

Solar Probe Plus

A NASA Mission to Touch the Sun

Integrated Science Investigation of the Sun Energetic Particles

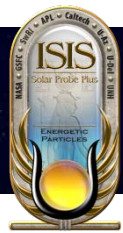
Preliminary Design Review

05 – 06 NOV 2013

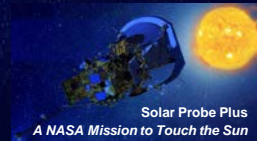
EPI-Hi Flight Software

Andrew Davis, Rick Cook

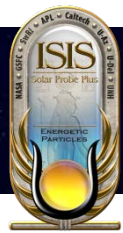




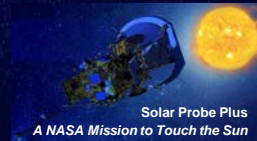
Outline



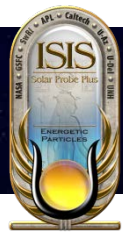
- Software requirements, software design and functionality, software documentation and unit testing as well as plans for software integration, testing and verification at the instrument level
- Plans for software maintenance for the duration of the mission
- Description of the software development environment and plans for preserving the development environment
- Maturity of the design
- Summary and follow-up from peer review held prior to the instrument PDR



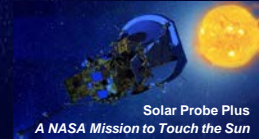
Software Requirements



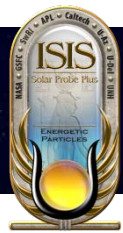
- The Flight Software Requirements Document details the flow-down from higher-level requirements, and from GIS and SIS ICDs
- These requirements are quite similar to those for our work on recent NuSTAR and STEREO missions
- The same modular flight processor and software architecture that we used for NuSTAR and STEREO will meet the EPI-Hi requirements
- The requirements are mapped into a set of modular software tasks, implemented on the EPI-Hi flight processors



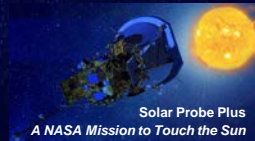
Documentation Status



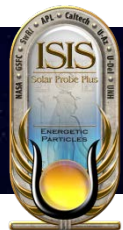
Document	Status/due date
Flight Software Requirements	Baseline draft Delivered
Flight Software Development Plan	Baseline draft Delivered
Flight Software Design Document	Prelim draft delivered at PDR
Flight Software Test Plan	Due by CDR
Test verification matrix	Due by CDR
Instrument Telemetry Data Format Document	Due by CDR
P24 MISC Processor Manual	Delivered



