The DPU shall be able to store up to 8 such tables, each with the following structure:

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Bit size** | **Range** | **Description** |
| EFM\_INDEX | 16 | 0-63 |  |
| Spare | 16 | 0 |  |
| EFM\_MASK | 64\*16 bits |  | This field has 1024 bits, each bit indicates whether a corresponding FFT bin should be masked. |

**Total size: 132 bytes.**

# LF TM packets (SW interface to OBC)

The following LF science TM data products can be generated.

|  |  |  |
| --- | --- | --- |
| **PACKET ID** | **Packet data** | **SID** |
| TM\_LF\_RAW | Raw LF frames | 0 |
| TM\_LF\_RSWF | Waveform snapshot (periodic) | 33 |
| TM\_LF\_TSWF | Waveform snapshot (triggered) | 34 |
| TM\_LF\_DWF | Decimated waveform | 3 |
| TM\_LF\_SM | Spectral matrix | 4 |
| TM\_LF\_BP0 | Simple spectral basic parameters | 5 |
| TM\_LF\_BP1 | Extended spectral basic parameters | 6 |
| TM\_LF\_DWFS | Decimated waveform snapshot | 39 |
| TM\_LF\_BP2 | Extra reduced basic parameters | 8 |
| TM\_LF\_STAT | Trigger statistical packet, incl. dust | 9 |

## TM\_LF\_RAW

This data product is transmitted in RAW mode and is basically a raw LF hardware frame with RPWI header. Designed for debug/calibration.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Offset (byte)** | **ID** | **Size in bytes** | **Range /type** | **Description** |
| **RPWI common header (6 bytes)** | | | | |
| 0 | SID | 1 | 0 | SID = 0: TM\_LF\_RAW |
| 1 | Acquisition Coarse Delta Time | 2 |  | Difference in seconds between the packet coarse time and acquisition coarse time |
| 3 | Acquisition Fine Time | 2 |  | Data acquisition fine time (in units of 2^-64 sec) |
| 5 | SEQ\_CNT | 2 | 0 - 0xFFFF | Sequential counter (per SID) |
| 7 | Aux Length | 1 | 0 |  |
| **Aux header (length = 8 bytes)** | | | | |
| 8 | LF frame | 2072 | data | Raw LF frame, including header, exactly as received from LF. |

## TM\_LF\_RSWF

Periodically collected waveform snapshots sampled at 48.8 ksps or 24.4 ksps, divided into multiple packets. Aux header only in the first packet.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Offset (byte)** | **ID** | **Size in bytes** | **Range /type** | **Description** |
| **RPWI common header (6 bytes)** | | | | |
| 0 | SID | 1 | 33 | SID = 33: TM\_LF\_RSWF |
| 1 | Acquisition Coarse Delta Time | 2 |  | Difference in seconds between the packet coarse time and acquisition coarse time |
| 3 | Acquisition Fine Time | 2 |  | Data acquisition fine time (in units of 2^-64 sec) |
| 5 | SEQ\_CNT | 2 | 0 - 0xFFFF | Sequential counter (per SID) |
| 7 | Aux Length | 1 | 0 |  |
| **Aux header (length = 8 bytes)** | | | | |
| 8 | SWITCHES1 | 4 | Bitmask | HW switches1 |
| 12 | SWITCHES3 | 1 | Bitmask | HW switches2 |
| 13 | COMPONENT\_MASK | 1 | Bitmask | A bitmask of components in the packet. **num\_comp =** number of nonzero bits in COMPONENT\_MASK |
| 14 | TOTAL\_PACKETS | 2 | Unsigned  <= 3072 | Number of packets forming one snapshot. |
| **Start of data (length = 4 + 2\*128\*num\_comp. Maximum length 2052 bytes)** | | | | |
| 16 | ARTEFACTS | 1 | Bitmask | ADC overflow bits. |
| 17 | SNAPSHOT\_NUMBER | 1 | Bitmask | Sequential counter incremented with each snapshot. |
| 18 | SEQ\_COUNTER | 2 | Unsigned | Packet number within one snapshot. |
| 20 | DATA | 2\*num\_comp  \*128 | Signed int16 | Waveform data encoded as 16b integers. 128 samples per channel, from num\_comp channels. |

## TM\_LF\_TSWF

Triggered waveform snapshot sampled at 48.8 ksps or 24.4 ksps, divided into multiple packets. Aux header only in the first packet.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Offset (byte)** | **ID** | **Size in bytes** | **Range /type** | **Description** |
| **RPWI common header (6 bytes)** | | | | |
| 0 | SID | 1 | 34 | SID = 34: TM\_LF\_TSWF |
| 1 | Acquisition Coarse Delta Time | 2 |  | Difference in seconds between the packet coarse time and acquisition coarse time |
| 3 | Acquisition Fine Time | 2 |  | Data acquisition fine time (in units of 2^-64 sec) |
| 5 | SEQ\_CNT | 2 | 0 - 0xFFFF | Sequential counter (per SID) |
| 7 | Aux Length | 1 | 12 | Aux\_len = 12 |
| **Aux header (length = 12 bytes) – only in first packet** | | | | |
| 8 | SWITCHES1 | 4 | Bitmask | HW switches1 |
| 12 | SWITCHES2 | 1 | Bitmask | HW switches2 |
| 13 | COMPONENT\_MASK | 1 | Bitmask | A bitmask of components in the packet. **num\_comp =** number of nonzero bits in COMPONENT\_MASK |
| 14 | TOTAL\_PACKETS | 2 | Uint4 + Uint12 | Low 12 bits: Number of packets forming one segment (< 3072)  High 4 bits: Number of segments the packet is divided into |
| 16 | TRIGGER\_INFO | 2 |  | Trigger algorithm information. See below. |
| 18 | QUALITY | 2 |  | Quality factor |
| **Data (the following block is repeated):** | | | | |
| 20 | ARTEFACTS | 1 | Bitmask | ADC overflow bits. |
| 21 | SNAPSHOT\_NUMBER | 1 | Bitmask | Sequential counter incremented with each snapshot. |
| 22 | SEQ\_COUNTER | 2 | Unsigned | Packet number within one snapshot. |
| 24 | DATA | 2\*num\_comp  \*128 | Signed int16 | Waveform data encoded as 16b integers. 128 samples per channel, from num\_comp channels. |

**TRIGGER\_INFO 16bit word contains the following:**

|  |  |  |
| --- | --- | --- |
| **Bits** | **ID** | **Description** |
| 0-2 | TRIG\_ALGO\_CODE | Configured triggering algorithm:  0 - LF\_ALGO\_EXTERNAL: External trigger (use LP trigger)  1 - LF\_ ALGO\_DUST\_WAVE: Trigger from WFS, detect dust  2 - LF\_ ALGO\_BP2: Trigger from BP2 |
| 3-5 | trig\_channel | LF channel used for triggering (for LF\_ ALGO\_DUST\_WAVE) |
| 6-9 | quality\_param | Quality parameter used. Same as trig\_quality in config. |
| 10-12 | SELECTION\_REASON | Reason for snapshot dump. One of:  *LF\_TRIG\_SEL\_IMMEDIATE* = 1 - *snapshot dumped immediately, LF\_TRIG\_SEL\_BEST\_CONCLUDE* = 2 - *best stored snapshot dumped on conclude trigger event, LF\_TRIG\_SEL\_ON\_CONCLUDE* = 3 -*snapshot taken on conclude trigger (buffer was empty), LF\_TRIG\_SEL\_BEST\_TIMER* = 4 - *best stored snapshot dumped on periodic timer.* |
| 13-15 | SIGNAL\_TYPE | Type of detected signal. Only non-zero for LF\_ ALGO\_DUST\_WAVE:  0 = unknown, 1 = wave, 2 = dust, 3 = other |

## TM\_LF\_DWF

Continuous waveform sampled at 763 Hz or optionally decimated down to 763/2^N Hz.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Offset (byte)** | **ID** | **Size in bytes** | **Range** | **Description** |
| **RPWI common header** | | | | |
| 0 | SID | 1 | 3 | SID = 3: TM\_LF\_DWF |
| 1 | Acquisition Coarse Delta Time | 2 |  | Difference in seconds between the packet coarse time and acquisition coarse time |
| 3 | Acquisition Fine Time | 2 |  | Data acquisition fine time (in units of 2^-64 sec) |
| 4 | SEQ\_CNT | 2 | 0 - 0xFFFF | Sequential counter (per SID) |
| 7 | Aux Length | 1 | 0 | Aux len = 0 |
| **Packet header (length = 8 bytes)** | | | | |
| 8 | SWITCHES1 | 4 | Bitmask | HW switches1 |
| 12 | SWITCHES2 | 1 | Bitmask | HW switches2 |
| 13 | ARTEFACTS | 1 | Bitmask | ADC overflow bits. |
| 14 | COMPONENT\_MASK | 1 | Bitmask | A bitmask of components in the packet.  num\_comp = number of nonzero bits in COMPONENT\_MASK |
| 15 | DECIMATION | 1 | 0-5 | Decimation factor. |
| **Start of data (length = 2\*128\*num\_comp. Maximum length 2048 bytes)** | | | | |
| 16 | DATA | 16\*num\_comp  \*128 | Signed int16 | Waveform data encoded as 16b integers. 128 samples per channel, from num\_comp channels. |

