THE FAST SPACECRAFT INSTRUMENT DATA PROCESSING UNIT

P. R. HARVEY, D. W. CURTIS, H. D. HEETDERKS, D. PANKOW,
J. M. RAUCH-LEIBA, S. K. WITTENBROCK and J. P. McFADDEN

Space Sciences Laboratory, University of California, Berkeley, U.S.A.

Abstract. The Fast Auroral Snapshot Explorer (FAST) is the second of the Small Explorer Missions which are designed to provide low cost space flight opportunities to the scientific community. FAST performs high time resolution measurements of the auroral zone in order to resolve the microphysics of the auroral acceleration region. Its primary science objectives necessitate high data volume, real-time command capability, and control of science data collection on suborbital time scales. The large number of instruments requires a sophisticated Instrument Data Processing Unit (IDPU) to organize the data into the 1 Gbit solid state memory. The large data volume produced by the instruments requires a flexible memory capable of both high data rate snapshots ($\sim$12 Mbit s^{-1}) and coarser survey data collection ($\sim$0.5 Mbit s^{-1}) to place the high rate data in context. In order to optimize the science, onboard triggering algorithms select the snapshots based upon data quality. This paper presents a detailed discussion of the hardware and software design of the FAST IDPU, describing the innovative design that has been essential to the FAST mission's success.

1. Introduction

The Fast Auroral Snapshot (FAST) is the second of the Small Explorer (SMEX) missions. The FAST mission is designed to measure the microscale physics of the auroral acceleration region with the highest time resolution possible. A sophisticated Instrument Data Processing Unit (IDPU) manages the entire science package, collecting and organizing the data before transmission to the ground. The IDPU controls a large number of instruments: 16 electrostatic analyzers to measure charged particle fluxes, a mass spectrometer, 10 electric field and 6 magnetic field sensors. In addition, support electronics include electric and magnetic field sensor signal conditioning and analysis, particle data averaging, and housekeeping monitoring. Since it was known that the envisioned data rates combined with instrument control could easily overwhelm a processor, a design philosophy was chosen to develop autonomous instruments that require little commanding and a data recording system that minimizes processor interaction. Numerous tasks normally performed by the processor have been delegated to custom design circuits using Field Programmable Gate Arrays (FPGAs). With a reduced work load for the processor, the software focuses on data optimization schemes that maximize the science return.

The high data accumulation rate ($\sim$12 Mbit s^{-1}) required during scientifically interesting events generates a data volume far too large for collection (a 40-min

auroral crossing generates $\sim$3600 MBytes), too large for transmission (a typical 20-min contact at 2.25 Mbit s^{-1} allows only $\sim$340 MBytes), and too large for conventional data recorders available at the time FAST was built. The science requirements drove the development of a solid state recorder whose 125 MByte (1 Gbit) size was limited by mass constraints. The recorder was included in the IDPU to eliminate a high bandwidth interface between the spacecraft and the IDPU, and to allow more control of the data recording by the IDPU. The IDPU formats data directly into packets and frames so that it is ready for transmission without further processing. Design specifications required the IDPU to simultaneously record and play back data from several different asynchronous channels, prioritizing the high resolution data for selective transmission to the ground. In order to place the high resolution data in a large scale auroral context, coarse survey data collection was required throughout the auroral zone. To accommodate different data types, the FAST team developed four levels of data collection: slow survey ($<$0.05 Mbit s^{-1}), fast survey ($\sim$0.5 Mbit s^{-1}), burst (5-12 Mbit s^{-1}), and high speed burst (80 Mbit s^{-1}). The IDPU controls the survey data collection rates based upon trigger data, and prioritizes high resolution bursts (5–30 s duration) for selective transmission to the ground. This data optimization scheme has led to high quality, scientifically interesting data being collected on nearly every orbit.

This paper presents a detailed discussion of the hardware and software design of the FAST IDPU. The IDPU consists of a processor board, formatter board, solid state memory, and power system, plus electric and magnetic field signal conditioning and analysis circuits described in Ergun *et al.* (2001). The processor is the primary interface to the spacecraft Mission Unique Electronics (MUE, a dual processor electronics package that monitors spacecraft health, controls telemetry, receives and relays commands, performs attitude control, and manages power, see Pfaff *et al.*, 2001). The IDPU controls and routes commands to the various instruments and IDPU electronics boards, controls the science power system, compiles housekeeping data, directs traffic to the memory, generates packet headers, and manages the onboard data selection using triggering algorithms. The processor also performs one-time tasks such as wire boom deployments (Pankow *et al.*, 2001). The formatter board acts as a router directing data from the particle instruments (Carlson *et al.*, 2001; Klumpar *et al.*, 2001) to the memory. It also forms survey averages of the particle data, compresses the data, and generates particle trigger inputs that are passed to the processor. The solid state memory is a 125 MByte error-corrected high-speed RAM that can be configured to store the multiple data types (survey data, burst data, etc.) at data rates up to 12 Mbit s^{-1}. The power system is designed to provide stable, regulated voltages and contains a convenient power switching relay system that efficiently reduces instrument power when data is not being collected. The innovative design of the FAST IDPU has been essential to the mission success and can be considered a model for integrating multiple spacecraft instruments into a single experiment.
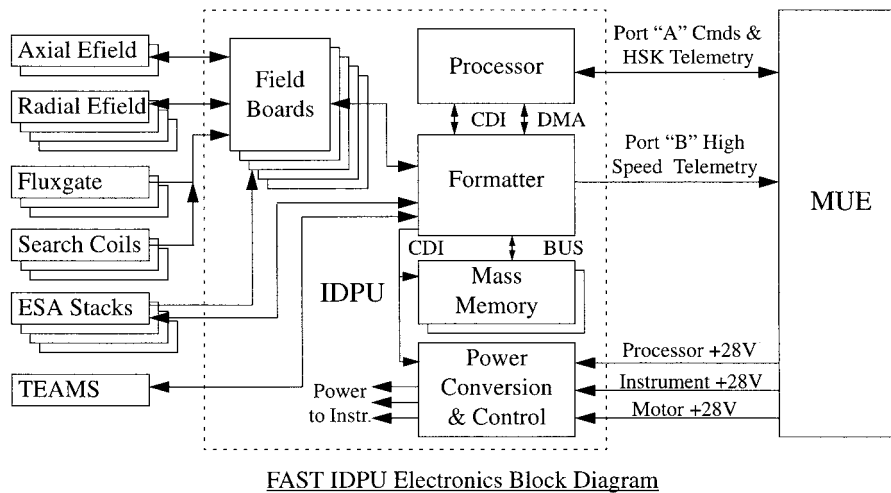
FAST IDPU Electronics Block Diagram

Figure 1. Block diagram of the FAST Instrument Data Processing Unit (IDPU) and interface to the spacecraft Mission Unique Electronics (MUE).

This paper is organized into two main sections covering hardware and software. The hardware section includes descriptions of the processor, formatter, memory and power subsystems. These subsystems form a single, compact electronics box that is housed near the center of the spacecraft to maximize radiation shielding and reduce harnessing. The software section examines the operating system, engineering subroutines, command management, telemetry management, and instrument control. The flight software was written entirely in assembly language to reduce its size and to optimize its performance, and requires a mere 31 KBytes of code space and 60 KBytes of RAM.

2. Hardware Description

The IDPU includes a Processor, Formatter, Mass Memory, and Power section, plus Electric Field and Magnetometer signal conditioning and analysis circuits (see Figure 1). The Processor card provides the intelligence of the system, controlling the flow of raw data through the Formatter and Memory system. The Formatter card performs data collection, averaging, formatting, storage into the memory, and transmission from the memory to the spacecraft with minimal Processor intervention. The Memory system provides 125 MBytes of error-corrected memory used for science and engineering data storage. The Power system provides DC/DC conversion, power distribution, current limiting and monitoring. The Electric and Magnetic Field conditioning circuits are described in a separate paper (Ergun *et al.*, 2001).

IDPU circuitry is packaged in a single box to reduce mass, simplify harnessing and interfaces, and reduce duplication in power converters and other common ser-
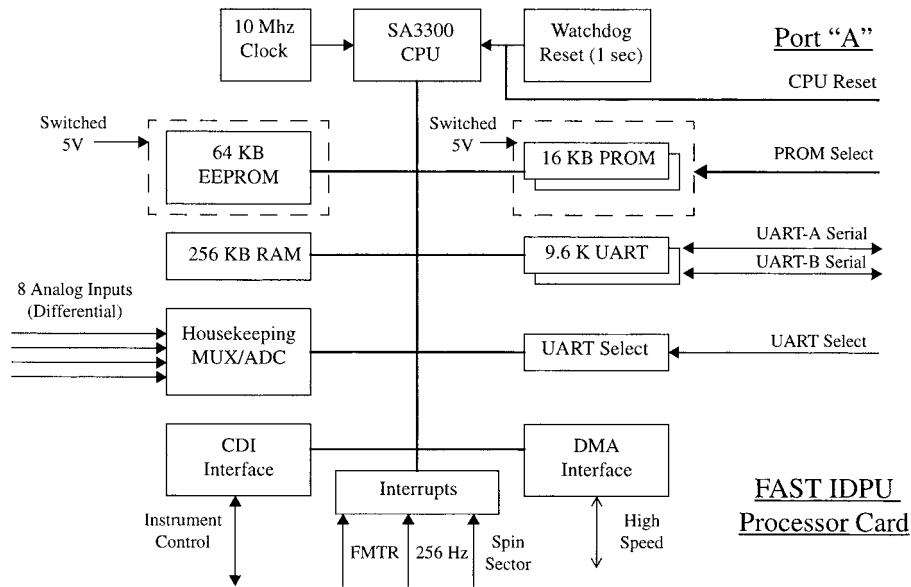
Figure 2. Block diagram of the FAST processor board.

vices. The IDPU makes heavy use of Field-Programmable Gate Arrays (FPGAs) to improve density over a small scale integration (SSI) logic based system, and speed over a processor-based system.

Electronic parts for the IDPU were selected according to criteria for functionality, reliability, and radiation hardness. Moderate reliability levels (e.g., Mil-STD-883C for microcircuits) were coupled with extensive instrument-level testing to verify reliability. Parts were specified and/or lot tested by the FAST team to achieve a minimum tolerance of 20 Krads, and shielded as necessary to 100 Krads. Parts with Single Event Upset (SEU) sensitivity were protected against latchup by current limiting circuits, and against bit-flips by Error Detection and Correction (EDAC) logic.

2.1. PROCESSOR

The Processor system is based upon a Sandia SA3300 32-bit processor, operating at 10 MHz. As shown in Figure 2, the processor card features redundant sections of 16 KByte boot PROM, 64 KBytes of EEPROM, and 256 KBytes of RAM. The card includes a housekeeping A/D system, watchdog timer, serial I/O to the spacecraft MUE, and special interfaces to the Formatter card.

The processor and local memory are rad-hard and SEU immune. The PROM is bipolar, which makes it rad-hard and fast, but power hungry. To save power, the PROM is turned off after being copied to RAM during power-on initialization. Two copies of the PROM code exist, selectable from the spacecraft interface to provide redundancy.

The EEPROM is also power-switched and isolated to save power and to allow the system to operate in case of an EEPROM failure. (The EEPROM is somewhat less rad-hard than the rest of the design, although it meets the mission requirements.) EEPROM is used to hold additional code which can be modified by uplink memory loads after launch as needed. After building the code from the PROM and EEPROM images at power-on, the code is executed out of RAM.

The Processor also includes a 10-bit low-power 'Flash' Analog to Digital Converter (ADC) for digitizing the instrument analog housekeeping. The converter is connected to a tree of analog multiplexers, some on the processor and some remote in the instruments. The multiplexers are controlled by the processor to allow the processor to sample a large variety of housekeeping signals with differing sample rate requirements.