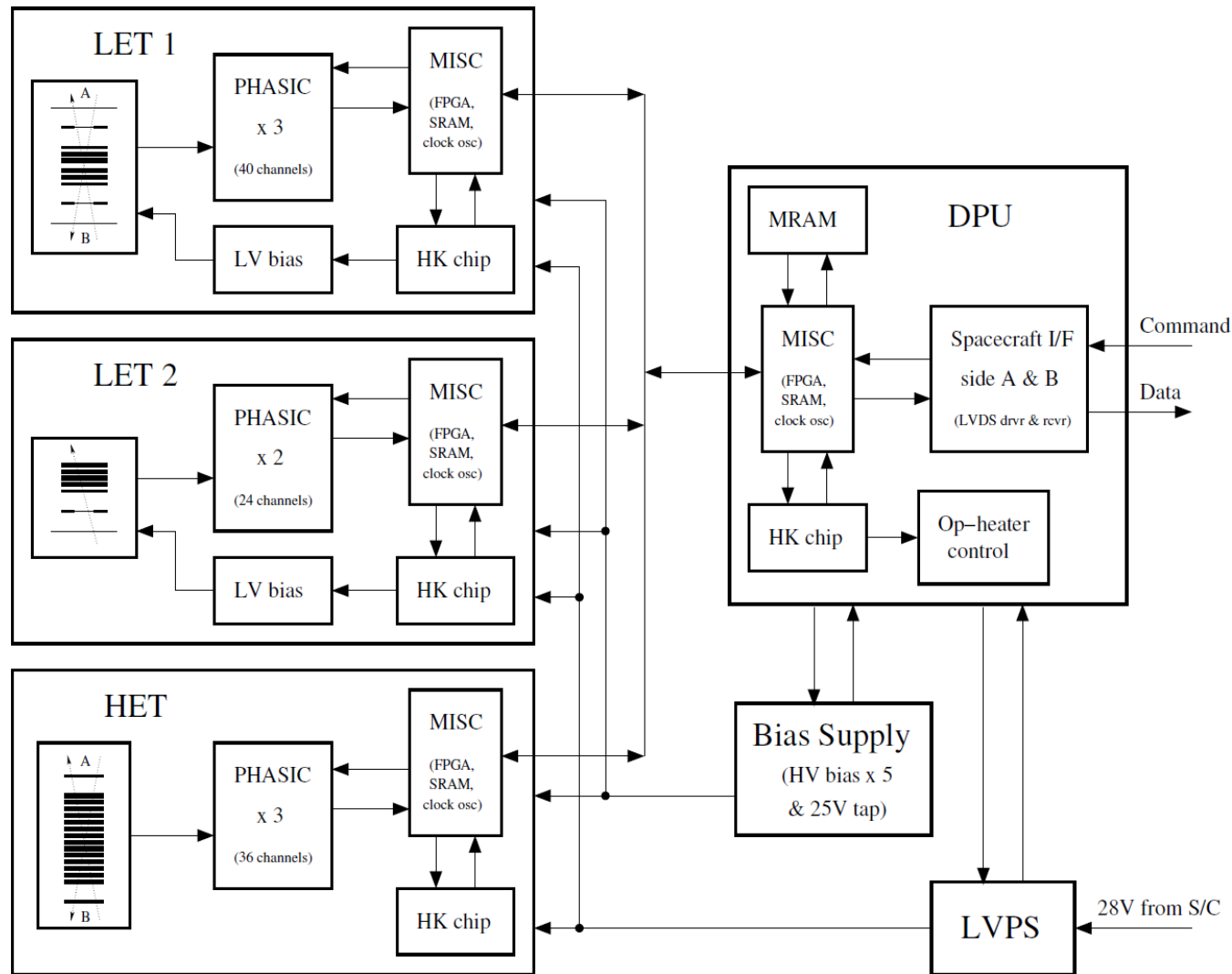
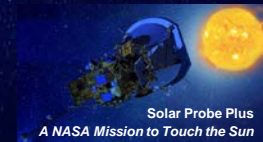
Processor and Software Architecture



