Table 7. Steps of SWA-PAS characterisation and calibration.

#	Action	Comment
1	CEM lifetime characterisation	
2	Detection system integrity	Check the signal propagation from the CEM to the telemetry with the real ion beam
3	CEM characterisation	CEM uniformity, working biases, etc.
4	Sensor optimisation	Optimal HVs for each elevation step
5	"Static" calibration	Determine SWA-PAS properties with quasi-static HVs
6	"Flight-like" calibration	Determine SWA-PAS properties with flight sweeping HVs

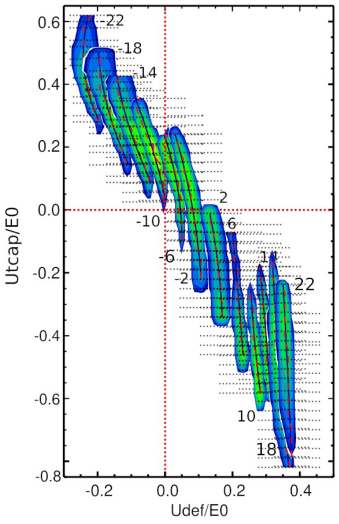

Fig. 24. SWA-PAS count rate (colour coded) as a function of "top-cap" voltage (Utcap, vertical axis) and deflector voltage (Udet, horizontal axis), normalised to the central energy of the hemisphere analyser (E0), for several elevation angles (labelled values adjacent to the responses). Black dots indicate the points of the measurement.

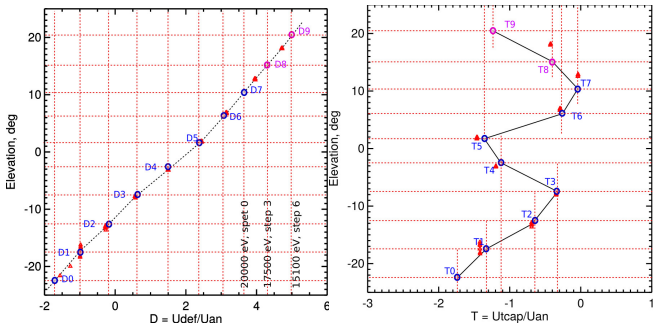

Fig. 25. SWA-PAS optimisation results that provide the basis for FPGA sequencer code inputs. *Left panel*: variation of the elevation angle as a function of the ratio between the voltage on the deflector (Udef) and that on the analyser (Uan), *right panel*: variation of the elevation angle as a function of the ratio between the voltage on the top cap (Utcap) and Uan.

and 9.3%, a range that brackets the target resolution of 5.5%. This deviation is a result of the trade-off between the instrument geometrical factor and the energy resolution. This is a generic property of the instrument which has a thin slit as an entrance aperture and a deflector behind this slit. The elevation range is

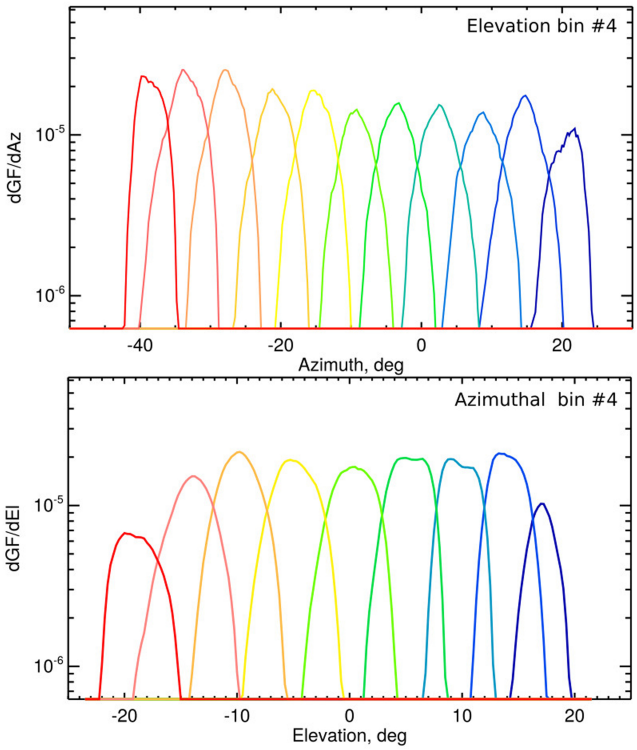

Fig. 26. *Top*: SWA-PAS azimuth responses for the elevation bin #4. This shows the differential geometric factor of each azimuth bin as a function of azimuth angle. *Bottom*: SWA-PAS elevation responses for Azimuthal bin #4. This shows the differential geometric factor of each elevation bin as a function of elevation angle.