## TM\_LF\_SM

On-board calculated full spectral matrix (8 x 8 or reduced to smaller dimensions).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Offset (byte)** | **ID** | **Size in bytes** | **Range /type** | **Description** |
| **RPWI common header** | | | | |
| 0 | SID | 1 | 4 | SID = 4: TM\_LF\_SM |
| 1 | Acquisition Coarse Delta Time | 2 |  | Difference in seconds between the packet coarse time and acquisition coarse time |
| 3 | Acquisition Fine Time | 2 |  | Data acquisition fine time (in units of 2^-64 sec) |
| 5 | SEQ\_CNT | 2 | 0 - 0xFFFF | Sequential counter (per SID) |
| 7 | Aux Length | 1 | 12 | Aux len = 12 |
| **Aux header (length = 12 bytes)** | | | | |
| 8 | SWITCHES1 | 4 | Bitmask | HW switches1 |
| 12 | SWITCHES2 | 1 | Bitmask | HW switches2 |
| 13 | Spare | 1 | 0 | Spare = 0 |
| 14 | SB\_INDEX + EFM\_INDEX | 1 | 5 + 3 bits | 5 bits SB\_INDEX MSbits  3 bits EFM\_INDEX LSbits |
| 15 | NUM\_FREQ | 1 | 1-128 | Number of frequency bins. |
| 16 | NUM\_AVG\_SPEC | 2 |  | Low 14 bits (0-13): Number of averaged spectra in SM.  Bit 14: If set, indicates a dynamic RW mask is used  Bit 15: If set, indicates new RW mask was applied. |
| 18 | COMPONENT\_MASK | 1 | Bitmask | Bitmask of components included in matrix. **num\_comp** = number of nonzero bits in COMPONENT\_MASK |
| 19 | BLOCK\_SIZE | 1 | 2 | = num\_comp\*(num\_comp+1)  Size of data ´block corresponding to one frequency bin in bytes. |
| **Start of data (NUM\_FREQ \*block size. Maximum 9216 bytes. Can be split in smaller packets.)** | | | | |
| 20 | ARTEFACTS | 2 | Bitmask | ADC overflow bits |
| 22 | Spare | 2 | 0 | Spare = 0 |
| 24 | RW\_FREQS | 8 | 8 x 0-255 | 8 masked reaction wheel frequency bins. |
| 32 | SM\_DATA | NUM\_FREQ\*  BLOCK\_SIZE |  | A sequence of NUM\_FREQ matrices, each BLOCK\_SIZE bytes large. Structure of the SM block defined below. |

Spectral matrix data block per frequency bin (BLOCK\_SIZE bytes each)

|  |  |  |  |
| --- | --- | --- | --- |
| **Offset (byte)** | **ID** | **Size (bytes)** | **Description** |
| 0 | AUTO\_SPECTRA | Num\_comp\*2 | Power spectrum encoded as 10(MSbits - mantisa)+6 (LSbits – exponent) float. |
| 2\*num\_comp | CROSS\_SPECTRA | num\_comp\*(num\_comp-1) | Off-diagonal SM elements encoded as 8+8 bit complex numbers. |

## TM\_LF\_BP0

Simple spectral product for selective downlink. Only contains E and B power spectra calculated in software from spectral matrices. Not split in multiple packets.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Offset (byte)** | **ID** | **Size in bytes** | **Range /type** | **Description** |
| **RPWI common header** | | | | |
| 0 | SID | 1 | 5 | SID = 5: TM\_LF\_BP0 |
| 1 | Acquisition Coarse Delta Time | 2 |  | Difference in seconds between the packet coarse time and acquisition coarse time |
| 3 | Acquisition Fine Time | 2 |  | Data acquisition fine time (in units of 2^-64 sec) |
| 5 | SEQ\_CNT | 2 | 0 - 0xFFFF | Sequential counter (per SID) |
| 7 | Aux Length | 1 | 0 | 0 |
| **Packet header (length = 12 bytes)** | | | | |
| 8 | SWITCHES1 | 4 | Bitmask | HW switches1 |
| 12 | SWITCHES2 | 1 | Bitmask | HW switches2 |
| 13 | ARTEFACTS | 1 | Bitmask | ADC overflow bits |
| 14 | AVG\_TIME\_FREQLOG2 | 1 | 4 + 4 bits unsigned | 4 bits spectral averaging  4 bits time averaging |
| 15 | SB\_INDEX + EFM\_INDEX | 1 | 5 + 3 bits | 5 bits SB\_INDEX MSbits  3 bits EFM\_INDE LSbits |
| 16 | NUM\_AVG\_SPEC | 2 | 0-4095 | Low 14 bits (0-13): Number of averaged spectra in SM.  Bit 14: If set, indicates a dynamic mask is used  Bit 15: If set, indicates new mask was applied. |
| 18 | NUM\_FREQ | 1 | 1-128 | Number of frequency bins. |
| 19 | MASK\_EB | 1 | Bitmask | Bitmask of E and B to include in averaging. |
| 20 | RW\_FREQS | 8 | 8 x 0-255 | 8 masked reaction wheel frequency bins. |
| **Start of data (length = 4\*NUM\_FREQ bytes, maximum 512 bytes)** | | | | |
| 28 | DATA | 2\*2\*NUM\_FREQ | 10+6 floats | 2 spectra of NUM\_FREQ bins. Spectral power encoded in 16bits (10-bit mantissa + 6-bit exponent). |

## TM\_LF\_BP1

Extended spectral product, to be used as quicklook for selective downlink, but also for science. Not split in multiple packets.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Offset (byte)** | **ID** | **Size in bytes** | **Range /type** | **Description** |
| **RPWI common header** | | | | |
| 0 | SID | 1 | = 6 | SID = 6: TM\_LF\_BP1 |
| 1 | Acquisition Coarse Delta Time | 2 |  | Difference in seconds between the packet coarse time and acquisition coarse time |
| 3 | Acquisition Fine Time | 2 |  | Data acquisition fine time (in units of 2^-64 sec) |
| 5 | SEQ\_CNT | 2 | 0 - 0xFFFF | Sequential counter (per SID) |
| 7 | Aux Length | 1 | = 0 |  |
| **Packet header (length = 12 bytes)** | | | | |
| 8 | SWITCHES1 | 4 | Bitmask | HW switches1 |
| 12 | SWITCHES2 | 1 | Bitmask | HW switches2 |
| 13 | ARTEFACTS | 1 | Bitmask | ADC overflow bits |
| 14 | AVG\_TIME\_FREQLOG2 | 1 | 4 + 4 bits unsigned | 4 bits spectral averaging  4 bits time averaging |
| 15 | SB\_INDEX + EFM\_INDEX | 1 | 5 + 3 bits | 5 bits SB\_INDEX MSbits  3 bits EFM\_INDEX LSbits |
| 16 | NUM\_AVG\_SPEC | 2 |  | Low 14 bits (**0-13**): Number of averaged spectra in SM.  **Bit 14**: If set, indicates a dynamic mask is used  **Bit 15:** If set, indicates new mask was applied. |
| 18 | NUM\_FREQ | 1 | 1-128 | Number of frequency bins. |
| 19 | MASK\_EB | 1 | Bitmask | Bitmask of B and E to include in averaging. Note: For BP1, Must be selected 3 electric and 3 magnetic. |
| 20 | RW\_FREQS | 8 | 8 x 0-255 | 8 masked reaction wheel frequency bins. |
| 28 | JMAG\_MANTIS | 3 | Signed U8 | Averaged JMAG vector (3 x mantis).  0xFF if JMAG data is off / not available.  0xFE signals an error.  Invalid value also signaled by JMAG\_MANTIS = (0,0,0) and JMAG\_EXP = 0x1f |
| 31 | JMAG\_EXP | 1 |  | **Bits 0:4**: Averaged JMAG vector exponent. This is a signed value between (-15 and 15)  **Bit 5:** If set, JMAG/LF times are not synced  **Bit 6:** If set, IBS sensor is used by JMAG  **Bit 7:** 0 – spare |
| **Start of data (length = 16\*NUM\_FREQ bytes, maximum 2048 bytes)** | | | | |
| 32 | DATA | 16\*NUM\_FREQ | See below | A sequence of 16-byte data blocks described below (one per frequency bin). |