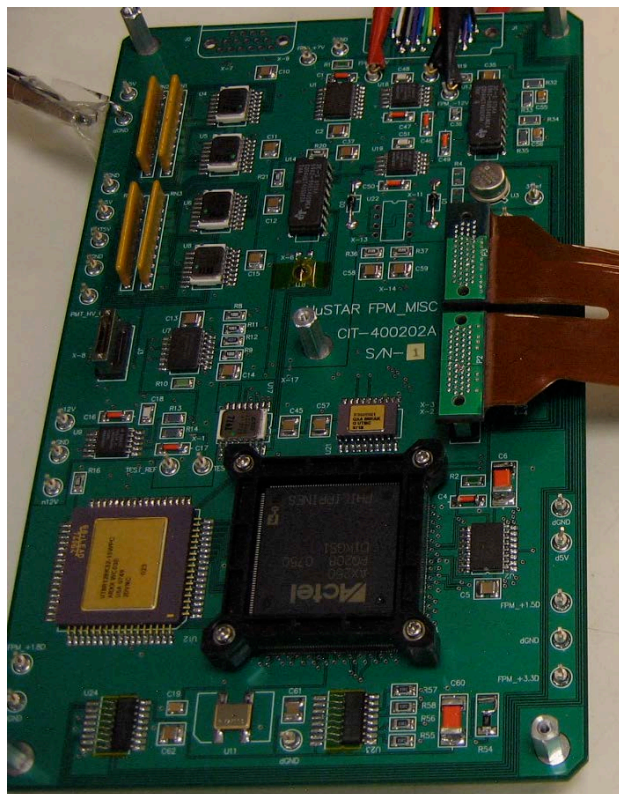
- Software distributed across 4 Minimal Instruction Set Computers (MISCs)
- 1 MISC in DPU, 1 each in LET1, LET2, HET
- Custom FPGA-embedded micro-controller
- Design implemented in Actel RTAX250SL
- Architecture used in STEREO and NuSTAR space missions
- 24-bit CPU, up to 512 Kwords SRAM, Forth operating system, can boot from MRAM or serial interface
- Only DPU MISC has MRAM (2M×8).
- DPU can boot from MRAM, or via serial uplink from ground
- The three “satellite” MISCs communicate with DPU MISC via RS-422 serial links, 460.8kbps
- Satellite MISCs booted via serial link by DPU MISC
- Satellite MISCs use boot images from DPU MRAM



Processor and Software Architecture

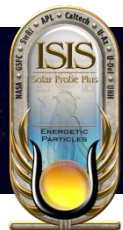


Development Environment

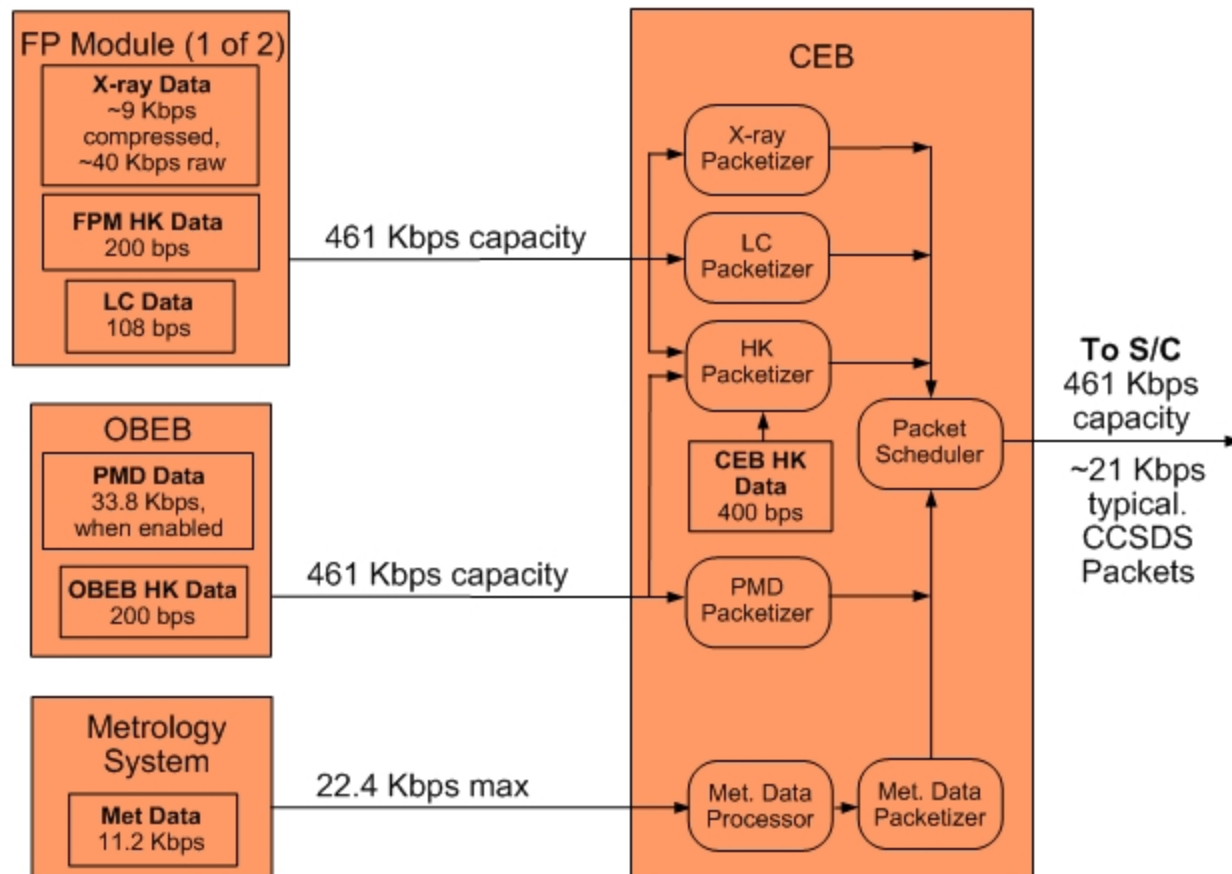
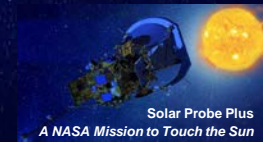