Table 8. SWA-PAS design target goals with build and calibrated properties for PFM instrument.

Parameter	Range and resolution	Design goal	Build and calibration
Energy	Range	200 eV–20 keV	70 eV–20 keV
	Resolution ($\Delta E/E$)	5.5%	3.0–9.3%
	Number of steps	96	92
	Analyser constant (eV/V)	13	13–14
Angle	Azim. range	−24° to 42°	−24° to +42°
	Elev. range	±22.5°	−20° to +23°
	Azim. resolution	<6°	5°
	Elev. resolution	<6°	5°
Temporal	Basic accumulation period	~1 ms	0.95 ms
	Normal mode	4 s moments and 3D VDFs	4s moments and 3D VDFs
	Burst mode	15 VDFs/s	15 VDFs/s
Sensitivity	Per angular bin (cm^2 sr eV/eV)	≥4×10^{-6}	From 4×10^{-6} to 6×10^{-6}

slightly asymmetric. The 3D sampling energy range used for an individual measurement is reduced to 92 steps, although there remain 96 steps covering the total possible range. Other properties are very close to the corresponding targets.

3.4. The SWA Data Processing Unit (SWA-DPU)

3.4.1. SWA-DPU introduction

The SWA Data Processing Unit (SWA-DPU, Fig. 27) is the "heart" of SWA. It represents the only SWA interface with

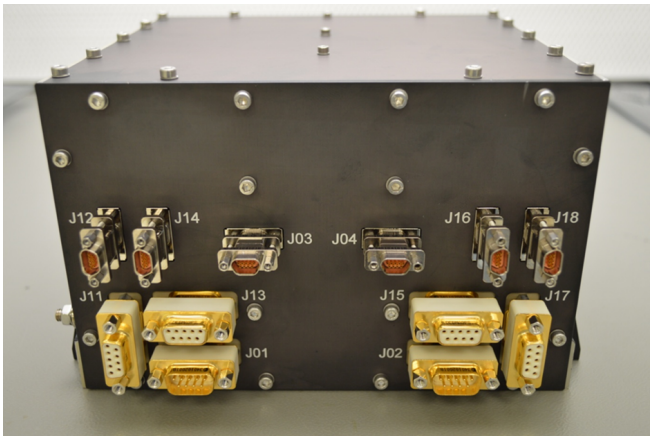

Fig. 27. SWA-DPU flight model hardware.

the spacecraft as far as commanding and communication are concerned: the SWA-DPU implements the data and command interfaces with the spacecraft via redundant SpW data links and the redundant HV-high power command (HV-HPC) interfaces. The SWA-DPU is also interfaced with SWA-EAS, SWA-PAS, and SWA-HIS sensors via SpW dedicated links. For SWA-EAS and SWA-PAS this interface supports functionality control, temporary storage, communication, and computational capability. In addition, the SWA-DPU will support SWA-HIS with communication to the spacecraft. The SWA-EAS and SWA-PAS sensors receive redundant power inputs from the SWA-DPU, although the SWA-HIS sensor draws power directly from the spacecraft bus.

3.4.2. SWA-DPU hardware design

The SWA-DPU architecture is derived from trade-off analyses to define a system able to perform the needed computational tasks while keeping mass, volume, and power consumption within the constraints imposed by spacecraft resources. Additionally, the SWA-DPU has been designed to be "single fault" tolerant and the "cold-spare" concept has been adopted as the redundancy philosophy. The SWA-DPU hardware architecture is based on:

- 1 × backplane and SpW splitter (BSS) front-end interface circuits;
- 2 × data processing module (DPM), each composed of:
  - 1 × communication and scientific data processing (CSP) Board;
  - 1 × central processing unit (CPU) Board;
- 2 × power conditioning and distribution module (PCDM).

Apart from the BSS, representing the SWA-DPU redundant communication node, the SWA-DPU is comprised of two sub-units (SWA-DPU-nominal and SWA-DPU-redundant), each of which includes one DPM and one PCDM and is capable of implementing separately all the functions of the SWA-DPU (see Fig. 28 for details). The BSS is a rigid-flex board that, in addition to the backplane board, includes the SpW front-end circuits serving the SWA sensors and the spacecraft communication interfaces (nominal and redundant); these circuits provide connections for the two DPMs, deployed in cold redundancy, and each SpW data link must have a connection to both the DPMs. For this purpose, each of the six independent SpW physical layer "splitter" circuits act as a repeater and distributor on two independently buffered outputs. Each splitter is composed of LVDS drivers and receivers, a local regulated power supply, and sectioning circuits. This design avoids fault propagation, ensuring the isolation of possible failure. It also provides independent switch-on and off capability for each splitter, thus saving power when the corresponding interface is unused. The SWA-DPU data communication and processing capabilities, provided by the DPM, are realised by the joint use of a CPU board and a FPGA-based CSP board. The latter acts as communication controller and can be also used as a processing engine for intensive computation. This architecture allows the implementation of a wide range of complex processing algorithms, by adopting appropriate hardware and software partitioning solutions.