Data block per frequency bin (16 bytes per block)

|  |  |  |  |
| --- | --- | --- | --- |
| **Offset (bits)** | **ID** | **Bit size** | **Description** |
| 0 | BP1\_EL\_DIAG | 3\*8 | Diagonal E-field elements (mantissas, 3 x uint8) |
| 24 | BP1\_MAG\_DIAG | 3\*8 | Diagonal B-field elements (mantissas, 3 x uint8) |
| 48 | BP1\_POYNITNG | 3\*8 | Normalized Poynting vector (3 x int8) |
| 72 | BP1\_MAG\_EXP | 6 | Common B-field exponent |
| 78 | BP1\_EL\_EXP | 6 | Common E-field exponent |
| 84 | BP1\_PV\_EXP | 2 | Additional exponent bits for Poynting vector. 0x3 means overflow |
| 86 | BP1\_MAG\_OFF\_DIAG | 6\*7 | 6 element array of 7-bit numbers, containing normalized off-diagonal elements of magnetic 3x3 spectral matrix. |

## TM\_LF\_DWFS

A waveform snapshot created from the DWF product, divided into multiple packets. Aux header only in the first packet.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Offset (byte)** | **ID** | **Size in bytes** | **Range /type** | **Description** |
| **RPWI common header** | | | | |
| 0 | SID | 1 | 39 | SID = 39: TM\_LF\_DWFS |
| 1 | Acquisition Coarse Delta Time | 2 |  | Difference in seconds between the packet coarse time and acquisition coarse time |
| 3 | Acquisition Fine Time | 2 |  | Data acquisition fine time (in units of 2^-64 sec) |
| 5 | SEQ\_CNT | 1 | 0 - 0xFFFF | Sequential counter (per SID) |
| 7 | Aux Length | 1 | 8 | Aux\_len = 8 |
| **Aux header (length = 8 bytes)** | | | | |
| 8 | SWITCHES1 | 4 | Bitmask | HW switches1 |
| 12 | SWITCHES2 | 1 | Bitmask | HW switches2 |
| 13 | COMPONENT\_MASK | 1 | Bitmask | A bitmask of components in the packet. **num\_comp =** number of nonzero bits in COMPONENT\_MASK |
| 14 | TOTAL\_PACKETS | 2 | Unsigned  <= 3072 | Number of packets forming one snapshot. |
| **Start of data (length = 4 + 2\*128\*num\_comp. Maximum length 2052 bytes)** | | | | |
| 16 | ARTEFACTS | 1 | Bitmask | ADC overflow bits. |
| 17 | SNAPSHOT\_NUMBER | 1 | Bitmask | Sequential counter incremented with each snapshot. |
| 18 | SEQ\_COUNTER | 2 | Unsigned | Packet number within one snapshot. |
| 20 | DATA | 2\*num\_comp\*128 | Signed int16 | Waveform data encoded as 16b integers. 128 samples per channel, from num\_comp channels. |

## TM\_LF\_BP2

Another simple spectral product for selective downlink. This contains very reduced power spectra and wave polarization parameters. Not split in multiple packets.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Offset (byte)** | **ID** | **Size in bytes** | **Range /type** | **Description** |
| **RPWI common header** | | | | |
| 0 | SID | 1 | 8 | SID = 8: TM\_LF\_BP2 |
| 1 | Acquisition Coarse Delta Time | 2 |  | Difference in seconds between the packet coarse time and acquisition coarse time |
| 3 | Acquisition Fine Time | 2 |  | Data acquisition fine time (in units of 2^-64 sec) |
| 5 | SEQ\_CNT | 2 | 0 - 0xFFFF | Sequential counter (per SID) |
| 7 | Aux Length | 1 | 0 | 0 |
| **Packet header (length = 12 bytes)** | | | | |
| 8 | SWITCHES1 | 4 | Bitmask | HW switches1 |
| 12 | SWITCHES2 | 1 | Bitmask | HW switches2 |
| 13 | ARTEFACTS | 1 | Bitmask | ADC overflow bits |
| 14 | AVG\_TIME\_FREQLOG2 | 1 | 4 + 4 bits unsigned | 4 bits spectral averaging  4 bits time averaging |
| 15 | SB\_INDEX + EFM\_INDEX | 1 | 5 + 3 bits | 5 bits SB\_INDEX MSbits  3 bits EFM\_INDE LSbits |
| 16 | NUM\_AVG\_SPEC | 2 | 0-4095 | Low 14 bits (0-13): Number of averaged spectra in SM.  Bit 14: If set, indicates a dynamic mask is used **(not implemented, not needed)**  Bit 15: If set, indicates new mask was applied. |
| 18 | NUM\_FREQ | 1 | 1-128 | Number of frequency bins. |
| 19 | MASK\_EB | 1 | Bitmask | Bitmask of E and B to include in averaging. |
| 20 | RW\_FREQS | 8 | 8 x 0-255 | 8 masked reaction wheel frequency bins. |
| 28 | FLAGS | 1 | Bitmask | Various flags indicating data state |
| 29 | Spare | 1 | 0 | Spare = 0 |
| **Start of data (length = 4\*NUM\_FREQ bytes, maximum 512 bytes)** | | | | |
| 30 | DATA | 4\*NUM\_FREQ | Packed structure | An array 4-byte of data blocks described below. |

Data block per frequency bin (4 bytes per block)

|  |  |  |  |
| --- | --- | --- | --- |
| **Offset (bits)** | **ID** | **Bit size** | **Description** |
| 0 | B\_TRACE | 8 | Trace of B matrix (encoded as 5+3 log2) |
| 8 | E\_TRACE | 8 | Trace of E matrix (encoded as 5+3 log2) |
| 16 | THETA | 4 | Theta angle of K-vector in B-aligned coordinates (0 to 90, encoded to 4 bits) |
| 20 | PHI | 4 | PHI angle of K-vector in B-aligned coordinates (-180 to 180, encoded to 4 bits) |
| 24 | ELLIPTICITY | 3 | Signed ellipticity (-1 to 1) encoded in 3 bits (resolution 0.25) |
| 27 | PLANARITY | 3 | Planarity (0 … 1) encoded in 3 bits (resolution 0.125) |
| 30 | S\_PAR | 2 | B-parallel component of Poynting vector, encoded in 2 bits (two positive and two negative values) |

## TM\_LF\_STAT

A very short TM packet containing triggering results, including dust and wave counters, peak/RMS amplitudes etc. Not split in multiple packets.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Offset (byte)** | **ID** | **Size in bytes** | **Range /type** | **Description** |
| **RPWI common header** | | | | |
| 0 | SID | 1 | 9 | SID = 9: TM\_LF\_STAT |
| 1 | Acquisition Coarse Delta Time | 2 |  | Difference in seconds between the packet coarse time and acquisition coarse time |
| 3 | Acquisition Fine Time | 2 |  | Data acquisition fine time (in units of 2^-64 sec) |
| 5 | SEQ\_CNT | 2 | 0 - 0xFFFF | Sequential counter (per SID) |
| 7 | Aux Length | 1 | 16 | Aux len = 16 |
| **Aux header (length = 16 bytes)** | | | | |
| 8 | SWITCHES1 | 4 | Bitmask | HW switches1 |
| 12 | SWITCHES2 | 1 | Bitmask | HW switches2 |
| 13 | NUM\_SNAP\_ALL | 1 | 0-255 | Number of snapshots per block, including saturated, minus 1 |
| 14 | NUM\_BLOCKS | 1 | 1-64 | Number of blocks in packet (up to 64) |
| 15 | ALGO\_CODE | 1 |  | Algorithm code |
| 16 | ALGO\_COMPONENT | 1 |  | Selected component used for trig/analysis |
| 17 | ALT\_BITMASK | 1 |  | Bitmask of Alt channels |
| 18 | SNAP\_PERIOD | 2 |  | Period of WFS snapshots (in 128sp blocks) |
| 20 | SNAP\_LENGTH | 2 |  | Length of snapshot (in 128-sample blocks) |
| 22 | Spare | 2 | 0 | Spare = 0 |
| **Start of data (M blocks, length = NUM\_BLOCKS\*20)** | | | | |
| Description of one stat block below, this is repeated NUM\_BLOCK times | | | | |
| 0 | NUM\_WAVES | 1 |  | Number of waves detected |
| 1 | NUM\_DUST\_POS | 1 |  | Number of positive dust spikes detected |
| 2 | NUM\_DUST\_NEG | 1 |  | Number of negative dust spikes detected |
| 3 | NUM\_ALL\_GOOD | 1 |  | Number of all snapshots, except saturated |
| 4 | WAVE\_ZEROX\_MED | 2 |  | Number of zero crossings (median) |
| 6 | WAVE\_PEAK\_AMP | 2 |  | Peak amplitude of identified waves |
| 8 | WAVE\_RMS\_AMP | 2 |  | RMS wave amplitude |
| 10 | DUST\_MED\_AMP | 2 |  | Median amplitude of dust |
| 12 | DUST\_PEAK\_AMP | 2 |  | Peak amplitude of dust (signed) |
| 14 | SNAP\_PEAK\_AMP | 2 |  | Peak snapshot amplitude |
| 16 | SNAP\_RMS\_AMP | 2 |  | RMS snapshot amplitude |
| 18 | WAVE\_RMS\_ALT | 2 |  | RMS amplitude of waves from Alt channels |