A pair of RS232-style serial interfaces, or Universal Asynchronous Receiver Transmitters (UARTs), on the Processor provide a redundant bi-directional interface between the IDPU and the spacecraft MUE. This interface is used for low rate data (9600 band) such as commands, status, housekeeping, timing, and attitude control information. A spacecraft-controlled line selects which UART to use.

The Processor communicates with the Formatter and all instrument electronics using a custom bi-directional serial Command and Data Interface (CDI). The CDI transfers data at 1 Mbit s^{-1} in 24-bit words, which include an 8-bit destination address and a 16-bit data value. The Formatter and instruments monitor this interface, decode the register address, and latch information sent to them. When Formatter status registers are addressed, the data direction of the serial interface is reversed, and the 16-bit data value is returned to the Processor.

A second serial interface between the Formatter and the Processor transfers large blocks of data between the Processor and the Memory system (via the Formatter) at 4 Mbit s^{-1}. This interface is called the Direct Memory Access (DMA) interface, and is used primarily to transfer engineering packets and science packet headers into the memory for later transmission.

2.2. FORMATTER

The Formatter board (10 FPGAs and some memory chips) was designed to perform processor-like tasks at high speed. The Formatter card interfaces the IDPU with the FAST instruments over twenty separate serial interfaces (see Figure 3). Some of these instrument interfaces provide spin-synchronous data, others provide time-synchronous data, and some provide data only when requested by the Processor. In addition to these data sources, the Formatter produces various averages of these data for lower telemetry rate 'Survey' data sets, bringing the total number of data sources to twenty-nine. Data from each source is formatted into its own 1024-byte frame and transferred into the Memory. The Processor adds CCSDS (Consultative Committee for Space Data Systems) telemetry frame and packet headers (one packet per frame) using the DMA interface. Each data source has its own desti-
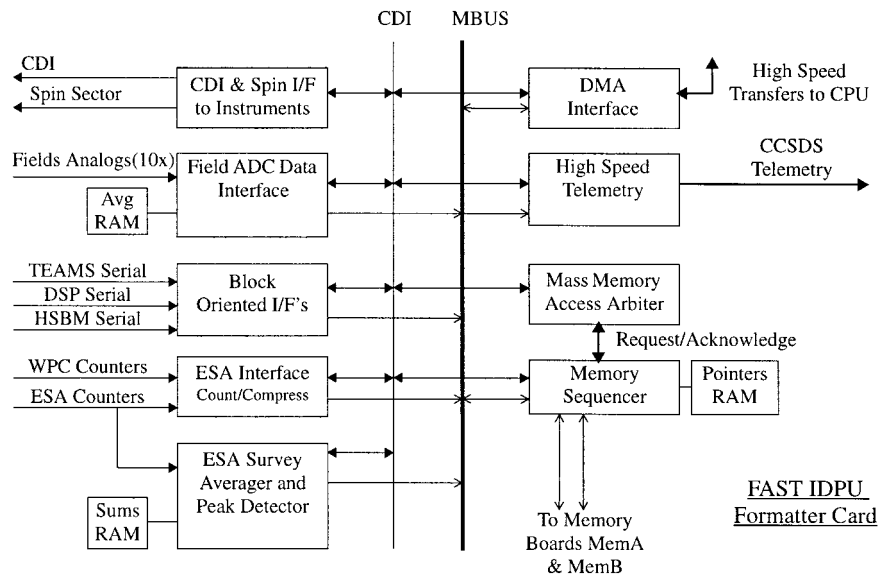
Figure 3. Block diagram of the serial interfaces and functions performed by the FAST formatter board.

nation pointer into memory, and the data is transferred directly to memory as it is collected and formatted. The Processor provides new memory pointers once per frame to each data source, upon request from the Formatter. Access to the memory is arbitrated between the various sources using a custom priority scheme designed to guarantee sufficient memory bandwidth to each source.

The Formatter has two data channels which transfer data from Memory to the high-rate telemetry interface. One is used for CCSDS headers and the other for the corresponding data. The Formatter does very little to most of the collected data before transferring it to memory. Some data from the particle instruments are compressed to 8 bits using a pseudo-square-root compression algorithm (linear for <15 counts, and approximating square-root at larger counts to match the counting statistics). Survey data are generated by the Formatter from the raw data by summing particle data or by averaging fields data. Resolution of the survey data is programmable. In addition, survey data collection involves de-spinning the particle data, since the spacecraft will rotate a significant amount during an accumulation interval. Survey data are collected in parallel with the raw, high-time resolution data to insure complete coverage, since only a fraction of the high-time resolution data are retained for transmission.

The Formatter provides a buffered spacecraft 4 MHz clock (or a lower rate clock divided down from this clock), plus a once-per-second synchronizing pulse to time-synchronous instruments and the Processor. A CDI command is also broadcast to all instruments indicating the Universal Time (UT) at the next once-per-second pulse.
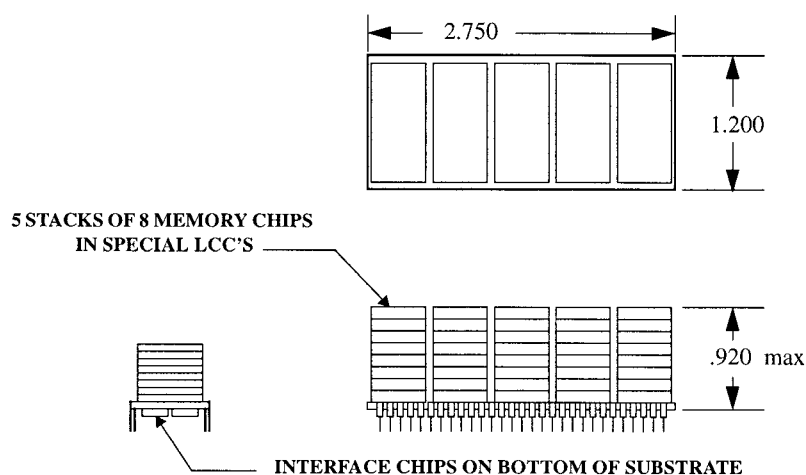
Figure 4. Mechanical drawing of a memory module made by DensePac, Inc., specifically for the FAST mission. Each module is organized as 1 million 40-bit words and weighs about 100 g.

The Formatter provides a sector clock of 12,288 pulses per spin, plus a once-per-spin pulse to spin-synchronous instruments and the Processor. Flight software controls the frequency of this sector clock using a programmable rate generator and phase locks the once-per-spin pulse utilizing spacecraft-provided sun and horizon sensor pulses. The Formatter also calculates a magnetometer spin phase defined as the phase angle between the zero crossing of a magnetometer axis and the sun pulse. Magnetometer spin phase is included in data packet headers to simplify data analysis and is used by particle instruments to control deflectors that track the magnetic field.

2.3. MEMORY

The FAST Mass Memory system provides the IDPU with 125 MBytes of error-corrected high-speed RAM, distributed on two electrically independent units. The memory uses a stacked-chip memory module made for the University of California, Berkeley (UCB) by DensePac, Inc., which contains 40 Hitachi 128kbits × 8 SRAM memory chips arranged in 5 stacks of 8 chips each, along with all necessary decoders and data buffers. Each module is organized as 1 mega-word by 40 bits. Module packages are about $1.25'' \times 2.75'' \times 1.0''$ in size and weigh very close to 100 grams (see Figure 4).

Each $10'' \times 10''$ memory board can accommodate sixteen modules, address decoders and two FPGA chips which provide the interface to the board connector and perform single bit error correction on the stored data. The overall organization of each card is 16 777 216 words of 32 bits, with the 40-bit width of the modules providing the additional storage needed for the Error Detection and Correction (EDAC) logic. Since the modules are relatively heavy, each module is glued into

an aluminum frame that is bolted to a board-mounted, waffle-shaped support frame. The frame prevents flexing of the circuit board during vibration which could easily break pins on the large memory module packages.

Concern that FAST might experience levels of radiation higher than expected and lose its entire memory system to radiation damage prompted the inclusion of a minimum amount of rad-hard memory. The highest addressed DensePac module was omitted from one memory card and a set of sockets installed instead. This allowed a separate $10'' \times 10''$ card to be electrically connected in place of the DensePac module. The card contains 40 IBM 32 kbits $\times$ 8 megarad hard SRAMs, organized as 262 144 words of 40 bits. It has $\frac{1}{4}$ the capacity of the DensePac module. In summary, the FAST memory consists of two memory boards, one of them having the full complement of 16 DensePac modules and the other equipped with 15 DensePac modules and the 'rad-hard IBM board'. This yields a total capacity of 125 MBytes.

Each memory card uses its EDAC circuit to 'scrub' out single-event upsets in the memory by simply reading, correcting and re-writing each memory location periodically. This keeps correctable single-bit errors from becoming multiple-bit (and therefore uncorrectable) errors. Upon command from the Processor, the memory card automatically scrubs a section of its memory specified by a 10-bit block address. The scrubber hardware steps through the identified section of memory, scrubbing each cell and counting both single-bit and multiple-bit errors. When the block is finished, flight processor reads back the number of errors encountered in that memory section and provides these data to housekeeping packets.

A special power control circuit protects the memory modules against latch-up induced over current, and isolates the system against failure of the memory chips on each module. This function, implemented on a FPGA chip, along with analog circuits contained on five small daughter boards mounted on the main memory board, provides a separately controllable circuit breaker on the power line of each module. Each module also contains a set of radiation hard 54AC245 drivers which isolate the system bus from failures in the memory chips.

Distribution of latch-up protected power to the 16 modules is performed with three 16-bit wide control registers called Power, Trip and Override. In normal operation, the Power register is configured to power the memory, turning on a 10 mA 'standby' current source which delivers +5 V to each DensePac memory module. The Trip and Override registers are disabled in normal operation. When a module is 'chip selected', a 300 mA 'active' current source is enabled to supply the extra current needed for operating the memory. If the voltage on any module drops below 4.5 V (in the event of an SEU), the corresponding bit in the Power register is reset and the 10 mA 'standby' current source to the module is turned off.

If the bit in the Trip register is set, the current source is held 'on' even in the event of an under voltage condition. If the bit in the Override register is set, the 300 mA current source is held 'on' continuously. This option would be used in the case of radiation damage to a module which still works, but draws more than

10 mA of standby current. Note that 'chip selecting' a module will always enable the 300 mA source, so to fully shut down a module the software must not address it.

2.4. POWER CONVERTER/SWITCHES

Science requirements on the FAST payload specified that data be collected at a high rate during a relatively small portion of the orbit (during auroral crossings), and at a much lower rate during the remainder of the orbit. The spacecraft required operation of different complements of instruments depending upon which data were to be emphasized. The capability of the SMEX launch vehicle limited the size of the solar array and batteries, and drove the payload power system toward a design with a large dynamic range. The power system design required high efficiency both at high power outputs, when all science instruments are collecting data at the maximum rate, and at low power outputs, during the majority of the orbit when limited data are collected. It also had to meet the usual requirements of isolation between power and signal grounds, and have the ability to provide regulated power at various positive and negative voltages to the science instruments while operating from an unregulated 28 ±7 V bus. In addition, to minimize power consumption while taking data with various combinations of instruments, it was required to be able to separately switch on and off the power to each instrument. Finally, to properly manage the system and minimize overall power, the current from each separately switched service had to be monitored.

Figure 5 shows the block diagram of the system used on the FAST payload to meet the above requirements. Two MUE-controlled 28 Volt services supply power to three separate DC-DC converters containing both isolated and regulated outputs. A third MUE-controlled 28 V service is used for boom deployments. Converter number 1 is always powered and supplies power to the IDPU and the instruments which are usually powered during the low data rate portion of the orbit. (This includes the mass spectrometer high voltage, the fluxgate magnetometer, and part of the electric field instrument.) Converter number 2 provides regulated isolated 28 V to unregulated DC-DC converters in the electrostatic analyzer (ESA) particle detectors and electric field boom units. Converter number 3 provides regulated ±12 V and ±5 V to the parts of the electric field analog signal processing electronics which are usually operated only during periods of rapid data taking.