The CPU board is based on the Leon2FT ASIC AT697F processor by ATMEL, running at 100 MHz and working under the VxWorks Real Time Multitasking Operating System. The CPU is equipped with EDAC-protected volatile and non-volatile memories, including a 64 Kbyte Boot EPROM, a 2 Mbyte + 2 Mbyte MRAM for onboard software and permanent data, and an 8 Mbyte SRAM for code execution and run-time data.

The CSP board is based on two RTAX2000 FPGAs (primary and secondary). It also has a 2.5 Gbit synchronous dynamic random access memory (SDRAM) mass-memory needed for buffering of the scientific data generated by the SWA-EAS and SWA-PAS sensors. The primary FPGA is directly interfaced with the CPU and the SDRAM, performing all the SWA-DPU communication tasks and the data handling functionalities not covered by the CPU. The secondary FPGA is used as a possible hardware accelerator, dedicated (as needed) to the implementation of parts of algorithms, in order to reduce CPU overhead and to ensure a significant margin in the overall SWA-DPU processing capabilities.

Each DPM module is supplied with a separate and protected power supply line from its associated sub-unit PCDM. It also directly interfaces with the PCDM via a SPI interface, for reception of HK data and for power distribution management for the SWA sensors by means of low-level control and status signals.

The PCDMs primary role is to implement the power-related functionalities, including: (i) power inputs (main and redundant) interface front-end, including EMI filter, inrush current limiter, protection circuits, etc; (ii) interface for the external HV-HPC commands from the spacecraft and BSM status towards the spacecraft (main and redundant); (iii) power conditioning, distribution, and protection for the internal SWA-DPU electronics; (iv) power distribution and protection for the sensors, by means of power switches controlled by opto-isolated circuits; and (v) acquisition of HK data related to currents and temperatures.

3.4.3. SWA-DPU software design

The overall SWA-DPU software consists of two separate independent executable software images: the boot software (BSW) and the FSW. The top-level software architecture for the SWA-DPU is shown in Fig. 29. The BSW is non-patchable software resident in a PROM. At start-up, the BSW performs the basic hardware initialisation, namely processor and memory controller initialisation. It then starts a memory check, which consists of a write and read test of the SRAM (8 MB) and an integrity check of the FSW stored in MRAM, based on the FSW code cyclic redundancy check (CRC) verification. Should the SRAM test fail, the redundant section of the SWA-DPU will be used. Should the FSW code CRC test fail, the BSW enters its diagnostic mode and waits for ground intervention. Upon successful execution of SRAM and MRAM tests, the BSW loads the FSW from MRAM to SRAM and starts it. Upon entering the diagnostic mode,