## Common header data

**HW\_SWITCHES\_ARTEFACTS:** A compressed bitmask containing complete information about LF hardware switches and data artefacts (overflows). Size of this bitmask is 6 bytes (48 bits).

|  |  |  |
| --- | --- | --- |
| **Bit** | **ID** | **Description** |
| **Switches 1 (32 bit word)** | | |
| 0 | AD1 SCM X EN | SCM channel 1 enabled |
| 1 | AD2 SCM Y EN | SCM channel 2 enabled |
| 2 | AD3 SCM Z EN |  |
| 3 | AD4 ESUMED EN |  |
| 4 | AD1234 REF EN |  |
| 5 | AD4 ESUMED MUX A |  |
| 6 | AD4 ESUMED MUX B |  |
| 7 | AD4 ESUMED MUX C |  |
| 8 | E SUM G0 |  |
| 9 | E SUM G1 |  |
| 10 | E SUM E1 EN |  |
| 11 | E SUM E2 EN |  |
| 12 | E SUM E3 EN |  |
| 13 | E SUM E4 EN |  |
| 14 | Spare = 0 |  |
| 15 | SAMP\_RATE | Waveform sampling rate: 0 if 48.8 ksps, 1 if 24.4 ksps. |
| 16 | ED INMX1P A |  |
| 17 | ED INMX1P B |  |
| 18 | ED INMX1N A |  |
| 19 | ED INMX1N B |  |
| 20 | ED INMX2P A |  |
| 21 | ED INMX2P B |  |
| 22 | ED INMX2N A |  |
| 23 | ED INMX2N B |  |
| 24 | ED INMX3P A |  |
| 25 | ED INMX3P B |  |
| 26 | ED INMX3N A |  |
| 27 | ED INMX3N B |  |
| 28 | ED INMXS ENN |  |
| 29 | Spare = 0 |  |
| 30 | Spare = 0 |  |
| 31 | Spare = 0 |  |
| **Switches2 (byte)** | | |
| 0 (LSB) | AD5 EDINMX1 EN |  |
| 1 | AD6 EDINMX2 EN |  |
| 2 | AD7 EDINMX3 EN |  |
| 3 | AD8 ESUMED EN |  |
| 4 | AD5678 REF EN |  |
| 5 | AD8 ESUMED MX A |  |
| 6 | AD8 ESUMED MX B |  |
| 7 | AD8 ESUMED MX C |  |
| **Artefacts (byte)** | | |
| 8 | ADC1\_OR | ADC n.1 overrange detected |
| 9 | ADC2\_OR | ADC n.2 overrange detected |
| 10 | ADC3\_OR | ADC n.3 overrange detected |
| 11 | ADC4\_OR | ADC n.4 overrange detected |
| 12 | ADC5\_OR | ADC n.5 overrange detected |
| 13 | ADC6\_OR | ADC n.6 overrange detected |
| 14 | ADC7\_OR | ADC n.7 overrange detected |
| 15 | ADC8\_OR | ADC n.8 overrange detected |

**RW\_FREQS:** 8-byte array where each byte indicates the FFT bin masked due to the reaction wheel signature. There are 4 RWs and for each RW we can mask the fundamental and the 8-th harmonic, hence up to 8 frequencies can be basked out (8 bytes are necessary).

The value of the byte N\_rw is a frequency bin corresponding to the original 2048 point FFT (masked freq = f\_samp \* N\_rw / 2048), where f\_samp is LFR sampling frequency (either 48828 Hz or 24414 Hz).

There are two special values: 0xFF indicates no mask is used and 0xFE is an invalid value (generally indicating and error).

# DPU-side software processing of LF data

This section describes the handling of the LF data in the DPU software and the LF software operation in the normal operational mode (LF\_NOMINAL\_SCI submode or the SCINECE mode). LF software also supports other modes (LF\_RAW and LF\_RFT\_FFT), where the processing is effectively disabled, but these are only used for testing.

In this section and below, the variables in***bold-italic*** refer to configuration parameters in the PCE configuration structure.

## Basic operation and configuration of LF software

The operation of LF in SCIENCE mode (NOMINAL\_SCI submode) is configured by the following data structures modifiable by telecommands.

Two structures which are a part of the LF configuration selected either by TC or by the sequencer

**HD\_Config** structure: Structure directly configuring some LF FPGA registers (For the description of the registers, see Section 4). In normal SCIENCE mode, only a subset of the register values in this structure is used:

* + HW\_SWITCH\_CONFIG1..4 - sets the hardware switches of LF board (registers 0x02 to 0x05)
  + PWR\_CTRL – configures register 0x0B (controls power of SCM and LF board and other hardware features).
  + NAS – Number of spectral matrices to average in FPGA (register 0x0A).
  + lf\_submode should be set to LF\_NOMINAL\_SCI.

The rest of values is ignored and the registers are set with values calculated by the software from the settings in the PCE structure. For the description of the registers, see Section 4.

* **PCE\_Config** structure: The main configuration structure, defining most of LF software behaviour and data generation. See Section 14.1 for description.

Multiple parameter tables needed for data processing:

* Spectral bin tables (SB\_TABLE): These tables configure the output “reduced” frequency bins of the spectral products. Only one table is active at a time, selected by an index in the PCE\_Config structure. See Section 14.2 for description.
* Include frequency mask table (EFM\_TABLE): These tables configure the which frequency bins should be included in the averaging of the spectral products. Only one table is active at a time, selected by ***SM\_INCLUDE\_MASK\_INDEX*** in PCE\_Config.
* There are more tables …. **TBW**

## High resolution waveform snapshots (TM\_LF\_RSWF and TM\_LF\_TSWF)

The TM\_LF\_RSWF (Regular Science Waveform Snapshot) data product returns multi-component waveform snapshots taken periodically in regular intervals specified by ***WFS\_PERIOD***. The TM\_LF\_TSWF product provides the same snapshots, but instead of being captured periodically, they are selected based on one of the triggering algorithms described below. The TM\_LF\_RSWF and TM\_LF\_TSWF products are mutually exclusive, only one can be enabled at a time.

Software will need to select a subset of the components specified by a bitmask ***WFS\_CHANNEL\_MASK*** in the configuration structure. Only the selected components will be put in the TM packet. The snapshot length (number of snapshots per channel) is specified in ***WFS\_LENGTH*** in multiples of 128 points.

## Decimated Waveform at 763/2^N sps (TM\_LF\_DWF and TM\_LF\_DWFS)

LF supports two decimated data products generated from the slow data sampled by the FPGA at 763 sps.

* **TM\_LF\_DWF** is a continuous time series than can be decimated from the original 763 Hz by a factor of 2^N configured by ***DWF\_DECIMATION*** using an FIR filter. This is used to generate the 24 Hz continuous data. Any set of field components can be selected by ***DWF\_CHANNEL\_MASK***. This product is always transmitted in individual data packets, each containing 128 samples**.**
* **TM\_LF\_DSWF** is a multi-component time series sampled always at 763 Hz and ***DWFS\_CHANNEL\_MASK*** configures the E/B components to be included. This product is generated in the form of “snapshots” similar in format to TM\_LF\_RSWF. The length of the DSWF snapshots in samples per component is set by the ***DWFS\_LENGTH*** parameter (in multiples of 128). DSWF product. TM\_LF\_DWFS can be generated either periodically (similar to TM\_LF\_RSWF) or in response to a trigger.

To enable periodic DWFS generation, the bit CFG\_EXTRAS\_DWFS\_NO\_SEQ has to be set in the EXTRA\_SETTINGS variable and the period (start to start) is set via DWFS\_PERIOD in multiples of 128 samples per channel. This can be used to generate a continuous DSWF waveform (such as in the in-situ burst mode) by setting ***DWFS\_PERIOD*** equal to ***DWFS\_LENGTH (****e.g. to 0x20).*

For description of how the triggered DWFS snapshots are generated, see Section 17.

Both TM\_LF\_DWF and TM\_LF\_DSWF can be generated at the same time and can be configured, enabled or disabled independently.

## Spectral data products (TM\_LF\_SM, TM\_LF\_BP0-2)

The LF board FPGA provides to the DPU averages spectral matrices. These are further processed in the software. Several types of post processing will be applied to SM packets to produce different spectral products:

### TM\_LF\_SM data product:

This product contains the full spectral matrices, further averaged in software and converted to a more efficient format.