Following the three DC-DC converters is a power controller switch bank which consists of 48 separately limited and monitored power services controlled by 26 logic signals which separately limit and control power to the various parts of the science instrument. A logic signal switches through four different power configurations depending on the requirements of the science instrument being controlled. Current limiters for the higher power services 'fold back' (exceeding the current limit will cause the current to fold back to a low current limit effectively shutting off power) to limit power dissipation in the pass transistors. The lower power ser-
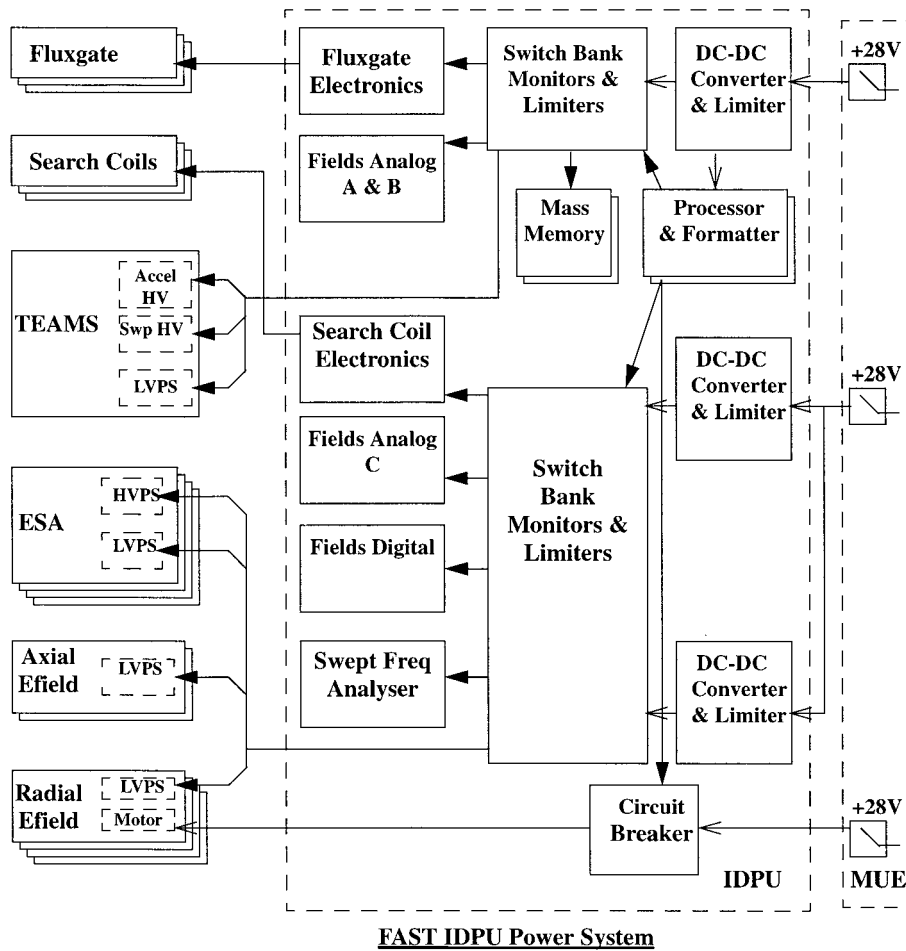
FAST IDPU Power System

Figure 5. Block diagram of the FAST power system.

vices have a constant limiting current. Both current and voltage are monitored on all services and these values are read by the processor and stored for instrument housekeeping.

The switches to the various instrument services are controlled by the Processor which applies power only to those instruments that are programmed to collect data, write data into the mass memory, or put data into the telemetry downlink. Power can be further reduced by using time tagged commands to request the MUE to turn off the input 28 V to converters 2 and 3 when in low power mode. In this way both high power and low power operation can maintain high efficiency.

3. Software Description

The FAST Instrument Data Processing Unit (IDPU) flight software is responsible for instrument power control, time and attitude determination, mass memory control, science instrument control, command distribution, telemetry formatting and boom deployments. Relying upon the MUE for mission critical operations allows the IDPU software to focus the processing power of the 10 MIPS 32-bit CPU primarily on science data collection and related engineering.

The IDPU flight software is organized into 21 modules, totalling just over 20 000 lines of assembly code. It requires a mere 31 KBytes of code space and 60 KBytes of RAM. The software was written in assembly language and represents about one man year of effort. For radiation tolerance and extensibility, a bootstrap version of the flight code is stored in a bi-polar PROM and later versions are kept in an EEPROM. Upon initialization, the bootstrap software copies itself into RAM, turns off the PROM, and checks over the four available EEPROM programs. Unless commanded to stop within ten seconds, it automatically loads and executes the latest flight software version. Thus, the flight code runs entirely in RAM, requires no PROM or EEPROM power, and is directly patchable by ground command.

The flight software can be functionally grouped into these main elements: (1) Operating System, (2) Engineering, (3) Commands, (4) Telemetry Management, and (5) Instrument Control.

3.1. OPERATING SYSTEM

The IDPU includes a very simple operating system for loading and executing new software, handling interrupts, distributing processor time to a variety of tasks, and functionally isolating hardware from software. Included in this functional grouping are three modules which manage the CPU's foreground, background and input/output, respectively.

3.1.1. *Foreground Management*
IDPU foreground management is a simple 'round-robin' polling loop in which each module gets called once per cycle. It is intended for non-time-critical applications and operations that require more CPU time than is allowed in an interrupt service routine. Since all available CPU cycles are expended in this loop, the loop rate is telemetered as a rough gauge of CPU load, and typically runs around 2 KHz.

3.1.2. *Background Management*
The background module is the central clearinghouse for interrupts, containing the logic for controlling CPU interrupts while calling subordinate modules as needed to service each one. Thus, all other modules are isolated from the details of the CPU interrupt hardware, and the background manager is able to level the load. During a typical science data collection, flight software must handle approximately 4000

interrupts per second, divided among five varieties: UART, CLOCK, SECTOR, FORMATTER, and DMA.

A Universal Asynchronous Receiver Transmitter (UART) interrupt occurs whenever a byte is received by or transferred to the MUE over one of the two redundant UART circuits. Under normal conditions, data flows on only one UART and in only one direction at a time, and so the nominal interrupt rate is 960 Hz. The Communication module is called to service these interrupts and determine which or both of the UARTs to enable (see Communications).

The CLOCK interrupt occurs at 256 Hz, driven off the MUE-provided 2^{22} Hz clock line, and in sync with the 1-s ticks. Time keeping, command processing, telemetry buffer management, time-tagging of the sun and horizon sensor inputs, etc., are time-sliced activities performed in this interrupt. This is implemented using a fixed table of vectors so each function gets a known execution rate. For example, commands are called at 128 Hz, the telemetry manager at 64 Hz, and so forth.

The SECTOR interrupt occurs 1024 times per spin, and under nominal 5-s spin conditions, these occur at 1024/5 s or 205 Hz. Under this interrupt, flight software manages the phase-locked-loop control, housekeeping sampling, particle instrument sectoring, and deployment logic.

The FORMATTER interrupt occurs whenever the formatter board completes any of 31 possible data transfers to or from the mass memory system. There are 29 instrument data sources producing 1 KByte packets which the formatter stores into the Mass Memory, and 2 telemetry packets (header and data) which the formatter transfers from Mass Memory to the MUE. On each interrupt, the software determines which memory transfers have completed and calculates the new memory positions for each pointer for the next memory transfers. For each completed data packet, the software formats a CCSDS header block, puts it in a First In First Out (FIFO) memory and sends a request to the DMA module to transfer it to the mass memory. With telemetry on and the IDPU running a typical mode, the formatter card is transferring data at 10 Mbit s^{-1} systemwide resulting in approximately 1600 formatter interrupts per second. To support this rate, flight software is formatting 320 Kbps of header data to tag the hardware-generated telemetry data.

The DMA interrupt occurs whenever packets are transferred between the CPU memory and the Mass Memory. Science frame headers, spacecraft engineering frames, scrubber packets, trigger packets and engineering packets are written to memory while magnetometer data are read out of memory using this facility. The transfer requests are buffered in a software FIFO in order to level the transfer load.

3.2. ENGINEERING

3.2.1. *The Communications Module*

For command and control functions, the IDPU communicates with the Spacecraft Mission Unique Electronics (MUE) using redundant 9600 baud UARTs and a bi-level control line called UARTSEL. After initialization, IDPU flight software receives communicated packets from the MUE from either UART and responds to the MUE using the same UART. Since the MUE automatically switches UARTs in the event of a repeated communication malfunction, this logic keeps the IDPU software in sync automatically. To prevent an errant UART from inventing its own characters and disturbing the communications, the UARTSEL line allows flight operations personnel to force communications to proceed using a selected UART. Software monitors the UARTSEL line and counts transitions to select which UART to use.

The relationship between the MUE and IDPU is strictly master-slave. The MUE initiates all transfers and the IDPU responds, sometimes merely with the equivalent of 'OK'. Each packet from the MUE to the IDPU begins with an opcode and a length byte, and ends with a 20 ms gap. IDPU software checks the validity of the opcode and verifies that the length byte matches the number of transmitted bytes in the packet. In the event of an unknown opcode or mis-match in the length, the IDPU responds with an error code.

Except for ground command packets which occur asynchronously, the communication schedule is defined by tables inside the MUE. Time and backup ACS data are sent to the IDPU once per second, and a status request is performed at the same rate. MUE housekeeping packets are sent to the IDPU at a very low $\frac{1}{8}$ Hz rate. In addition, one of the status bits of the IDPU is a request for more telemetry on a 'space-available' priority.

Across this interface, the MUE sends two types of engineering frames to the IDPU for downlink. Virtual Channel 0 (VC0) frames are real-time engineering frames which the IDPU transmits at high priority while VC1 engineering frames are stored in an IDPU memory segment until commanded to dump.

3.2.2. *Housekeeping Data Collection*

One of the tasks of the flight software is the collection and transmission of housekeeping analog and digital data. Since the A/D converter is power strobed with the 1024 spin sector pulses, the housekeeping module runs at that rate (205 Hz). On each interrupt, the software is allowed to make three A/D samples, which are allocated to 'slow', 'fast' and 'ESA' housekeeping. Slow housekeeping provides a multiplexed slow sampling for all instrument voltages, currents, and temperatures. Fast housekeeping samples a small number of electric field signals or a boom deployment at a higher rate. ESA housekeeping is used to monitor high voltage turn-on. A foreground task monitors housekeeping levels for limit violations and can turn off systems that exceed these limits.

Since the IDPU is slaved to the MUE on the communication port, IDPU software formats CCSDS housekeeping packets only when called to do so by the MUE. This provides the most-recent data to telemetry, but has the disadvantage that the sampling and telemetry are asynchronous.

3.2.3. *Power Management*

In order to conserve power, almost all of the IDPU instrumentation is powered off except while taking data over the north or south auroral zones. The background module keeps instruments informed of which instruments are 'on' by broadcasting the 32-bit power configuration once per second. This information is primarily used by the various electronics boards to disable inter-instrument interfaces in which one of the components is unpowered.

As a safety measure in case the main bus voltage is low, the spacecraft MUE can command the IDPU to go to one of four power levels (0, 1, 2, 3). This is implemented by masking the 32-bit Power Request register with one of four masks, each corresponding to a level. As the selected power level changes, the one-second broadcasts keep instruments aware of the power configuration. At reset, the IDPU is required to be at power level 0, which allows only the CPU and Memory to be powered. Nominal operation leaves the IDPU at power level 3. Intermediate power levels are implemented by the MUE when the battery drops to low charge states.