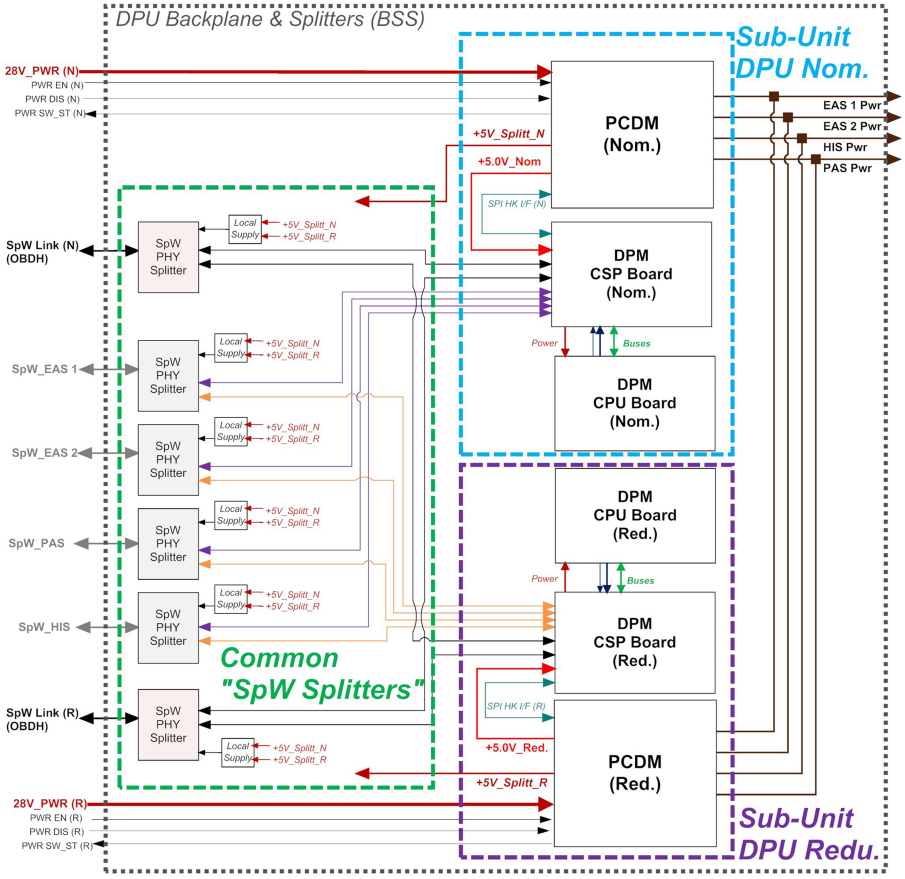

Fig. 28. Schematic of the SWA-DPU hardware architecture.

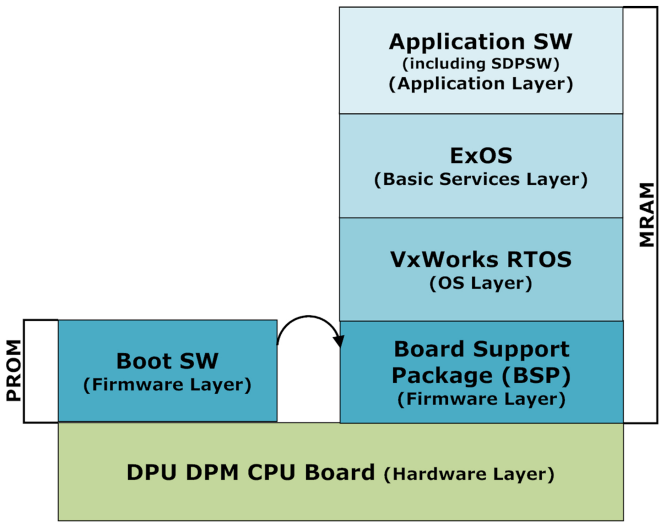

Fig. 29. SWA-DPU top-level software architecture.

the BSW initialises the communication through the main and redundant SpW interfaces with the spacecraft on board computer (OBC) in order to manage a reduced set of TM and TC services. These include Service 9 TC, to synchronise the SWA-DPU internal reference time with the spacecraft event time (SCET), Service 3 TM, to generate and transmit the SWA-DPU HK telemetry, and Service 6 TC, to perform check, dump, and load of memory blocks (for FSW patching).

The FSW is composed of three software layers: the application software layer (ASW), the application service software layer (ASSW), and the machine services software layer (MSSW). The ASW includes SWA-DPU TC and TM and process control management software (PCMSW), and the scientific data processing software (SDPSW). The PCMSW is a logical composition of sub-components which manage TC validation and execution, SWA-DPU and sensor state management, the SWA sensors (SWA-EAS1, SWA-EAS2, SWA-PAS, and SWA-HIS) themselves, and the science data acquisition. It also controls algorithm execution on acquired scientific data (i.e. moment computation, compression, etc.), SWA-DPU HK data handling, FDIR, and time management. The SDPSW provides post-processing functionalities on the SWA-EAS and SWA-PAS acquired science data (moment calculation, raw data compression, formatting of compressed stream for downlink). The ASSW includes the packet utilisation standard library software (PUS) and mission services for the ASW. It provides the set of functions for implementing the PUS services required for the SWA-DPU ASW. The MSSW includes the real time operating system (RTOS) kernel, the extended operating system service (EXOS) software, and the low level drivers in the board support package (BSP). The RTOS provides the SWA-DPU DPM runtime resource management, the basic mechanisms for task execution, and inter-process communication. The EXOS provides a library used by the ASW and ASSW layer components to gain access to the SWA-DPU hardware resources (memory, registers, time, SpW, sensors, etc.). The EXOS also efficiently executes some basic functions for the configuration and management of the system and its various subsystems. It provides appropriate functionalities to support the exchange of information among the subsystems and between the SWA-DPU and spacecraft. The BSP provides access services to all SWA-DPU internal hardware and physical interfaces of the system to the above software layers.