* Can be produced in parallel with BP0, BP1 or BP2.
* Selection of components: Typically, not all 8x8 components are valid/useful and the software will reduce the matrix to a subset of e.g. 6x6 or 3x3. These components are selected by the ***SM\_COMPS*** bitmask.
* Reduction of the 64 bits integers to a more compressed format. The diagonal elements of the matrix are compressed to a 16-bit float value and each pair of the non-diagonal elements (forming a complex number) are normalized to an integer complex number encoded in a 16-bit integer (2 x 8 bits).

### TM\_LF\_BP0 data product:

This product contains integrated power spectra calculated from the SM, summed over several components

* BP0 can be produced in parallel with TM\_LF\_SM but not with BP1/BP2. BPx products are mutually exclusive.
* The data product contains two averaged auto spectra (E and B):
* These are calculated as summed auto-spectra (diagonal elements of the SM). Summation is performed according to the bitmask specified in ***BP\_MASK\_EB***.
* Spectra can be reduced by averaging in time and frequency according to ***BP\_AVG\_TIME*** and ***BP\_AVG\_FREQ\_LOG2***.

### TM\_LF\_BP1 data product:

Calculation reduced version of spectral parameters (reduced spectral matrix + Poynting vector). Also includes in the header the magnetic field vector provided to RPWI by JMAG. The calculation is performed by

* Averaging of transformed spectral matrices in time and frequency (set via ***BP\_AVG\_TIME*** and ***BP\_AVG\_FREQ\_LOG2).***
* Processing of averaged spectral matrices to obtain, for each frequency step:
* three components of E-field spectrum encoded as 3 x 8 bits mantissa + a common 6-bit exponent
* three components of B-field spectrum encoded as 3 x 8 bits mantissa + a common 6-bit exponent
* Off-diagonal elements
* Poynting vector : 3x8=24 bit,

→ Total set: 128 bit (16 byte) per frequency and time interval.

### TM\_LF\_BP2 data product:

This product contains the wave polarization parameters (power, ellipticity, planarity and Poynting vector direction) in a highly compressed form. Again, this product can be generated in parallel with TM\_LF\_SM, but not with other BPx products. The calculation involves transforming the spectral matrices to field aligned coordinates using JMAG data, averaging, applying on-board calibration and afterwards calculating the spectral parameters.

Details of the spectral calculation are given in an appendix (Section 18).

### Configuration of spectral bins and include masks

As described in Section 13, the LF FPGA calculates the spectrum with 2048 FFT, yielding 1024 “source” frequency bins, and afterwards accumulates the spectral matrices both in time and frequency. All spectral data use a common “frequency” axis (specifying the output frequency bins after frequency reduction) and include mask (which allows to include/exclude individual bins of the source 1024 point spectrum – masked out bins, those set to 0 in the mask, are excluded from averaging/summing).

In the PC configuration, the index ***SM\_BINEDGES\_TABLE\_INDEX*** gives and index to the active SB\_TABLE (a table containing multiple sets of frequency bins – see 14.2) and ***SM\_INCLUDE\_MASK\_INDEX*** is an index to the active EFM\_TABLE, where the include masks are specified as 1024bit bitmasks.

The number of output frequency bins for the TM\_LF\_SM data product must be configured in ***SM\_NUM\_FREQ\_BINS***.

For BP0/BP1/BP2 products, the same frequency axis is used, but may be further reduced in frequency by summing adjacent frequency bins together. This frequency reduction is configured by ***BP\_AVG\_FREQ\_LOG2.*** The software sums together 2^***BP\_AVG\_FREQ\_LOG2*** adjacent bins, so for BP\_AVG\_FREQ\_LOG2 = 0, the BPx frequency axis is the same as SM axis, for BP\_AVG\_FREQ\_LOG2 = 1, the number of frequency bins in BPx products is reduced to one half, comparing to SM.

### Handing of reaction wheel data

LF software receives reaction wheel information from the OBC and dynamically updates the include mask in the FPGA to mask out the RW frequencies. There are 4 reaction wheels and the software allows to mask out the frequency f and its 8th harmonic 8\*f. The frequencies to mask are configured by the **RW\_MASKING** parameter in PCE configuration (bit 0 = fundamental of wheel 1, bit 1 – 8th harmonic of wheel 1, bit 2 – fundamental of wheel 2, ….).

Once the LF SW receives a new value for the RW frequencies, it calculates the frequencies to mask and applies a new mask on the start of the next LF\_SM data product. This new mask is constructed from the static mask specified by ***SM\_INCLUDE\_MASK\_INDEX*** with additional frequencies masked due to RW. The fact that a new mask has been applied is indicated in the TM\_LF\_SM/TM\_LF\_BPx packet headers by bit 15 in NUM\_AVG\_SPEC. The packet headers also reflect the masked frequencies in the 8-byte field RW\_FREQS.

## Interaction with RPWI software sequencer

The LF hardware does not require real time control from the sequencer and runs autonomously. In usual operation modes, the sequencer / software initializes the LF to a given configuration and LF operates in this mode autonomously, independently of the sequencer cycle. However, the LF software handles the following sequencer events on software level:

**RPWI\_EV\_INTERFERENCE\_START and RPWI\_EV\_INTERFERENCE\_STOP:** These two events signal to the LF software that E-field data is most likely affected by LP or MIME sweeps and do not contain a valid natural signal. LF software reacts in the following manner:

1. It pauses averaging of spectral matrices and BPx data products. During this interval, no new datapoints are included in the spectral averaging.
2. LF internal triggering in (in LF\_ALGO\_DUST\_WAVE and LF\_ ALGO\_BP2 modes) is paused, so the SW will not trigger inside this interval.

**RPWI\_EV\_TRIGGER:** This event is used only in triggering mode based on LP detection (LF\_ALGO\_EXTERNAL). In this mode, LF software triggers a snapshot on reception of the event.

**RPWI\_EV\_CONCLUDE\_TRIGGER:** Used always in LF\_ALGO\_EXTERNAL mode and in in LF\_ALGO\_DUST\_WAVE and LF\_ ALGO\_BP2 modes when ***TRIG\_ALGO\_CONCLUDE*** bit is set in configuration. In this configuration, the SW dumps the content of the temporary buffer only when this event is received, not autonomously.

**RPWI\_EV\_INIT:** LF software performs static initialization

**RPWI\_EV\_CONFIG:** LF applies a given configuration and starts LF science operation

**RPWI\_EV\_STOP and RPWI\_EV\_ABORT\_R:** When either of those events is received, RPWI SW resets the LF board.

# Triggering, dust detection and statistics (TM\_LF\_STAT)

The LF software allows multiple modes of triggering of its triggered snapshots (TM\_LF\_TSWF and TM\_LF\_DWFS). The global modes of LF trigger operation is specified by TRIG\_ALGO variable in the PCE structure. The ALGO\_CODE can be set to one of the following:

* LF\_ALGO\_EXTERNAL: LF uses external trigger from the LP board, controlled by the sequencer. This is the same operation as was implemented in Release 1.
* LF\_ ALGO\_DUST\_WAVE: LF internal triggering, when LF triggers autonomously based on the 49 ksps data it collects. It attempts to detect waves and dust impacts in the data via a set of criteria. In this regime (and only in this regime), the TM\_LF\_STAT product is generated.
* LF\_ ALGO\_BP2: Triggers the waveform snapshots based on spectral data, calculated by LF.

## Basic triggering using LP detection (LF\_ ALGO\_EXTERNAL)

In this mode, LF triggers using the EV\_TRIGGER and EV\_CONCLUDE\_TRIGGER events received from the sequencer, based on LP data. This mode only works with the sequencer, if run without a sequencer, no triggered snapshots are transmitted.

The basic operation is as follows:

- Whenever LF receives the EV\_TRIGGER from LP, it commands the FPGA to capture TSFW snapshot (if TSWF enabled) and stores the snapshot in its internal buffer.

- A DWFS snapshot is captured as well (if DWFS product is enabled and the ***CFG\_EXTRAS\_DWFS\_NO\_SEQ*** bit is not set), centered on the time of EV\_TRIGGER and stored in a separate temporary buffer.

- If a new trigger EV\_TRIGGER arrives, the buffers are overwritten.

- When EV\_CONCLUDE\_TRIGGER is received the content of the temporary buffers is transmitted in the form of TM\_LF\_TSWF and TM\_LF\_DWFS products and the buffers are emptied. If the buffers are empty, the snapshots are taken at the time of EV\_CONCLUDE\_TRIGGER and transmitted immediately.

## Autonomous triggering with dust detection (LF\_ ALGO\_DUST\_WAVE)

The triggering is performed by analyzing waveform snapshots taken periodically by LF.