3.2.4. *Time Management*

The background module uses the CLOCK interrupt to maintain Universal Time (UT) and to ensure that all data generator circuits have the correct time. The MUE regularly sends its Mission Elapsed Time register and a UT offset value to the IDPU, which converts these into a UT clock, formatted as 4 bytes of seconds and 2 bytes of sub-seconds. IDPU software adds 1 second to determine the time at the next tick mark, saves this in a temporary register, and broadcasts this time via the CDI bus to all data generator circuits. When the 1-second tick mark is observed, the temporary register is latched into the IDPU Time register by the interrupt software, while all data generators latch their time registers. The module then uses the 256 Hz interrupt to increment the Time register for use by the rest of the flight software.

3.2.5. *Attitude Determination and Spin Control*

One of the IDPU software functions is to electronically sector or 'point' instrument data in a fixed, sun-referenced coordinate system. If the vehicle is in sunlight, this is pretty straightforward through the use of a sun-sensor and phase-locked loop circuit. However, when the spacecraft enters the shadow, it spins up due to thermal contraction of the spacecraft booms, so maintaining a sun-referenced system throughout the shadow is non-trivial. Since FAST also contains horizon sensors, knowledge of the Earth's horizons can be combined with orbit knowledge to determine the sun-referenced spin phase. Determination of the sun-referenced

spin phase is divided into three parts: object timing, spin phase control, and shadow operations.

3.2.5.1. *Object Timing.*   Objects are any physical measurement that can be used to determine spacecraft spin phase. Object timing is performed by a combination of Formatter board circuits and software which puts UT time-tags on MUE-supplied Sun and Horizon-crossing signals. Redundant time-tagged versions of these are also communicated via the IDPU-MUE communications link. Both hardware- and MUE-reported Horizon crossing signals are used to calculate the Nadir spin phase. This calculation is slightly complicated by the fact that the Sun and the Moon also trigger the Horizon sensor, but these objects are easily identified by their relatively small angular widths. Sunrise and sunset cause a unique signature in the horizon pulses that is easily handled by the Nadir calculation. IDPU software constantly maintains and telemeters the sun crossing times and earth nadir crossing times from both primary and backup sensors. Magnetic field sensor zero crossings are used to calculate an additional object time.

One of the above object times is selected for spin phase determination and flight software maintains a virtual object which is a copy of the selected object's time plus an optional offset. A bit in the object selection register enables a time-offset register to be applied to the object time, so that sectoring will point ahead or behind a true object. This feature is used for correcting the differences between the response of sensors and to allow Sun pointing to be calculated from Nadir pointing, as described later. Finally, software calculates the object's period using an infinite response digital filter, rejecting invalid points prior to filtering. Points are deemed invalid when too large a phase shift is observed.

3.2.5.2. *Spin Phase Control.*   Spin phase control is maintained by a combination of hardware and software. Using a programmable divider of the 2^{22} Hz clock, IDPU hardware broadcasts a 'spin-sync' pulse and 12×1024 'sector' pulses per spin to all instruments. Phase synchronization between 'spin-sync' and the selected 'object' is maintained by a software 'phase-locked loop' that adjusts the clock divider value. The allowable range in hardware is 512 to 4096 which corresponds to spin periods of 1.5 to 12 s.

Since a nominal spin period of 5 s results in a divider value of 1280, maintaining a fixed value over a spin would yield only a modest 0.3 deg precision. Thus, flight software calculates an interpolation point within the 1024 sector spin in which to switch this divider from 'N' to '$N + 1$', giving a much more precise solution. For example, if the spin period is 5.000015 s, the divider should be 1280.0039. This is approximated by $0.0039 \times 1024 = 4$ sectors at 1281 and 1020 sectors at 1280.

Software maintains phase synchronization by comparing the times of the local spin-sync pulse and the selected object. When this error is above 1 degree, it calculates the divider value that will achieve an immediate match between spin-sync and
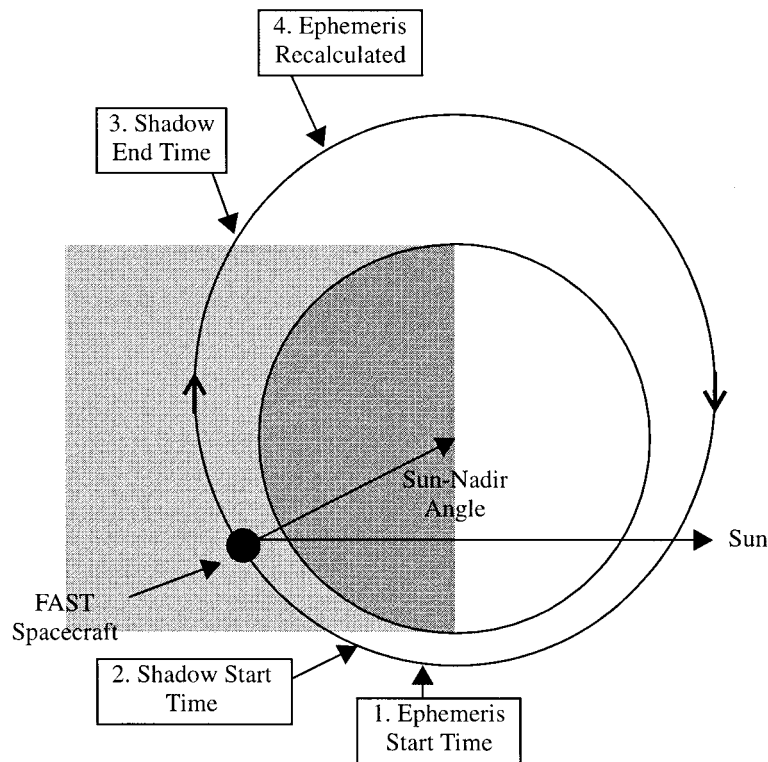
Sun-Nadir Operation Points of Interest

Figure 6. Drawing showing a sequence of time-tagged events that are used to turn on and off the Sun-Nadir routine. The Sun-Nadir routine allows the spacecraft to calculate the sun direction while in shadow using knowledge of the nadir direction provided by horizon sensors. This routine provides for smooth transitions in and out of eclipse for data organized by spin phase.

the object using the known period. For small error-angles, software uses a filtered phase error to determine the divider value and when to flip from 'N' to '$N + 1$'.

3.2.5.3. *Shadow Operations.* Shadow operations are implemented to provide smooth transitions in and out of eclipse. The role of the 'shadow' software is to keep the sectoring pointed at the sun, even during shadows. To do this, the Science Operations Center or SOC (see McFadden *et al.*, 2001) calculates a Sun-Nadir ephemeris file which is loaded into the IDPU once every few days. Included in this file are the orbit period, the UT to start using the new file, the shadow start and end times, and two tables. The first table is a piece-wise linear approximation of the Sun-Nadir angles during the first orbit, and the second is the differential orbit-to-orbit change in Sun-Nadir angles. The two tables can be combined to generate Sun-Nadir angles that are accurate enough to operate for several days.

As shown in Figure 6, the shadow sequence begins at an Ephemeris Start Time. Using the piecewise linear angles array, software begins calculating the offset between the earth and the sun once per second. The start time is in sunlight to allow comparisons between the real and the calculated Sun-Nadir offset. The pointing object is still the sun, and the sun-nadir offset is only telemetered. At the Shadow Start Time defined in the ephemeris, software switches the selected object to 'Nadir-plus-offset', and calculates the difference between the true and calculated sun angle to generate the 'onset error'. The 'onset error' is used to force agreement between the ephemeris-determined sun crossing time and actual Sun crossing time. Thereafter, software simply calculates the Sun-Nadir angle, subtracts the onset error, and sets the object offset. (Note: This has been implemented so that UT does not need to be monotonic. Spacecraft UT can be adjusted without problem.) Finally, at the Shadow End Time, software switches the object back to Sun, and recomputes the ephemeris for the next orbit. The Ephemeris Start Time is updated by adding the orbital period, and the delta-angle table is simply added to the angle table. The ephemeris gradually gets worse at predicting the position of the Sun, and computer models show a maximum usable duration of about a week. On-orbit performance has been superb, routinely demonstrating better than 1 deg pointing accuracy in shadow.

3.2.6. *Deployments*

Flight software responsibilities include deploying the four spin-plane wire boom systems, and verifying the deployments of the axial and magnetometer booms. Each boom unit is equipped with a turns-counter microswitch which is sampled by the IDPU software and converted to a length accurate to about 10 cm. Software is normally commanded to deploy a pair of booms on opposite sides of the vehicle to a length defined in 'turns' or 'clicks' on a microswitch closure caused by a drive-wheel cam.

Since booms deploy at slightly different rates, software monitors the lengths and if one boom gets more than 2 'clicks' ahead of the opposite boom, the longer boom is paused until the shorter boom catches up. If either boom indicates a 'jam' condition, the deployment is halted within 0.1 s.

A unique feature of the FAST deployment software is that it synchronizes deployment with the oscillation period of the deployed booms in order to cancel boom motion. Decisions to start and stop motors are made in phase with the predicted oscillation, so the booms are not swinging at the end of each deployment. On-orbit performance of this logic has been verified.

3.2.7. *Mass Memory Control*

3.2.7.1. *Memory Integrity.*   In order to maintain memory integrity in the presence of penetrating radiation, the memory has an Error Detection And Correction circuit (EDAC). This circuit is capable of correcting all single-bit errors and some multi-bit errors during the read cycle. Using the EDAC, a 'scrubber' circuit simply reads
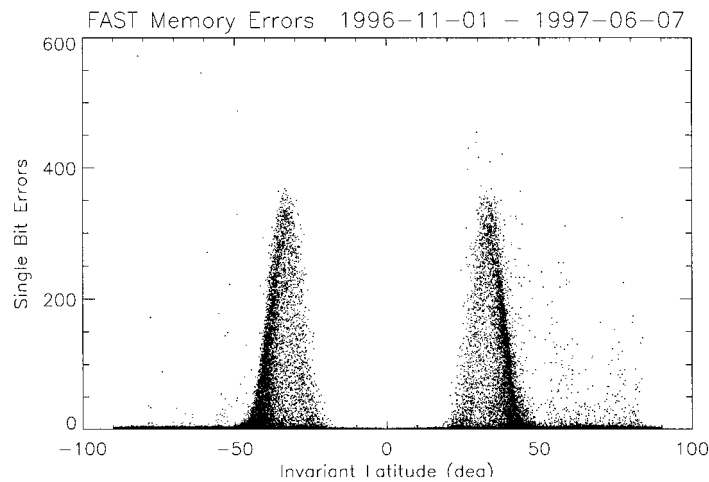
Figure 7. The graph shows the invariant latitude location of single bit errors in the mass memory. Most errors are associated with penetrating radiation as the spacecraft passes through the inner radiation belt. Single bit errors are corrected by Error Detection and Correction (EDAC) hardware and scrubber software which periodically pages through the memory.

and re-writes memory locations correcting single-bit errors before they become uncorrectable double bit errors.

The flight software initiates scrubs on each of the 1024 × 128KB pages using an 8-bit timebase, whose default is 4 pages per second. On each scrub cycle, the software reads the number of multiple and single-bit errors found in the page, and generates 'scrubber' telemetry packets. The software keeps the total number of errors for the entire valid memory, as well as identifying the worst memory segment with its error count. These data are available in the housekeeping telemetry.

On-orbit performance of the scrubber circuit has demonstrated that the corrections are usually single-bit, correctable, and uniformly distributed throughout the memory. In a seven month study (1447 orbits), the errors were found to occur primarily between ± 50 deg of invariant latitude, within the radiation belts, as seen in Table I and Figure 7.