NuSTAR MISC EM board

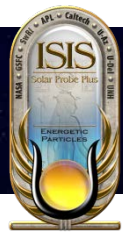
- Initial testing of new code takes place on MISC development boards
- Forth system on development boards accessed via serial link to PC
- New code is easily uploaded via the serial link
- Testing of code that accesses hardware is done on EM units
- Low-level MISC I/O routines are in everyday lab use, where MISCs are used to acquire test data and control test runs
- The same central MISC / multiple satellite MISC architecture was used for NuSTAR and STEREO
- EM units are preserved after launch, and used for testing/verification of software uploads and command sequences



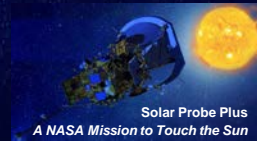
Instrument Data Flow



NuSTAR version – to be updated for EPI-Hi



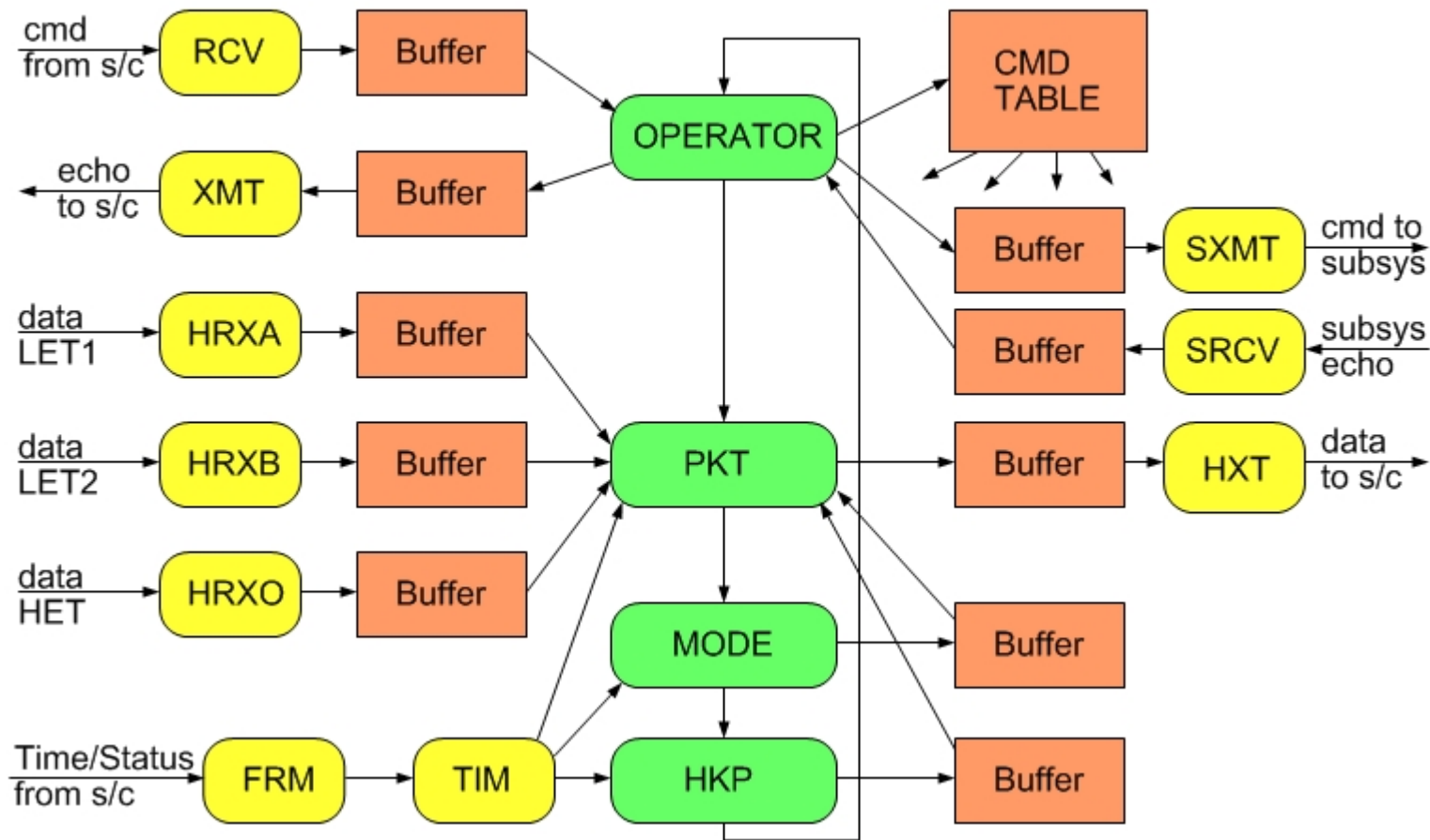
DPU Software Tasks

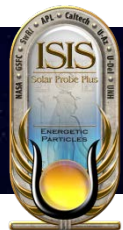


- Forth operating system and low-level I/O routines
- Power-on and initialization sequence management
- Satellite MISC serial boot sequence management
- Housekeeping data collection
- Inter-MISC communication management, and data acquisition/buffering from satellite MISCs
- Setup and control of instrument LVPS, bias supply, heaters, and other electronics systems