* The source snapshot length and cadence is configured in PCE structure via ***wfs\_length*** and ***wfs\_period*** parameters.
* Triggering is performed on one component which is selected by ***trig\_channel*** variable in PCE structure.

Below is described the algorithm (Note: the ***bold-italic*** variables in this section correspond to configuration parameters from the PCE structure):

Inputs: snapshot E(t), N = ***wfs\_length***\*128 samples, triggering done based on a selected component

1. **For each snapshot we compute:**

* MAX = max(abs(E))
* MED = median(abs(E))
* ZX = number of zero crossings of (E – ***trig\_zx\_offset*** offset). Proportional to frequency for narrowband waves.
* RMS\_E = RMS value of E
* RMS B = RMS value of B (summed over alt channel bitmask– specified by bits in **trig\_alt\_channel\_mask**)

1. **Snapshot is classified as either wave or dust**

if ((MAX/MED > ***trig\_thr\_ratio\_dust***) & (ZX < Thresh3) & (RMS\_B < Param\_A)) **=> the snapshot is dust**

else if ((MAX/MED < Thresh2) & (ZX > Thresh4) & (RMS\_B > Param\_B)) **=> the snapshot is a wave**

1. **Quality factor calculation:**

A quality factor **Q** is calculated from the snapshot. A method is chosen by the ***trig\_quality*** variable and can be one of:

* RMS\_AMP – largest rms amplitude signal stored
* PEAK\_AMP – largest peak amplitude signal stored
* RATIO – largest MAX/MED ratio
* ZEROX -- number of zero crossings
* DUST\_PEAK – peak amplitude with preference to dust (+ 32k if dust)
* DUST\_RATIO – RATIO with preference to dust (+ 32k if dust)
* WAVE\_PEAK - peak amplitude with preference to waves (+ 32k if waves)
* WAVE\_RMS -
* RMS\_ALR\_CH - RMS value from alternate channels (summed).

1. **Triggered snapshot selection**

The algorithm keeps the highest quality snapshot in an internal buffer (unless immediate triggering is configured), until a higher quality snapshot is found. If the current snapshot has a higher **Q** than the stored one, the stored one is replaced by the new one. The triggered snapshot is transmitted as TM\_LF\_TSWF and the internal buffer is cleared using one of the approaches defined in section 17.4.

*DFWS triggering:*In this triggering mode the DWFS snapshot can either be triggered at the same time as the TSWF (the DWFS trigger following the TSWF trigger) or the DWFS triggering can follow the basic mechanism described in 17.1, when DWFS is triggered by the EV\_TRIGGER event from sequencer.

1. **Statistic (TM\_LF\_STAT) calculation.**

The results of the processing of all the waveform snapshots processed by the algorithm are used to collect statistics. The results are transmitted in the form of STAT blocks, with each block containing information collected over ***TRIG\_NUM\_SNAP\_STAT*** processed waveform snapshots. This includes the number of positive and negative dust spikes identified, number of waves identified, maximum and average signal amplitude etc. (see the TM\_LF\_STAT packet description). To reduce packet overhead, multiple STAT blocks can be combined in a single TM\_LF\_STAT telemetry packet. The number of blocks in a packet is set by ***STAT\_BLOCKS\_PER\_PACK.***

## Triggering based on BP2

In this regime, the trigger for the TSWF triggered snapshot is based on a quality factor calculated from the BP2 spectral product. This importantly requires that WFS and BP2 products are configured synchronously (WFS snapshot starts at the same time as BP2 spectral data collection). So, the WFS\_PERIOD must match BP\_AVG\_TIME and NAS setting.

The following algorithm is then used to calculate the quality factor which is then used for triggering.

quality = 0;

for (idx ***= trig\_bp2\_index\_low***; idx < ***trig\_bp2\_index\_high***; idx++)

{

bool BtraceOK = (Btrace >= ***trig\_bp2\_b\_low***) && (Btrace < ***trig\_bp2\_b\_high***)

bool EtraceOK = (Etrace >= ***trig\_bp2\_e\_low***) && (Etrace < ***trig\_bp2\_e\_high***)

bool ElipOK = …. and analogously for planarity, theta, phi, Sz.

if (BtraceOK && EtraceOK && ElipOK && ......) {

quality += Btrace

}

}

if (quality > trig\_bp2\_q\_threshold) {

**trigger the snapshot!**

}

*DFWS triggering:*In this triggering mode the DWFS snapshot can either be triggered at the same time as the TSWF (the DWFS trigger following the TSWF trigger) or the DWFS triggering can follow the basic mechanism described in 17.1, when DWFS is triggered by the EV\_TRIGGER event from sequencer.

## Common additional triggering settings (handling of long snapshots, immediate trigger, dump cycling)

For all the triggering algorithms (LF\_ ALGO\_DUST\_WAVE and LF\_ ALGO\_BP2), a common mechanism is used to determine when is the triggered snapshot sent out (and the internal buffer cleared in applicable). This general mechanism is described here together with some specifics.

In normal operation, the currently “best” triggered snapshot is kept in memory buffer and is sent to the spacecraft after the end of the current cycle (ether driven by sequencer or by internal LF cycle – see below). This is however only possible for snapshots up to than 96 ksamples per component, due to memory limitation of RPWI. To allow LF to transmit longer snapshots, up to 256 ksamples a specific mechanism must be enabled as follows (This can be used with all LF\_ALGO\* algorithms):

* CFG\_EXTRAS\_WFS\_EX\_LONG bit has to be set in the ***EXTRA\_SETTINGS*** to allow for the snapshot length to exceed 32 ksamples (per channel). When this is set, the RSWF and TSWF snapshots are splits in blocks of 32 ksamples. This is also requires the snapshot length specified in WFS\_LENGTH to be a multiple of 32 ksamples.
* If snapshots longer than 96 ksamples are required, the TRIG\_ALGO\_IMMEDIATE bit must be set as well. In this case, the triggered snapshots are not kept in the internal temporary buffer, but are immediately transmitted to the DPU. To avoid overloading telemetry in case of frequent triggers, the *TRIG\_ALGO\_LIMIT* bit should be set, limiting the number of triggered snapshots to be sent to one per *TRIG\_DUMP\_CYCLE*.

For internal triggering algorithms (LF\_ ALGO\_DUST\_WAVE and LF\_ ALGO\_BP2), additional trigger behavior is determined by the setting of ***TRIG\_ALGO\_CONCLUDE*** bits in configuration:

* if ***TRIG\_ALGO\_CONCLUDE*** is set, the snapshot is dumped when EV\_CONLCUDE\_TRIGGER is received from the sequencer.
* if ***TRIG\_ALGO\_CONCLUDE*** is not set, the LF dumps the data autonomously after it processes ***TRIG\_DUMP\_CYCLE*** snapshots.

# Appendix 1: BP2 spectral parameter calculation description

This was taken from an internal specification, needs clarification.

## Configuration parameters:

* **nBins** = number of frequency bins in spectral matrix.
* **indicesED** = indices of E antennas to use in Poynting calculation (there can be up to 5 antennas)
* **bp\_avg\_time** and **bp\_avg\_freq\_log2** are used the same way as for BP0 and BP1
* **indicesTrace** = indices of E antennas to use in calculation of the trace of Electric field (1,2 or 3 antennas can be selected)

Note: indices start at 1 in this section.

## Calibration tables/matrices:

These are defined in LF\_Tables.h/.c

1) **TM\_scm\_sc (lfTransMatB in code):** 3x3 real matrix, transforming from SCM coordinates to SC coordinates. For example

**B\_sc** = **TM\_scm\_sc** \* **B\_scm** (B\_scm is SCM magnetic field in original sensor coordinates)

2) **Ant\_dir\_sc (lfTransMatEAnt in code):** 3 x 5 real matrix, columns are unit directions of antennas corresponding to E1,E2,…, E5 in SC coordinates

3) **CalMat** (8 x 64 complex – **lfSmCalMatrices in code**), read from lfSmCalMatrices. Frequency dependent calibration of 8 components for a given mode at 64 frequency bins. Used to generate calibration matrices to convert the integer M\_fpga matrix to physical units.

4) **TM\_ant\_sc (lfTransMatE in code):** 3x3 matrix corresponding to a chosen indicesTrace. A matrix to convert a 3 component E-field vector from antenna coordinates to SC coordinates. Will include antenna lengths. The matrix is computed on ground and uploaded to SW. Software should have a set of 4 such matrices, selectable by a TC

5) **TM\_mag\_sc** **(lfTransMatJmag in code):** Transformation matrix from JMAG sensor coordiates to SC coordiates.