3.2.7.2. *Memory Power.*   As described in the memory hardware section, the memory subsystem is powered in 32 sub-sections, each with its own current trip circuit. This strategy guards against the possibility of a single latchup disabling the entire memory. The software monitors the status of these 32 trip circuits and includes it in the memory scrubber packet telemetry. Only one memory trip was observed during the first two years on orbit. This module was reset and continued to function properly.

3.2.7.3. *Virtual Memory.*   Concern that parts of the Mass Memory might fail either during launch or due to radiation prompted IDPU software to provide a

TABLE I

During a seven month study, memory errors occured primarily within the radiation belts

Item	Single-bit errors	Multiple-bit errors
Total error count	918383	2915
Errors per orbit (7990 s)	630 ($\pm$320)	2 ($\pm$1.8)
Errors | ILAT | < 50 deg	96.8%	93.7%
Errors | LAT | < 50 deg	97.2%	94.9%

virtual-to-physical mapping of the memory. The software remaps the upper 10-bits of the address in order to subdivide the memory into 1024 pages of 128 KBytes each. Using a bitmap loaded by ground command, the memory controller software recalculates the virtual-to-physical mapping automatically, and adjusts the amount of telemetry allocated to each Virtual Channel. Thus, bad sections of memory can be removed from operation, while the only observable effect is that the Mass Memory shrinks! After 2 years of operation, only one page of memory has been removed because it contained unusually high bit errors.

3.2.7.4. *Memory Configurations.*   The Mass Memory is used exclusively for CCSDS frames, each being 1064 bytes in length. In order to make the hardware processing of science data easier, the 40-byte header and 1024-byte data sections are stored in separate areas of the Mass Memory. All science data is provided by the Formatter directly into the 1024-byte sections, while the IDPU formats and writes all headers to the Header Segment.

   All 40-byte headers are packed together into the Header Segment, abiding by the hardware restriction that no header cross a 1 K boundary. Thus, only 25 headers are stored per KByte. The size of the Header Segment is determined by how much usable Mass Memory is available. The software calculates this and uses only the amount of memory necessary. If all 125 MBytes of memory are available, the headers require 5.17 MBytes for the 126144 packets.

3.3. COMMAND MANAGEMENT

3.3.1. *Command Strings*
The command module executes strings of 32-bit commands, either sent from the ground through the spacecraft MUE or commanded internally by other IDPU software modules. If no spacecraft commands are present, flight software executes command strings from internal sources. New instrument operating modes, Fast and Slow Survey data rate transitions, and instrument initialization sequences are examples of internal strings of commands.

While commands are nominally executed at a rate of 128 Hz, a delay command is provided for slowing down command sequences as required in some cases. The module also provides a counter function in order to verify reception of long ground-commanded sequences.

3.3.2. *Memory and Table Operations*

The command module includes a very basic set of memory commands for loading, dumping and executing. In addition, there are 18 tables which allow memory operations to use relative addressing. This facility allows loads to be somewhat independent of software version.

A special feature of memory loads involves the EEPROM. Since writing to the EEPROM is limited to 1000 writes, software checks each word to see if the EEPROM already has the value and writes data to it only as necessary. In addition, performing EEPROM writes requires an enable command.

3.3.3. *Modes*

The command module is responsible for loading, checking and executing special sets of command strings known as 'modes'. There are two basic varieties of modes, Particle modes and Field modes. The IDPU stores up to eight of each type, with each mode containing a maximum of 512 commands. Modes are identified using an 8-bit code and periodically verified using an integral checksum.

Modes are executed using a 'start mode' command which requests one Particle and one Field mode identifier. In response, the command module simply looks up these command strings and uses its internal command channel to execute the strings.
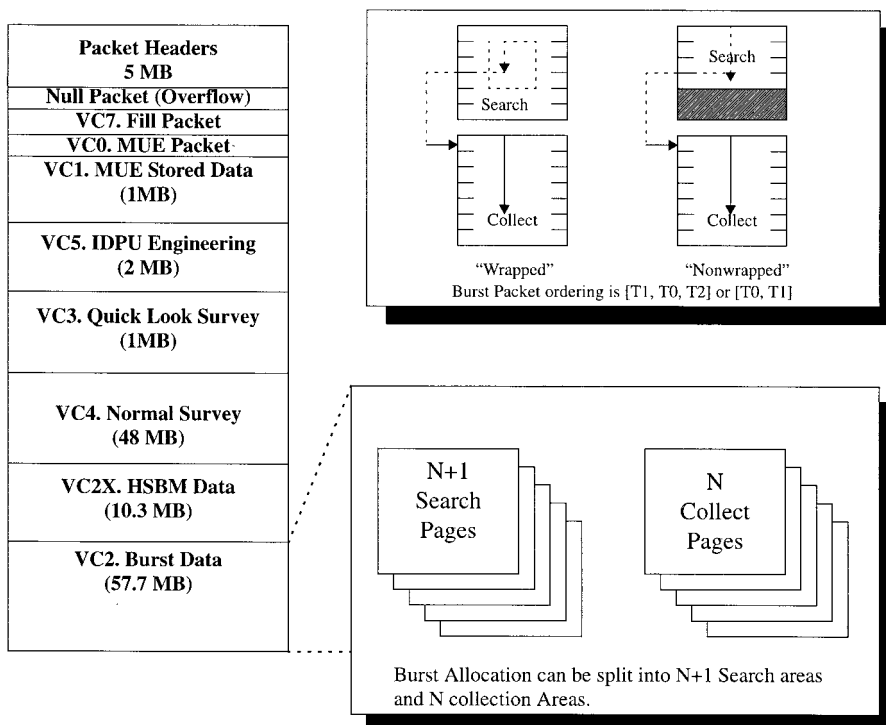
It is worth noting that a variety of IDPU operations, particularly instrument driver commands, are implemented by spawning command strings for instrument configuration. When executed from within a mode, these spawned command strings must wait for the mode string to complete in order to execute.

Contained in each mode definition are two command sequences, one for Slow-to-Fast and one for Fast-to-Slow survey data rate transitions. These sequences contain a maximum of 256 commands and define for the IDPU what it should do when switching between different survey data collection speeds. Thus, a given Fast-to-Slow or Slow-to-Fast transition can have up to 512 total commands to tell the science instruments how to change the data rate.

3.4. TELEMETRY MANAGEMENT

3.4.1. *Virtual Channel Allocations*

In order to handle the various data types (housekeeping, survey data, burst data, etc.) with different readout priorities, the Mass Memory is partitioned into several blocks. Flight software subdivides the Mass Memory into eight Virtual Channel storage areas, an area for packet headers, a VC7 (fill) packet, and a spare packet

| Packet Headers 5 MB |
| Null Packet (Overflow) |
| VC7. Fill Packet |
| VC0. MUE Packet |
| VC1. MUE Stored Data (1MB) |
| VC5. IDPU Engineering (2 MB) |
| VC3. Quick Look Survey (1MB) |
| VC4. Normal Survey (48 MB) |
| VC2X. HSBM Data (10.3 MB) |
| VC2. Burst Data (57.7 MB) |

FAST IDPU Mass Memory Allocations

Figure 8. Diagram of the FAST mass memory allocation into Virtual Channels (VCs). Most memory segments or VCs are operated as FIFOs. VC2 burst data differs as shown with this area of memory further subdivided into N collect and $N + 1$ search pages. One collect and one search page combine to form a single burst collection, with readout priority given to the burst with the highest quality. The extra search page is used to temporarily store data until a trigger algorithm selects the new data and a collect page of the least interesting burst is overwritten.

to handle full-memory cases. Figure 8 illustrates these storage areas and Table II shows typical size and allocation priorities. During a memory reconfiguration, memory is allocated on a table-driven priority basis, with 0 the highest priority. Lower priority virtual channels may not get their full allocation if the sum of the requested memory is less than the available memory, which may occur if memory modules are disabled.

3.4.2. *MUE Engineering Telemetry Management (VC1)*

The VC1 segment is used to store MUE Engineering telemetry frames sent to the IDPU via the IDPU-MUE interface. These MUE generated packets contain housekeeping data on various spacecraft systems (command status, battery state of charge, temperature monitors, voltage/ current monitors, attitude control inputs, etc.). MUE telemetry frames are stored in a circular fashion, with the newest data

TABLE II

Typical mass memory allocation among various data products including memory allocation priority with 0 the highest priority

Segment	Allocation	Priority	Description
HEADERS	5.2 MB	0	Header area for all frames
NULL	0.001 MB	0	Location to direct overflow packets
VC7	0.001 MB	1	CCSDS fill data frame
VC0	0.001 MB	2	MUE real-time engineering frame
VC1	1.0 MB	3	MUE stored engineering data
VC5	2.0 MB	4	IDPU engineering data
VC3	1.0 MB	5	Survey data (quick-look)
VC4	48.0 MB	6	Survey data
VC2X	10.3 MB	7	High speed burst memory data
VC2	57.5 MB	8	Burst data
Total	125.003 MB		

overwriting the oldest. A playback command dumps the contents of the buffer, but does not automatically clear it. Thus, the flight controllers can repeat the playback without losing any data. A separate 'clear' command resets the buffer.

3.4.3. *IDPU Engineering Telemetry Management (VC5)*
Similar to the VC1 segment, the VC5 segment is a circular buffer used to store IDPU Engineering frames generated by IDPU flight software. These blocks are stored in the circular FIFO until the transmitter is available. In the event of overflow, the IDPU saves the most recent engineering data by overwriting the oldest data.

3.4.4. *Survey and Quick-Look Telemetry Management (VC4 and VC3)*
In order to provide a large-scale context for the high-rate burst data snapshots (see Section 3.4.5), a 'survey data' collection scheme was implemented. Survey data are generated by all the instruments during science data collection, either by averaging the data or by filtering the output of analog signals before digitization. Two different survey data collection rates, termed fast survey and slow survey, are used to optimize the science. Fast survey data collection rates ($\sim$0.5 Mbit s^{-1}) are about an order of magnitude higher than slow survey rates ($<$0.05 Mbit s^{-1}) and about an order of magnitude lower than high-rate burst data collection ($\sim$5–12 Mbit s^{-1}, see Section 3.4.5.). Onboard data evaluation schemes determine which survey collection rate to use and a memory management scheme prevents survey memory overflow (described below). In addition, survey data can be stored in two

separate memory buffers (VC3 and VC4) which have different readout priorities and are used to enhance real-time science operations.

The VC3 and VC4 segments are independent circular FIFO buffers providing comprehensive storage of survey science data. While VC4 provides the bulk of the storage, VC3 is intended for quick-look datasets during telemetry contacts. By simply directing a subset of the survey data to an empty VC3 buffer, users can avoid waiting for VC4 to empty before getting real-time survey data.

In addition to storing and playing survey data, the VC4 manager is responsible for switching the IDPU between Fast and Slow survey modes. The concept is simple: whenever the trigger value 'Flevel' exceeds the commanded threshold, a transition to Fast survey data collection mode is initiated. To prevent memory overflow and trigger oscillations, a memory management scheme was implemented as outlined below.

First, in order to prevent the system from simply filling up all of the survey buffers with a few minutes of Fast survey data, leaving no room to complete the orbit, the VC4 manager limits Fast survey data collection to a commandable 'FSALLOC' page limit. The page limit is set by stored command after each data dump, with page limit allocations calculated by the science operations center based upon contact schedules and data collection periods (see McFadden *et al.*, 2001). The Fast survey page limit is implemented by maintaining a bitmap of the VC4 memory, indicating which pages are Fast Survey data and which are Slow Survey data, and keeping track of how much Fast survey is in memory at a given time as data is collected and telemetered.

Second, to prevent system oscillation between Fast and Slow survey data collection during contacts where the data is being simultaneously collected and transmitted, software limits the transition rate. These oscillations would happen when the Fast survey data collection rate exceeds the telemetry rate, and the Slow survey collection rate is below the telemetry rate. For example, the system would go into Fast mode, filling the VC4 memory until it hits the FSALLOC limit, switch to Slow mode allowing the telemetry to catch up, then quickly switch back to Fast mode. To limit this switching, the software allows Slow-to-Fast transitions only if there is $\frac{1}{8}$th of the FSALLOC memory available for Fast survey data.