- Receive/Monitor status and time-synchronization data from the spacecraft, and perform autonomous mode adjustments as needed
- Management of software uploads and MRAM “burns”
- Formatting and transfer of science & housekeeping data and command responses to the Spacecraft
- Monitor heartbeat from peripheral MISCs, and perform autonomous diagnostics/reboot as needed

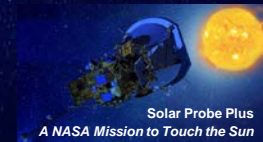
Software Flow – DPU MISC

DPU MISC S/W FLOW

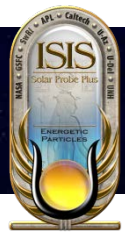




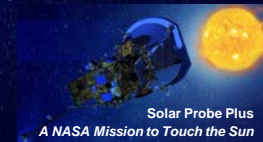
Satellite MISC Software Tasks



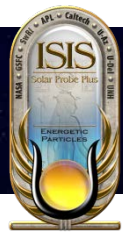
- Forth operating system and low-level I/O routines
- Science data acquisition
- Science data processing and reduction (particle ID)
- Housekeeping data acquisition
- Processing of status, time-synchronization, and command data from the EPI-Hi DPU
- Monitor key counting rates and adjust the telescope operating condition to optimize data quality
- Monitor temperatures and adjust detector gain/offset settings to compensate for temperature variations
- Formatting and transfer of science & housekeeping data and command responses to the EPI-Hi DPU
- Setup and control of any instrument HV, bias supply, heaters, etc. not controlled by the EPI-Hi DPU.