## Definitions:

**M\_fpga** (matrix 8 x 8 x nbins, 64bit integer) – matrix from FPGA, averaged in frequency in SW if configured.

**EB(i,j,f)** [3 x 5 x nbins, complex float] – submatrix of ExB elemens of M\_fpga, calibrated

**BB(i,j,f)** [3 x 3 x nBins, complex float] – submatrix of BxB elemens of M\_fpga, calibrated

**TM\_sc\_mfa** [3 x 3 x nBins, real float] – transformation matrix from SC to field aligned (MFA) coordinates, where

- axis mfa3 is along B\_jmag

- axis mfa1 is orthogonal to 3, ad lies in a plane containing X axis of SC system.

- axis mfa2 is orthogonal to mfa1 and mfa3

TM\_sc\_mfa is calculated as (where x is Vector cross product):

T(3,:) = B\_jmag / norm(B\_jmag);

T(2:,) = B\_jmag x [1,0,0] / norm(B\_jmag x [1,0,0] )

T(1,:) = T(2,:) x T(3,:)

## Initialization steps (to be done on EV\_CONFIG, before first matrix calculation)

Step 1: **CalMatInt** = CalMat, interpolated to actual frequency bins (nbins). Can be done “virtually” in a function to save memory.

Step 2: Precalculate CM\_BE: 3 x 5 x nBins complex calibration matrices for BE submatrix

**CM\_BE**(i=1..2, j=1..5,f) = CalMatInt(i,f)\*conj(CalMatInt(j+3,f)), where f = 1..nbins;

Step 3: Precalculate CM\_BB: 3 x 3 x nBins calibration matrices for BB submatrix and EE submatrix

// Calibration matrix for BB submatrix

**CM\_BB**(i=1..3, j=1..3,f) = CalMatInt(i,f)\*conj(CalMatInt(j,f)), where f = 0..nbins-1;

// CM\_EE: 3x3 calibration matrix for trace. Only needed if (length(indicesTrace) == 3)

**CM\_EE**(i=1..3.j=1..3,f) = CalMatInt(3+indicesTrace(i),f)\*conj(CalMatInt(3+indicesTrace(j),f))

// Calibraiton coefficients for diagonal of E

**CM\_Ediag**(i=1..5,f) = CalMatInt(i+3,f)\*conj(CalMatInt(i+3,f))

Step 4: Initialize averaged matrix BB\_avg(i,j,f), trace of E components E\_trace(f) and Poynting component Sz(f) to zero.

## Routine processing, for every SM matrix received:

Step 1: From FPGA, SW receives an a matrix **M\_fpga** (8 x 8 x nbins). Apply averaging in frequency, if applicable to get “nBins” frequency bins.

Step 2: Generate calibrated sub-matrices EB and BB

BB(i,j,f) = M\_fpga(i,j,f) \* CM\_BB(i,j,f), for all i, j, f - element-wise multiplication

BE(i,j,f) = M\_fpga(i,3+j,f)\*CM\_BE(i,j,f), for all i, j, f - element-wise multiplication

And diagonal of calibrated E-field matrix:

E\_diag(i,f) = M\_fpga(3+i,3+i,f)\*CM\_Ediag(i,f);

Step 3: Software waits until JMAG B field, spanning the time when M\_fpga was averaged, is available.

**B\_jmag\_mag** = JMAG vector averaged over this interval (in MAG instrument coordinates).

Step 4: Using B\_jmag, calculate **TM\_sc\_mfa**

// Convert JMAG B from JMAG coordinates to SC coordinates

**B\_jmag\_sc = TM\_mag\_sc\* B\_jmag\_mag;**

// Create transformation matrix

T(3,:) = B\_jmag\_sc / norm(B\_jmag\_sc);

T(2:,) = B\_jmag\_sc x [1,0,0] / norm(B\_jmag\_sc x [1,0,0] )

T(1,:) = T(2,:) x T(3,:)

**TM\_sc\_mfa =** T**;**

Step 5: Calculate

**TM\_scm\_mfa** = TM\_sc\_mfa\*TM\_scm\_sc;

Step 6: Transform B components and antenna directions to MFA

for all **f** do:

BB\_mfa = TM\_scm\_mfa\*BB\*conj(TM\_scm\_mfa)

BE\_mfa = TM\_scm\_mfa\*BE

Ant\_dir\_mfa = TM\_sc\_mfa\*Ant\_dir\_sc; // Ant\_dir\_\* are unit vectors

Step 7: Calculate projections of Poynting vectors to JMAG B-field direction (Z-axis in MFA)

numavg = number of averaged FFTs in FPGA (constant);

for all frequencies **f** do:

Sz(**f**) = 0;

for **i** in <indicesED>:

for **j**=1,2

// Calculate variance (sigma^2) of each element of BE\_mfa

// Use a formula from Priestley et al. (p. 702)

var(BE\_mfa(**j**,**i**)) = BB\_mfa(**j**,**j**)\*E\_diag(**i**) + real(BE\_mfa(**j**,**i**))^2 - imag(BE\_mfa(**j**,**i**))^2;

end

// calculate a projection of the Poynitng vector to MFA Z-axis

// Ant\_dir\_mfa(i,j) is the i-th component of the unit direction vector of j-th antenna in MFA

Sz\_proj = real(BE\_mfa(2,i)) \* Ant\_dir\_mfa(1,i)  **-** real(BE\_mfa(1,i)) \*Ant\_dir\_mfa(2,i)

// calculate a normalization factor

normf = sqrt((var(BE\_mfa(2,i))\*Ant\_dir\_mfa(1,i)^2 + var(BE\_mfa(1,i)) \* Ant\_dir\_mfa(2,i)^2) / numavg);

// accumulate the projections for all antennas

Sz(f) += Sz\_proj / normf;

end

end

Step 8: Add to averaged values

for all f,

BB\_avg(:,:,f) += BB\_mfa(:,:,f);

Sz\_avg(f) += Sz(f);

// EE\_avg64 (I64 3x3 matrix): akumuluje se nekalibrovana 3x3 sub-matice EE podle indicesTrace

EE\_avg64(:,:,f) += M\_fpga(3+indicesTrace,3+indicesTrace,f);

endfor

## Final processing after averaging of all matrices is completed:

Step 1: Calculate E\_trace: Trace of E components averaged. If exactly 3 E components are selected, transform to orthogonal components before computiong trace.

for all f do:

EE\_avg(i,j) = EE\_avg64(i,j,f) \* CM\_EE(i,j,f), for all i, j

len = length(indicesTrace);

if (len < 3)

E\_trace(f) = sum(EE\_avg(i,i,f)), for i=0:len-1

else // only if (length(indicesTrace) == 3)

EE\_trans = TM\_ant\_sc\*EE\_avg\*conj(TM\_ant\_sc);

E\_trace(f) = sum(EE\_trans(i,i,f)), for i=0:len-1

end

Step 2: Generate a packet with the following 4-byte structure for every frequency bin:

// Total number of averaged FFT spectra (FPGA & SW averaging combined)

tot\_num\_avg = numavg\*num\_averaged\_matrices;

**B\_trace** = trace(BB\_avg)/tot\_num\_avg, converted to 8 bits via Util\_LogEncode32to8

**E\_trace** = E\_trace/tot\_num\_avg, converted to 8 bits via Util\_LogEncode32to8

Run SMX\_Prassadco(BB\_avg) to get:

float k\_vector[3], ellipticity, planarity (ellipticity = -1 to 1, planarity 0..1, |k\_vector| = 1)

Reduce to the following structure:

**theta** (4 bits 0-90deg, resolution 5.625deg): Angle between of k\_vector and B0 magnetic field.

Theta [radians] = arccos(abs(kz)); // assuming k\_vector = [kx, ky, kz] a |k\_vector| = 1

**Phi**  (4 bits, -180-180deg, 22.5deg resolution): Azimuthal angle of k\_vector “around” B0.

Phi [radians] = atan2(ky, kx); // check if atan2 works OK if ky == 0 or kx == 0

**ellipticity** (3bits, signed -1..1),independent of B0, step 0.25

**Planarity (3bit, unsigned, between 0-1),** step 0.125, independent of B0

**Sz\_avg (2bit):** Sz\_avg/num\_averaged\_matrices reduce to 2 bits by comparing absolute value to a defined threshold (set in config) and keeping the sign. If below threshold => 0 or -1, if above threshold => -2 or 1.