Third, in order to prevent oscillations between Fast and Slow modes caused by fluctuations in the Flevel trigger calculations, the software only allows transitions to Slow mode when the Flevel falls below the threshold for 20 s. Hysteresis in this decision could have been employed, except the performance would have depended on filtering characteristics, which was judged more difficult than simple timing.

Once the software has decided to switch between Fast and Slow modes, the actual implementation is quite easy. The Slow-to-Fast and Fast-to-Slow transitions are simply command strings set in memory and maintained by 'mode' definitions (see Section 3.3.3), so the VC4 manager merely instructs the command module to execute one of the strings.

3.4.5. *Burst Telemetry Management (VC2)*

The most complex of the storage areas is VC2 memory. It can operate as either a circular FIFO like VC3 and VC4, or as a collection of smaller 'burst' memories. When operated in 'burst' mode, the memory is subdivided into $N + 1$ Search areas and N Collect areas, where N is up to 64. Search areas are used to store data before a triggered event and Collect areas contain data sampled after the event. For each burst collection, there is a 16-bit 'science merit' evaluation of the data within that collection called EVALMAX. This value is calculated by a weighted average of the Burst 'Goodness' or 'Bgood' value during the Search and Collect periods. 'Bgood' is evaluated from trigger inputs, with an algorithm similar to the trigger algorithm as described below in Section 3.4.6.

To start a burst collection, the flight software finds the open Search area and directs all VC2 data sampling into that area of memory. At 10 Hz, the Burst trigger level or 'Blevel' parameter is calculated and compared to the minimum of the EVALMAX values for all other stored bursts which are not currently being transmitted. If the 'Blevel' seen during the Search period exceeds the EVALMAX value of another burst, then that burst is selected to be overwritten. Flight software directs all new VC2 data into the overwritten burst's Collect area of memory. When the Collection area is filled, the EVALMAX parameter is calculated and the process begins again.

Flight software selects bursts to be played out on a 'best-first' policy, and, for simplification, does not retract a decision once made. Thus, it is possible that N great events show up during the playing of a less interesting event, and that only $N - 1$ can be stored. Given a large enough 'N', however, this should be very unlikely. FAST typically operates with $N = 5$ which gives bursts with about 10 s durations.

3.4.6. *Trigger Calculations*

Among other things, the IDPU software has to distinguish interesting data from uninteresting data. This is accomplished by applying a pair of user-selected algorithms, called BSALG and FSALG, which control Burst collections and Fast Survey mode, respectively. The user selects which two functions to use, consistent with the planned mode of the instrument, from a library of available functions. These include eight electric field inputs; four electric field functions that combine the electric field inputs to trigger on AKR, Electrostatic Shocks, VLF, etc.; and three ESA functions to trigger on electron and ion counting rates or on a function that tracks the energy with the highest counting rate.

Software executes BSALG and FSALG every 64th of a spin (about 13 times per second), and filters the outputs using three slope-sensitive 16-bit infinite response digital filters, producing the quantities BGood, BLevel, and FLevel. Each filter responds to rising signals with different parameters from falling signals, allowing users to make filters that have both good sensitivity and long retention. BGood is a longer averaged version of BLevel, and the two are used by the Burst man-

ager in the determination of burst collection quality (see Section 3.4.5). Similarly, Flevel is used by the Survey manager to determine Fast survey data collection (see Section 3.4.4).

In order to avoid false triggers during mode transitions and power switching, the module includes an inhibit timer which is usually set by the start of a mode. While active, this clears Flevel and Blevel so that no Bursts or Fast survey modes are activated. In addition, trigger calculations are disabled during both electric field diagnostic sweeps and during trigger algorithm changes.

The trigger packet provides engineering diagnostic data needed to identify 'interesting' regions. Basically all of the available raw trigger data is provided along with the filtered results of two selected functions. Thus, even if the selected functions do not trigger on interesting regions, the raw data can be used to determine which functions would have triggered.

3.4.7. *Telemetry Playback*

Flight software handles up to 530 interrupts per second in processing telemetry frames, feeding the formatter card with the addresses of frames and headers to be telemetered. Each of the VC managers passes segment addresses and lengths (each segment is 16 to 64 telemetry frames) to the formatter control module for transmission. The formatter module saves the segment addresses in a 2-stage list for each VC, providing quick access to addresses yet allowing VC managers to operate at a relaxed pace.

In order to deal with ground segment limitations, the IDPU has to limit data rates by VC (see Table III). For example, VC4 data is limited to 1 Mbit s^{-1}, VC3 to 50 kbps, VC6 to 100 kbps. Software implements frame-rate limiting using 3 rate tables, one for each telemetry rate. In each table are the allowable rates for each of VC1 through VC2X. (Note: VC2X is treated as a separate virtual channel by the IDPU onboard, though its CCSDS header identifies it as a VC2 during transmission.) As telemetry frames are processed, software keeps 6 running rate-adders whose overflows enable respective VC channels for transmission. Once a VC frame is transmitted, another of the same type cannot be transmitted until its rate-adder overflows again. Naturally, spacecraft real-time engineering data (VC0) is not rate limited.

Of the VC types enabled for transmission, the IDPU selects the next VC by simple ordering: i.e VC0, VC1... VC2X. If none of these frames is available to transmit, it will try VC2 (Burst data) and, if nothing else, use the 'Fill' packet VC7. The net effect of all of this is that priority is given to the low volume engineering data (VC0, VC1 and VC5), then to real-time science data (VC3 and VC4), and then to Burst (VC2) and HSBM (VC2X) data.

Finally, in order to allow the ground station to acquire the carrier and determine good quality before dumping science data, each of the VC's can be enabled or disabled for transmission. Software automatically disables all science VC's whenever

TABLE III

Telemetry rate limits for FAST data channels

Telem. priority	Virtual channel	Data bandwidth at 900 KHz	Data bandwidth at 1500 KHz	Data bandwidth at 2250 KHz
1	VC0	Unlimited	Unlimited	Unlimited
2	VC1	49.2	500	563
3	VC5	Unlimited	Unlimited	Unlimited
4	VC3	49.2	49.2	49.2
5	VC4	900	1000	1000
6	VC2	Unlimited	Unlimited	Unlimited
7	VC2X	98.4	100	200

the science telemetry port is cycled, and the ground station enables science VC's only after a good link is established.

3.5. INSTRUMENT CONTROL

The FAST science instrument complement is complex, having hundreds of operating modes, requiring active control of data collection at >10 Mbit s^{-1}, and generating telemetry at several Mbit s^{-1}. In order to deal with this complexity, the flight software includes several methods of instrument control: (1) direct commands using the Command and Data Interface (CDI); (2) Mode strings which organize instrument configurations; and (3) the instrument drivers for dynamic control needs. While the first two have been discussed earlier, the following section sketches three software drivers for handling Fields, ESA and TEAMS control functions.

3.5.1. *Field drivers*

The electric and magnetic field experiments include a number of signal processing and data capture systems that must be actively managed by the Field Drivers (see Ergun *et al.*, 2001, for a description of the experiments). These drivers control data collection by the digital signal processor (DSP) and high speed burst memory (HSBM), monitor the magnetic field, determine the optimal tracking frequency for the swept frequency analyzer (SFA), and control electric field probe voltage and current biases. These drivers play a crucial role in assuring that the electric and magnetic field instruments are operated in proper configurations and that data collection is coordinated to maximize the science return.

3.5.1.1. *Digital Signal Processor.* The Digital Signal Processor (see Ergun *et al.*, 2001) is a 32 MHz AT&T DSP32C, performing Discrete Fourier Transforms (and

more) on digital data from the Field analog boards or the High Speed Burst Memory. The DSP driver in the IDPU is very simple, controlling the DSP-to-Mass Memory packet transfers and formatting frame headers with mode information. Subsecond timing data from the DSP is transferred to the processor, through the formatter interface, at the end of each data packet and is included by the driver in the packet header.

3.5.1.2. *High Speed Burst Memory.*   The HSBM system (see Ergun *et al.*, 2001) digitizes four analog signals at up to 2 MHz and stores these data in one of several 2.5 Mbyte buffers on the HSBM board. Using internal triggering logic, with 10 available signals to choose from, the HSBM autonomously determines and transmits the 'best' data encountered while its acquire signal (ACQ) is active. The functions of the Fields HSBM driver are to raise the ACQ signal whenever Fast Survey or Burst collections are occurring, and to control the HSBM-to-Mass-Memory packet transfers. Since the processor does not have timing information about the HSBM data, the HSBM internally latches the start time of each buffer and sends this data to the processor, through the formatter interface, for inclusion in the packet headers. Since a buffer requires a large number of frames, only the last frame of an HSBM transfer is marked with this latched time. Finally, the driver enables HSBM transfers only if there is sufficient memory storage in VC2X, so that only whole buffers are sent in telemetry.

3.5.1.3. *Magnetic Field Strength Calculations.*   In addition to collecting data packets and formatting headers for the Fluxgate Magnetometer, flight software samples the magnetometer data directly by performing DMA transfers from the Mass Memory to the IDPU local memory, and extracting B1, B2 and B3 from the frame. Each 16-bit magnetometer value has its DC offset removed, is scaled once to reverse the effects of the hardware digital filtering in use, and is scaled a second time to convert to 2.5 nT resolution. From these values, software calculates the unfiltered magnitude from the sum of the squares, as well as a filtered version using a 3-to-1 infinite response filter. The unfiltered and filtered magnitudes are then used by the ESA and SFA drivers, respectively.

3.5.1.4. *Swept Frequency Analyzer (SFA) Tracking.*   In order to allow high frequency resolution measurements, such as auroral kilometric radiation (AKR) near the electron cyclotron frequency, a frequency tracking driver was developed. When enabled by ground command, the SFA driver software calculates and sets the ideal tracking frequency for the SFA board using the computed magnitude of the magnetic field. The frequency is calculated as $TRK = (0.026817 \times MAGnT + 10255.3)$ Hz, where MAGnT is the magnitude of the magnetic field in nanoTesla.

3.5.1.5. *Configurations for Northern and Southern Hemisphere.*   Some of the instruments and trigger algorithms require knowledge of the nadir direction relative

to the magnetic field, which changes with Earth hemispheres. The fields driver software provides command strings which configure the instruments for Northern and Southern hemisphere data collections. Ground system provided time-tagged commands keep the driver aware of the location of the spacecraft relative to the ecliptic plane, so that when modes are started, they need only request the driver to configure instruments for the appropriate hemisphere.

3.5.1.6. *Sensor Bias Tables.*   In order to organize the voltage and current biasing of the ten electric field probes through varying plasma densities, flight software manages four 'bias tables'. Each table is implemented as a command string of up to 31 commands which are loaded into a fixed area of local memory by special field modes. Typically, each bias table holds ten biasing commands, one for each probe, and each command contains two bias values, one for voltage mode and one for density mode. In executing each bias command, software simply checks the voltage/density status of each probe in order to command the appropriate bias value.

3.5.1.7. *Sensor Bias Sweeps.*   Flight software includes a bias sweep module for in-flight determination of sphere photo-emission current and other characteristics. The number of steps in a bias sweep is programmable. For operational simplicity, sweeps can be initiated by direct commands (from modes for example) or run automatically on a timebase. Software separates these two types such that 'automatic' and 'commanded' sweeps occur without interfering with each other. For example, a direct command can sweep a different pair from those set for automatic mode.

Automatic sweep timing is set using a 4-bit parameter 'ASWP' which requests sweeps at $2^{\text{ASWP}} \times \text{spinperiod}/32$. For example, an ASWP of 5 requests a sweep every spin. Setting ASWP to 15 disables automatic sweeps. If one sweep is still in progress when another is requested, the new sweep is delayed until the prior one ends. This facilitates continuous sweeping profiles.