Software Flow – HET MISC



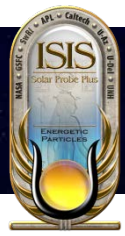
TBD



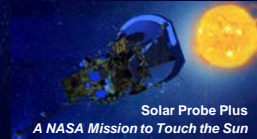
Integration, Testing and Verification



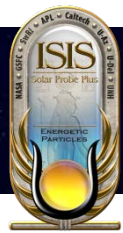
- The EPI-Hi Flight Software Test Plan document will define this plan in detail. This plan will include a software requirements verification matrix.
- Software tests will start at the module level. As the code builds up, system-level testing will start, and the majority of the test time will be targeted at the system level.
- A Test Readiness Review will precede formal acceptance testing. SPP Project and ISIS personnel will oversee/participate in this review.
- Formal acceptance tests will be performed on the EPI-Hi instrument, including the flight software, during the EPI-Hi integration and test phase.
- During environmental tests and SPP I&T, more experience will be gained with the flight software, real sensor data, and with controlling the instrument via the SOC-MOC interfaces. Flight software changes may be expected as a result. Any changes in the flight software at this stage will result in CCB review, and a repeat of these acceptance tests.
- The acceptance tests for the EPI-Hi flight software will be designed to verify each of the software functional requirements as called out in the requirements documents and also the functions provided by the Forth operating system, the I/O API, and the multitasking software running on the MISC processors.
- A software requirements verification matrix will be created and maintained by the EPI-Hi team to aid in verifying that the software meets the requirements. The matrix will be available for review by the SPP Project. A preliminary software requirements verification matrix will be presented at CDR and a final version will be presented at the Software Acceptance Review, following successful completion of the acceptance tests.
- A Software Acceptance Review will follow the successful completion of the acceptance tests, and the EPI-Hi team will present a test report. SPP Project will oversee/participate in this review.



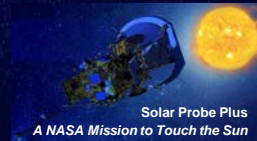
Software Maintenance, Configuration Management



- Version control and software archiving: achieved via Subversion version control system (SVN) (<http://subversion.apache.org/>). The subversion server is hosted and maintained by Caltech Space Radiation Lab IT personnel. It is access-controlled for security, and is backed-up regularly.
- The lead engineer will maintain the structure of the SVN repository, including separate directories for each of the MISC's flight software, separate directories for different software builds, etc.
- All problems and change requests will be documented/tracked in the Software Development Log by the lead engineer. Given such a small dev team, a change request tool such as JIRA is an unnecessary overhead.
- After the beginning of I&T with flight hardware, all flight software changes will be approved by a CCB prior to being loaded into flight hardware. Also after this point, Version Description Documents (VDDs) will accompany all Software releases. The VDD will contain the functionality of the release, a list of closed software problem reports, a list of any liens/workarounds, installation instructions, and a list of deliverable source code files.
- The CCB membership will consist of the EPI-Hi lead engineer and software developer, and the Epi-Hi Project Manager, plus any other ISIS or SPP Project engineers that the QA team at SWRI deems necessary.

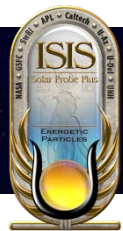


Software Status and Margins

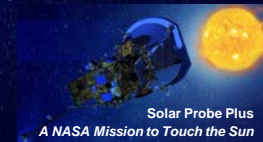


TBD

- Provide estimates of % complete status of the various software tasks
- Provide estimates of CPU, RAM, MRAM margins



Pre-PDR Peer Review Summary



TBD