Reset BB\_avg, Sz\_avg and EE\_avg64 to zeros and start new averaging

# Appendix 2: Defaults configurations in SW2.0

## Built-in LF configurations

The table below lists the default LF configurations hard-coded in SW2.0.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Cfg # | Name | SIDs generated | MUX config | Description |
| 0 |  |  |  |  |
| 1 | Long snapshot trigger |  | Dipole\_LP | Triggering with long snapshots @ 24 kHz.   * 3xB DWFS@763Hz, continuous (37 kbit/s) * SMX 3B+3E, 1 x per 32s, 40 log bins. * BP2, 3B+3E, 1 x per 1s, 40 log bins. * STAT * TSWF 7x96k samples, 1x per 2 minutes cycle (90 kbit/s)   Triggering on AD5 (E2-E3). Preference waves |
| 2 | LF trigger 2P dipole | 4, 6, 9, 34, 39 | Dipole\_2P | Similar to config5, but only with 2 probes (P3/P4) enabled. Data products reduced.   * no BP, SMX 4x4 every 8 sec |
| 3 | LF trigger 2P dipole 24kHz | 4, 6, 9, 34, 39 | Dipole\_2P | Same as config2, 24k kHz sampling |
| 4 | Linear BP2 spectra | 6, 39 | Dipole\_LP | BP2 linear spectra (4 per second), , 24 kHz sampling   * 3xB DWFS@763Hz, continuous (37 kbit/s) * BP2, 120 linear bins, 0.2-11.5 kHz, averaged over 250 ms (17 kbit/s). * No snapshots, no trigger. * No JMAG or RW data used |
| 5 | LF trigger 4P dipole | 4 (SMX),  6 (BP1),  9 (STAT),  34 (TSWF),  39 (DSWF) | Dipole\_LP | Standard dust detection mode. 49 kHz sampling   * 3xB DWFS@763Hz, continuous (37 kbit/s) * SMX 3B+3E, 1 x per 32s, 40 log bins (~0 kbit/s). * BP1 3B+3E, 1 x per 4s, 20 log bins (~0 kbit/s). * STAT * TSWF 8x16k samples, 1x per 30s cycle (60kbit/s) * Triggering on AD4 (E1)   No JMAG, no RW |
| 6 | LF trigger 4P mono | 3, 4, 6, 9, 34, 39 | Mono\_4P | Same as config5, but with monopole MUX configuration (use AD5678).   * TSWF 7x16k samples, 1x per 30s cycle * Triggering on AD8 (E4) |
| 7 | LF trigger 4P dipole 24kHz | 3, 4, 6, 9, 34, 39 | Dipole\_LP | Same as config5, but with sampling reduced to 24 kHz configuration. |
| 8 | EMC / test mode | 33, 39 | Mono\_4P | Generates regular snapshots 7x8k every 5 seconds + continuous DSWF. **Should be run without sequencer.** |
| 9 | SCM sweep config | 33, 39 | Mono\_4P | SC config sweep. Takes 1 long snapshot + DSWF. **Should be run 2 minutes with or without sequencer.** |
| 10 | LF density 4P 24 kHz |  | Dens\_4P | Heaven mode - triggered snapshots 32k @ 24 kHz, no spectra   * 3xB DWF@24Hz * 3xB DWFS@763Hz, continuous * STAT * TSWF 7x32k samples, 1x per 30s cycle   Triggering on AD5 (D1). |
| 11 | No trigger mode dipole | 4 (SMX),  6 (BP1),  34 (TSWF),  39 (DSWF) | Dipole\_LP | 49 kHz No internal triggering. TSWF is taken on conclude event.   * 3xB DWFS@763Hz, continuous * SMX 3B+3E, 1 x per 32s. * BP1 3B+3E, 1 x per 8s. * TSWF 7x16k samples, 1x per 30s cycle |
| 12 | No trigger mode monopole |  | Mono\_4P | Same as cfg 11, monopole config. |
| 13 | Test mode | 4 (SMX),  6 (BP2),  9 (STAT),  34 (TSWF),  39 (DSWF) | Mono\_4P | Hell mode for testing of RW and JMAG on-board configuration:   * 24 kHz sampling * 3xB DWFS@763Hz, continuous * BP2 every 1 sec, 40 log bins (1.3 kbit/s) * SMX 3B+3E, 1 x per 1s., 40 log bins * TSWF 7x32k samples, 1x per 30s cycle   Enabled RW and JMAG data use in BP2  BP2 synchronized with snapshots |
| 14 | Earth-Ganymede mode | 4 (SMX),  8 (BP2),  9 (STAT),  34 (TSWF),  39 (DSWF) | Dipole\_LP | Earth flyby whistler mode testing the Ganymede orbit. High resolution BP2 spectra + long snapshots,   * 24 kHz sampling * 3xB DWFS@763Hz, continuous (37 kbit/s) * SMX 3B+3E, 1 x per 28s, 120 lin bins (1.5 kbit/s). * BP2, 4 x per sec, 120 lin bins (17 kbit/s) * STAT * TSWF 7x96k samples, 1x per 60s preference waves (180 kbit/s) * No JMAG, no RW   Triggering on AD5 (E2-E3) |
| 15 | Survey minimum TM | 8 (BP2) | Dipole\_LP | Low resolution BP2 spectra + no snapshots,   * 24 kHz sampling * BP2, 1 x per sec, 40 log bins(1.3 kbit/s) * No JMAG, no RW |

## Standard E-field mux configs used in the above LF configurations

Table ‑: Configuration of multiplexers used in standard configurations. Highlighted components are the ones which are usually transmitted (others are filtered by software)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Name | AD4 | AD5 | AD6 | AD7 | AD8 | Register values / SW selection |
| Dipole\_4P | E1 | E1-E3 | E2-E4 | E2-E1 | SUM | SW= (031f, 803F, 001f, 8112)  3 differential signals, 1 x SUM, 1 x E1 |
| Mono\_4P | E1 | E1 | E3 | E2 | E4 | SW= (031f, 8000, 021f, 8DEE)  4 single probe signals |
| Dipole\_2P | E1 | -E3 | **E3-E4** | E2 | **E4** | SW= (031f, 8000, 021f, 8D23)  Only P3 and P4 used, single probe signal + difference |
| Dens\_4P | E1 | D1 | D3 | D4 | D2 | SW= (031f, 803f, 051f, 8777)  4 single probe signals in density mode |
| Dipole\_LP | E1 | E2-E3 | E3-E4 | E2-E1 | SUM | SW= (031f, 803F, 001f, 8121)  Compatible with LP configuration  3 differential signals, 1 x SUM, 1 x E1 |
| Dipole\_uber | E1 | E2-E3 | E2-E4 | E4-E1 | SUM | SW= (031f, 803F, 001f, 8211)  Super awesome dipole configuration  3 differential signals, 1 x SUM, 1 x E1 |

## Use of LF configurations in RPWI operational sequences

This table lists the proposed default assignment of LF configurations to the LF sequences.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **seq#** | **LF** | **LP** | **Radio** | **MIME** | **LF Comment** |
| **1** | LF01 | 11.1 | Radio Burst | MIME Sweep | 96k snapshot 3D 24kHz |
| **2** | LF02 | 11.1 | Radio Burst | MIME Tracking | 16k snapshot 2D 49kHz |
| **3** | LF15 | 10.0 | Radio Full | MIME OFF | minimum TM survey |
| **4** | LF14 | 11.2 | Radio Full | MIME Sweep | Earth and Ganymede mode 1 |
| **5** | LF03 | 11.1 | Radio PSSR-1 | MIME Sweep | 16k snapshot 2D 24kHz |
| **6** | LF05 | 11.1 | Radio PSSR-2 | MIME Sweep | 16k snapshot 3D 49kHz |
| **7** | LF06 | 11.1 | Radio PSSR-3 | MIME Sweep | 16k snapshot 3Dmono 49kHz |
| **8** | LF07 | 11.2 | Radio PSSR-3 | MIME Wide Sweep | 16k snapshot 3D 24kHz |
| **9** | LF01 | 11.1 | Radio PSSR-3 | MIME Tracking | 96k snapshot 3D 24kHz |
| **10** | LF11 | 11.1 | Radio Full | MIME B-field | No trigger |
| **11** | LF04 | 11.2 | Radio Full | MIME Sweep | whistler test 1 |
| **12** | LF12 | 11.1 | Radio Full | MIME Tracking | No trigger mono |
| **13** | LF01 | 11.1 | Radio Burst | MIME B-field | 96k snapshot 3D 24kHz |
| **14** | LF10 | 11.4 | Radio full | MIME OFF | density interferometry |
| **15** | LF14 | 11.2 | Radio Full | MIME Wide Sweep | Earth and Ganymede mode 2 |
| **16** | LF04 | 11.2 | Radio Burst | MIME Wide Sweep | whistler test 2 |
| **17** | LF13 | 11.3 | Radio Full | MIME Wide Sweep | LF on-board processing advanced configuration test mode |
| **18** | LF01 | 11.3 | Radio Burst | MIME Wide Sweep | 96k snapshot 3D 24kHz |
| **19** | LF01 | 11.1 | Radio Burst | MIME Tracking | 96k snapshot 3D 24kHz |
| **20** | LF10 | 11.4 | Radio Burst | MIME OFF | density interferometry |