In performing a sweep, software first marks the 'Sweep-in-Progress' bit in order to inhibit the trigger module and then records the bias settings of the probes. If requested, software waits for a given magnetic angle and/or one-second boundary to begin sweeping probes. Once these requirements are met, software simply steps a selected sphere (or pair of spheres) from one bias point to another (and optionally back again). When biasing a pair of spheres, the software will optionally sweep them together or alternate a full sweep on sphere A followed by B. Once the bias stepping is finished, software places the original bias values onto the spheres and removes the triggering restriction. The data is collected and telemetered by normal VC4 (survey) real-time sampling.

Bias steps are implemented under interrupt at 128 Hz using unused command channel capacity. The step rate can be slowed in factors of two from 128 Hz to 1/256vHz, or can be synchronized to 1 point per spin.

3.5.1.8. *Automatic Re-configuration for Sunlight and Shadow.* Early in-flight performance of FAST showed that during shadow periods, the electric field probes in the density mode caused the spacecraft potential to become negative, attracting ions into the ESAs. Due to orbit mechanics and a limited stored command volume, it was deemed impractical to use stored commands to switch probe modes to match shadow periods. Instead, this function was added to the flight software.

The Sun–Shadow manager operates at a 1 Hz rate, determining whether or not the vehicle is in sun or shadow by watching sun pulse times and defining a shadow as twelve seconds without a sun pulse. Whenever a transition in or out of the shadow is detected, flight software creates a command set and submits it to the command string processor. Following that, the module rebiases the probes by requesting the appropriate 'sun' or 'shadow' bias table.

3.5.2. *The ESA Driver*

The electrostatic analyzer (ESA) driver actively manages the 16 electrostatic analyzers on the FAST payload. This driver controls low and high voltage turnon and turnoff; performs pulse height distribution tests of the microchannel-plate detectors and noise and electronic tests of the preamplifier-counter circuits; initiates high rate ESA engineering data collection; determines the tracking energies for the 12 stepped electrostatic analyzers (sESAs); calculates the particle energy flux and peak energy for trigger algorithms; controls the ion and electron ESA deflector settings which provide continuous monitoring of the field aligned fluxes; and actively monitors the health of the ESAs. The ESAs are typically operated with two pairs of ESAs sweeping over the entire energy range (3–30 000 eV) and measuring ions (iESA) and electrons (eESA) with 78 ms resolution. The 12 remaining ESAs form the stepped ESA (sESA) data product. The sESA performs high time resolution (1.6 ms) measurements of electrons over a selectable energy range. The ESA driver performs instrument commanding through a small set of transitions sequences (or command sets) outlined below. The driver also programs hardware on the formatter board to perform many of the sESA and trigger calculations.

The ESA driver software has a small number of operational modes as shown in Table IV. Transitions between modes are made by Transition Sequences, started by ground or stored commands. These Transition Sequences consist of lists of operations to be performed to change the driver to a selected mode. Before starting a transition sequence, the software first checks to see that the sequence is legal. For example, the 'Turn High Voltage On' sequence cannot be run unless the high voltage enable key has been set. The sequence execution rate is limited by CDI command bandwidth considerations, and may also be intentionally slowed down with embedded pauses for other timing considerations. Some Transition Sequences are used for instrument tests and calibrations, and return to the original mode at the end of the sequence.

TABLE IV

ESA operational modes controlled by the ESA driver

ID	Mode	Description
00	OFF	ESA power is off and driver is idle
01	Low voltage On	ESA low voltages are on and driver is active, but high voltages are off
02	High voltage On	ESA is fully on
16+n	Transition	In the process of sequencing from one mode to another, using transition sequence number 'n'

3.5.2.1. *Commanding*   The default software operation is based on a table of parameters called the ESA Instrument Configuration Table (ESA_ICT). The ESA_ICT default values are loaded upon processor reset from the IDPU EEPROM, and thereafter can be modified by the IDPU Table Load Command. This table includes high voltage levels, sweep rates, preamplifier gain settings, etc.

In addition, there are a few commands which control the ESA driver. These include the high voltage enable/disable commands, diagnostic telemetry request commands, a high voltage ramp-up command, and a Transition Sequence request command.

3.5.2.2. *Transition Sequences.*   There are 8 transition sequences, started by the IESASEQ command. Special cases of this command are the IESALVON, IESAHVON, IESALVOFF, and IESAHVOFF commands, which start sequences 0, 1, 2, and 3, respectively. Sequences 4, 5, and 6 are diagnostic test sequences. Sequence 7 is a spare for later use.

3.5.2.2.1 Sequence 0: IESALVON.   IESALVON can only be run from the OFF mode (00), and transitions to the Low Voltage On mode (01). It first turns on the ESA Low Voltage supply of each stack whose bit is set in the ESA power (ESA_PWR) entry in the ESA_ICT. Normally all bits are set, but a bit may be cleared before starting the IESALVON sequence to prevent a stack from being powered on. After a short wait (about 150 ms) for the power to stabilize, the ESA stack and formatter registers are set to their default values in the ESA_ICT, with the exception that all high voltage levels are set to zero and the high voltage enable bits are set off. At the end of the sequence, science telemetry is started.

3.5.2.2.2. Sequence 1: IESAHVON.   IESAHVON can be run from either OFF mode or Low Voltage On mode, and transitions to the High Voltage On mode. If starting from OFF, the IESALVON sequence is first run. Next the high voltage

sequence is run if the high voltage software enable key has been set (see Section 3.5.2.2.3 below). This sequence first sets all high voltage control registers in the ESA stacks to off. It sets the telemetry to 'diagnostic' mode (all ESA burst telemetry is channelled to survey memory, VC4, to ensure that it is not discarded), and turns on ESA Burst housekeeping to provide monitors of the high voltages for ground diagnostics while the supplies are turned on. The ESA High Voltage power switches are turned on, followed by a 150 ms delay. Next the sweep high voltage (SHV) supplies are enabled and ramped to their default values at a rate specified in the ESA_ICT. The ESA sweep supplies are next set to their default values. Then, the microchannel-plate (MCP) supplies are enabled and ramped up like the SHV supplies. Diagnostic telemetry mode ends, and the ESA burst housekeeping mode is changed to monitor the high voltage sweep waveforms for one spin, then turned off.

3.5.2.2.3. Sequence 2: IESAHVOFF.   ESAHVOFF can be run from any state, but is nominally run from the High Voltage On state. It ends up in the Low Voltage On mode. The ability to run from any state provides a means of switching off the high voltages safely even if they were not turned on by the IESAHVON sequence (e.g., by direct CDI command). The sequence first sets the sweep voltages to zero, then the MCP supplies, then the SHV supplies. It then pauses for about 1 s for high voltages to discharge before setting all enables off and finally setting the ESA High Voltage power switches off.

3.5.2.2.4. Sequence 3: IESALVOFF.   ESALVOFF can be run from any state, but is nominally run from the High Voltage On or Low Voltage On state. It ends up in the OFF state. The sequence first executes the IESAHVOFF sequence to be sure that the high voltages are off. Next all ESA telemetry is disabled. Finally, all ESA power switches are turned off.

3.5.2.2.5. Sequence 4: Pulse Height Distribution Test.   This sequence measures the pulse height distribution generated by the ESA detectors, to determine the optimal MCP high voltage settings. It performs this by ramping the preamplifier Gain Digital to Analog Converter (GDAC) slowly while collecting data with the sweep voltages fixed at 110 V (sampling 900 eV particles). GDAC sets the threshold for MCP charge pulses that are counted. Thus the count rate reflects an integral of the pulse height distribution above a threshold set by GDAC. The test is repeated at five MCP high voltage settings centered at the current nominal setting. The five test cycles are run with MCPs set to nominal -100 V, nominal -50 V, nominal, nominal $+50$ V, and nominal $+100$ V. This test is intended to be done on-orbit with the high voltages on and the spacecraft in a stable plasma environment. High rate burst telemetry is collected and redirected to Survey memory (VC4) to ensure that the data is saved. On completion, the instrument is returned to the mode in which it started.

3.5.2.2.6. Sequence 5: Noise Test.   This test checks the background noise levels to be sure that the preamplifier gain settings are correct. It is nominally designed to be run with high voltage off, so that only noise is measured, but the high voltage registers are not modified by this sequence. The instrument setup consists of redirecting Burst telemetry to Survey memory (VC4) to ensure that the data is saved. No other registers are modified. The GDAC is then ramped exactly as in the PHD test described above. At the end of the test, the high rate telemetry is returned to its nominal VC2 and the instrument is returned to its previous mode.

3.5.2.2.7. Sequence 6: Pulser Test.   This test checks out the ESA electronics using the internal test pulsers. It also checks the GDAC threshold levels by ramping up the test pulser amplitude with the GDAC levels kept at their nominal levels. It is nominally designed to be run with high voltage off, so that only noise is measured, but the high voltage registers are not modified by this sequence. The instrument setup consists of redirecting Burst telemetry to Survey memory (VC4) to ensure that the data is saved. Next the test pulsers are enabled in fixed (non sweeping) full rate mode. The test Pulser gain Digital to Analog Converter (PDAC) is then ramped through a sequence of levels, with pauses for data collection between levels. The data collection delay is the same scheme used for Sequence 4 and 5.

At the end of the PDAC sweep, the test pulsers are changed from 'Fixed' to 'Sweep' mode, where the test pulse rate is proportional to the sweep high voltage (it does not matter if the high voltages have been enabled, although the sweep mode amplitude must be set above zero to get a reasonable pattern). It waits in this mode for data collection, then returns the registers to nominal values and stops. The sweep mode tests the relative timing between data collection and the instrument's energy sweep.

3.5.2.3. *High Voltage Enable.*   The IESAHVON sequence is enabled by the software high voltage enable key. This key was implemented to prevent unwanted application of high voltage which could damage the electrostatic analyzers. This key can be set using the IESAHVEBL command with the proper KEY value as a parameter (command IESAHVEBL KEY= n). The key can be reset (to disable future attempts to turn on the high voltages) using the IESAHVDIS disable command. The default state for the High Voltage Enable key is disabled.

Note that even if the software enable key has been set, there are a number of other interlocks. A special CDI command to the power converter is required before power is sent to the ESA high voltages. In addition, the appropriate bits must be set in the power enable, high voltage enable, and high voltage DAC (Digital to Analog Converter) entries in the ESA_ICT. By default, the ESA_ICT registers are set for normal high voltage operation, so that all high voltages will come up in response to the IESAHVON sequence once the high voltage enable keys have been given to the software and the power converter.

3.5.2.4. *High Voltage Ramp Command.*   This command can be used to ramp up the ESA high voltages once the instrument is in High Voltage On mode. This is the preferred way to change an MCP or SHV high voltage supply setting after the default has been set by the IESAHVON command. The command has three parameters. First the ESA stack name (ESA1, ESA2, ESA3, or ESA4); next the supply name (MCPA, MCPB, MCPC, MCPD, SHV1, or SHV2); and finally the desired setting (LEVEL= n). The level is set in DAC units (0–255). If the requested level is less that the current setting, the supply is reprogrammed to the new level in one step. If the requested level is greater than the current level, the supply is ramped up using the same scheme that is used for the IESAHVON transition sequence, which is controlled by entries in the ESA_ICT table.

3.5.2.5. *Diagnostic Telemetry Commands.*   There are two diagnostic packet types (VC5) related to ESA operation: ESA burst housekeeping and CDI/ESA.ICT dump. They can be requested by command or can be generated periodically.

   ESA analog housekeeping is normally reported with VC0 and VC1 housekeeping packets at a slow sample rate. Higher sample rates are sometimes desirable (e.g., during high voltage ramp-up, or to monitor the waveform on sweep supplies). The ESA Burst Housekeeping samples a selected ESA housekeeping parameter from each stack 1024 times per spin (about 200 samples per second). There are a number of different modes which determine how the ESA housekeeping channel select is controlled. The mode is reflected in the housekeeping packet header. ESA Burst Housekeeping is requested automatically during some transition sequences (e.g., IESAHVON) or is commandable with a selectable number of packets.

   A diagnostic packet is generated periodically containing the current contents of all the IDPU control registers (CDI, 256 registers, 512 bytes), plus the ESA_ICT table (the first 512 bytes). The rate at which these packets come out can be set from once a second to every 65535 seconds (18 hours). This packet is primarily designed to diagnose problems, and to ensure that the state of the instruments is known for ground processing.

3.5.2.6. *Automatic Instrument Controls.*   While in an operating mode, the ESA software performs a number of tasks to dynamically control the ESA analyzers, and to protect the instrument from faults. These tasks include sESA sweep voltage tracking, peak count rate and energy flux determination, ESA deflector control, and high voltage over-current protection.

3.5.2.6.1. sESA Sweep Voltage Tracking.   The sESA (stepped electrostatic analyzer) tracking mode allows the 12 ESAs that generate the sESA data to perform high time resolution measurements of electrons over a limited energy range determined by a variable algorithm that uses the energy spectra from the eESA (electron electrostatic analyzer). The sESA Tracking mode involves having the Processor set the sESA sweep mode Top Energy step to track the energy at which the highest

count rates is observed by the eESA. The Formatter contains a circuit to find the peak counts in the eESA data over a selected range of energies and anodes. The processor reads this data once per sweep, and passes the information to the sESA Tracking Module. This module then computes the new sESA Top Energy step, which can be set '*n*' steps above or below the 'peak count' energy step, and reprograms the sESA sweep mode accordingly. In tracking mode, the sESA produces a 6 energy × 16 pitch angle measurement of a portion of the electron distribution function determined by the spectral peak. Both the tracking rate (number of eESA sweeps between sESA adjustments) and the energy spacing between the 6 pairs of sESA analyzers are selectable.

While in tracking mode, the sESAs can also be set to toggle between 2 energy steps producing a 12 energy step measurement with $\frac{1}{2}$ the time resolution, or to toggle between 4 energy steps producing a 24 energy step measurements with $\frac{1}{4}$ the time resolution. We note that the sESA was operated without tracking (fixed energy steps) during most of the first two years to facilitate analysis of long time periods, however most sESA operations during the following years have used this tracking.

3.5.2.6.2. Peak Detector. The formatter contains 'Peak Detector' circuits which find the energy at which the maximum count rate occurs in the eESA and iESA analyzers. Values in ESA_ICT are used to program the energy range and anode set that are used in the search. Anodes are grouped into 8 sets covering 45 deg each. The ability to select a piece of pitch angle space with the anode mask is desirable. This involves de-spinning the anodes, and rotating the anodes to align them with the magnetic field before selecting which anodes are to be used in the peak detector. The Formatter peak detector hardware does the anode de-spinning. (The de-spinning can be enabled or disabled, and the direction of de-spin controlled.) The de-spin is done to the closest anode at the start of the spin, and the rotation of the anode mask to align it with the magnetic field is done by software. Individual stacks can also be disabled from the peak detect computation. Peak Detector results are accumulated into a buffer for use by the Burst and Fast/Slow survey triggers. The values are also telemetered in the Triggers diagnostic packet.

3.5.2.6.3. Deflector Control. The eESA and iESA analyzers have deflectors designed to shift their Field of View (FOV) to include the magnetic field vector. Two deflectors are contained in each analyzer: one to deflect up and one to deflect down. The FOV can be deflected by up to about ± 7 deg (the spacecraft orientation should keep the magnetic field direction within these limits most of the time). Software controls the amplitude of the deflector setting, based on magnetic field information obtained from the science magnetometer. Hardware modulates this amplitude with the analyzer sweep voltage, since the deflector setting is proportional to the energy of the particles for a given deflection angle.

The deflector control computes the sign and amplitude of the deflector control setting relative to the sweep voltage, and programs the appropriate part of the eESA and iESA Sweep Mode registers. The deflector control module receives the magnetic field vector and magnitude (B1NT and BMAGNT) from the Fields Driver, and the result of the calculation is converted from 2's complement to sign-magnitude for the deflector control hardware and sent to the instruments.

3.5.2.6.4. High Voltage Over-current Trips and Other Protection.   The software periodically checks the high voltage current monitors to ensure that the supplies are not exceeding a programmable current limit. If the current limit is exceeded for longer than a specified interval, then the supply is shut off. When a supply is shut off, a bit is set in the Fast Digital Monitor word. The time at which the most recent supply shut-down occurred is also recorded, and may be extracted from the ESA_ICT Diagnostic Telemetry packet. If a supply exceeds its limits for less than the specified time, a 'Glitch' bit is set for that supply which may also be monitored in the ESA_ICT Diagnostic Telemetry packet. The current trip limits are specified in the ESA_ICT. They default to about 20 mA for the MCP supplies, and 8 mA for the SHV supplies. The MCP supplies also have a special limit which applies when the MCP voltage setting is less than about 200 V. The trip current is much lower in this case, about 6 mA. This allows early detection of a shorted MCP supply, before the voltage goes very high.

In addition to current limit monitoring, there are some consistency checks made on the programmed high voltage settings. The high voltage optocoupler enables (see Carlson *et al.*, 2001, for high voltage hardware description) are switched off if the corresponding SHV supply is disabled. Also, if the SHV supply level is not high enough to support a programmed sweep top voltage step, the top voltage step is reduced. The ESA_ICT defines what minimum SHV setting is required to support a given sweep.

3.5.2.7. *Telemetry.*   The ESA driver makes headers for all ESA science packets, diagnostic packets, and housekeeping packets. Science packet headers include a number of instrument configuration items. These are assembled synchronously with the start of data collection. The exception is the sESA Top energy, which is sampled several times during packet collection. The packet headers also include the particle mode setting from the ESA_ICT table. ESA Housekeeping packets include the status of the ESA high voltage enable registers and sweep modes (copied from the CDI status table), and the most recent readings of the ESA analog housekeeping (16 values per stack). The ESA analog housekeeping is sampled periodically, cycling through the samples roughly 8 times a spin (about 1.2 samples per second). The sweep voltages and currents are sampled synchronously with the spin so that the sample will always take place at the same part of the sweep.

3.5.2.8. *Routine Operations.*   For routine operations, the commands required in Mode strings are typically a few commands to change some default values in the ESA_ICT, followed by the IESAHVON command. Typically, the Fast to Slow and Slow to Fast Survey commands change only the ESA Averager mode registers in the Formatter, by direct instrument CDI commands. The IESADIAG command may also be included to generate diagnostic packets more often during a pass. The mode command table should include a table load command to the ESA_ICT table to set the particles mode number, so that it may be reported in the telemetry packet headers.

3.5.3. *The TEAMS Driver*

The TEAMS mass spectrometer consists of an electrostatic analyzer combined with a time-of-flight velocity analyzer that together perform a 3-D measurement of the ion distribution function covering 4 ion masses (Klumpar *et al.*, 2001). Using spin-phase information generated by the IDPU, the TEAMS instrument sorts the incoming ion flux by spin phase, mass, and energy, and accumulates these events in a local memory. IDPU flight software transfers accumulated data from the TEAMS local memory through the formatter card and into the IDPU Mass Memory. In the process, the formatter card provides '16-bit to 8-bit', '14-bit to 8-bit' counter compression, and 'no' compression options depending on the type of data being transferred.

TEAMS table loads, time-of-flight high voltage setting and other instrument commands are serviced directly by CDI commands, while instrument analog levels are read back in 'slow' housekeeping (Section 3.2.2). This leaves the driver the functions of: (1) ramping up the MCP voltage at the start of each pass, (2) selecting which area of the TEAMS memory to transfer, (3) coordinating the transfer and (4) formatting the header for the resulting data.

The TEAMS instrument provides 8 data product types at a variety of rates controlled by internal registers. To keep the readout pattern in sync with the accumulation, the software driver uses the current spin phase plus identification in the particle mode commands to identify one of 13 required readout patterns and to determine which TEAMS memory page to read. The page number is then sent via a CDI command to the TEAMS unit before the transfer is initiated in the formatter card. Worst-case delays in CDI latency and the large VC2 data volume combine to make the TEAMS channel operate nominally at 95% of capacity, limiting the spin period to 4.7 s or greater.

The TEAMS interface presented an unexpected challenge as the only data generator which provided a mixture of VC2 and VC4 data. Since the hardware Mass Memory pointers in the formatter card are double-buffered, the TEAMS driver must predict what VC memory is going to be used in the next transfer before starting the current transfer. This is implemented by maintaining a simple 3-element transfer FIFO. In order to verify that all TEAMS data types in all 13 readout pat-

terns were read in the proper phase despite FIFO-induced variations, driver logic had to be prototyped and characterized using a ground simulator.

4. Summary

The FAST satellite requires an intelligent, flexible Instrument Data Processing Unit (IDPU) in order to operate the large array of instruments and to collect the high rate data required by the scientific objectives. The IDPU controls and routes commands to the various instruments and electronics boards, controls the power system, oversees instrument housekeeping and boom deployments, directs data to the mass memory, generates data headers, and manages the data selection and triggering algorithms that optimize the data. A design philosophy was chosen to develop autonomous instruments that require minimal commanding and a data recording system that minimizes processor interaction. Custom designed field programmable gate arrays perform numerous tasks normally delegated to the processor. With a reduced work load for the processor, the software focuses on data optimization schemes that maximize the science return. The innovative design of the FAST IDPU has been essential to the success of the mission. It is hoped that this description can be used in the future to help integrate multiple spacecraft instruments into a single experiment.

References

Carlson, C. W., McFadden, J. P., Turin, P., Curtis, D. W., and Magoncelli, A.: 2001, 'The Electron and Ion Plasma Experiment for FAST', *Space Sci. Rev.* (this issue).

Ergun, R. E., Carlson, C. W., Mozer, F. S., Delory, G. T., Temerin, M., McFadden, J. P., Pankow, D., Abiad, R., Harvey, P., Wilkes, R., Primbsch, H., Elphic, R., Strangeway, R., Pfaff, R., and Cattell, C. A.: 2001, 'The FAST Satellite Electric Field and Magnetic Field Instrument', *Space Sci. Rev.* (this issue).

Klumpar, D. M., Möbius, E., Kistler, L. M., Popecki, M., Hertzberg, E., Crocker, K., Granoff, M., Tang, L., Carlson, C. W., McFadden, J., Klecker, B., Eberl, F., Kunneth, G., Kastle, H., Ertl, M., Peterson, W. K., Shelley, E. G., and Hovestadt, D.: 2001, 'The Time-of-Flight Energy, Angle, Mass Spectrograph (TEAMS) Experiment for FAST', *Space Sci. Rev.* (this issue).

McFadden, J. P., Ergun, R. E., Carlson, C. W., Herrick, W., Loran, J., Vernetti, J., Teitler, W., Bromund, K., and Quinn, T.: 2001, 'Science Operations and Data Handling for the FAST Satellite', *Space Sci. Rev.* (this issue).

Pankow, D., Besuner, R., Ullrich, R., and Wilkes, R.: 2001, 'Deployment Mechanisms on the FAST Satellite: Radial Wires, Stiff Axials, and Magnetometer Booms', *Space Sci. Rev.* (this issue).

Pfaff, R. C., Carlson, C., Watzin, J., Everett, D., Gruner, T., Curtis, D., Pankow, D., Heetderks, H., Harvey, P., Ergun, R., and McFadden, J.: 2001, 'An Overview of the Fast Auroral SnapshoT (FAST) Mission', *Space Sci. Rev.* (this issue).