

STEREO

HET CPU24

Flight Software Requirements
February 20, 2002

PRELIMINARY

Bob Baker, Goddard Space Flight Center
Don Reames, Goddard Space Flight Center
Tycho von Rosenvinge, Goddard Space Flight Center
Kristin Wortman, CSC, Goddard Space Flight Center

1. Overview.....	3
1.1. Document Conventions	3
1.2. Applicable Documents	3
1.3. Acronyms	3
2. Host System and Interfaces.....	3
2.1. System Overview	3
2.2. CPU24 Microprocessor.....	4
2.2.1. Code Memory	5
2.2.2. Memory Map.....	5
2.2.3. I/O Bus (G-Buss) Peripherals	5
2.2.4. Boot ROM.....	5
2.3. External Interfaces	5
Interface between the SEP and the HET and SIT sensors	5
2.4. Hardware/Software Interfaces.....	6
3. HET CPU24 Software Requirements	6
3.1. HET CPU24 Software Overview.....	6
3.2. Interrupt Processing Overview.....	7
3.3. HET CPU24 Stimulus Events Overview	8
3.4. HET CPU24 Power-On Initialization	8
3.4.1. HET CPU24 Boot-up Procedure Requirements.....	8
3.5. 1-minute Timer Interrupt	9
3.5.1. 1-minute Timer Interrupt Software Requirements	9
3.6. HET CPU24 Soft Reset.....	9
3.6.1. HET CPU24 Soft Reset Requirements	9
3.7. Science Data Acquisition.....	9
3.7.1. Science Data Acquisition Software Requirements	11
3.8. Science Data Processing	11
3.8.1. Science Data Processing Software Requirements	13
3.8.2. HET Event Interrupt Service and Processing DFD	13
3.9. Beacon Data Processing.....	15
3.9.1. Beacon Data Processing Requirements.....	15
3.10. Housekeeping and Status Data Acquisition.....	15
3.10.1. Housekeeping and Status Data Acquisition Requirements	16
3.11. Science Data Formatting.....	17
3.11.1. Science Data Formatting Requirements	18
3.12. Command Processing.....	18
3.12.1. Command Processing Software Requirements	19
3.13. Reliability.....	19
3.13.1. Reliability Requirements.....	19
4. Ground Support Equipment Software.....	20
4.1. Ground Support Software Requirements	20
4.1.1. Boot Code Loading	20
4.1.2. Display of HET CPU24 Data Bytes.....	21
4.1.3. Plotting of the HET CPU24 Data Bytes.....	21
4.1.4. Command Processing To and From the HET CPU24	21

1. Overview

The HET flight software shall run on a CPU24 processor. The HET sensor shall interface with its front-end electronics and the SEP Central MISC through a series of Universal Asynchronous Receiver/Transmitters (UART).

1.1. Document Conventions

1.1.1.1 In this document, TBD (To Be Determined) means that no information currently exists.

1.2. Applicable Documents

Some of these documents and drawings can be found on the Berkeley STEREO/IMPACT website: <http://sprg.ssl.berkeley.edu/impact/dwc/>

- STEREO SEP HET and SIT CPU24 Processors, Flight Development Plan
- PHA User's Manual, Rich Cook, Caltech internal memo
- CPU24 Processor Manual, Bob Baker
- PHASOFT.DOC, Don Reames
- PHAINT.TXT, Don Reames

1.3. Acronyms

CPU24 24-Bit Embedded Microprocessor
DPU Data Processing Unit
HET High Energy Telescope
ICD Interface Control Document
IMPACT In situ Measurements of Particles and CME Transients
LET Low Energy Telescope
MISC Minimal Instruction Set Computer
SEP Solar Energetic Particles
SIT Suprathermal Ion Telescope
SEPT Solar Electron Proton Telescopes
SRL Space Radiation Laboratory
UMD University of Maryland
GSFC Goddard Space Flight Center

2. Host System and Interfaces

2.1. System Overview

The LET, HET and SIT sensors each require a dedicated microprocessor for onboard data processing. The microprocessor used for LET shall be the P24 MISC (Minimal Instruction Set Computer), described in Reference 5. The microprocessor used for HET

and SIT shall be the CPU24 (24-Bit Embedded Microprocessor), which is described below and in Reference 4. Processed data from all of the microprocessors associated with these three sensors shall be gathered by the SEP Central MISC (P24 processor), and formatted for transmission to the IMPACT DPU (per Reference 7 ICD). The SEPT sensor does not have a dedicated microprocessor, and data from SEPT shall flow directly to the SEP Central MISC. Some processing of SEPT data shall occur in the SEP Central MISC before the data are formatted and transmitted to the IMPACT DPU. Figure 1 shows a block diagram of the SEP Instrument Suite.

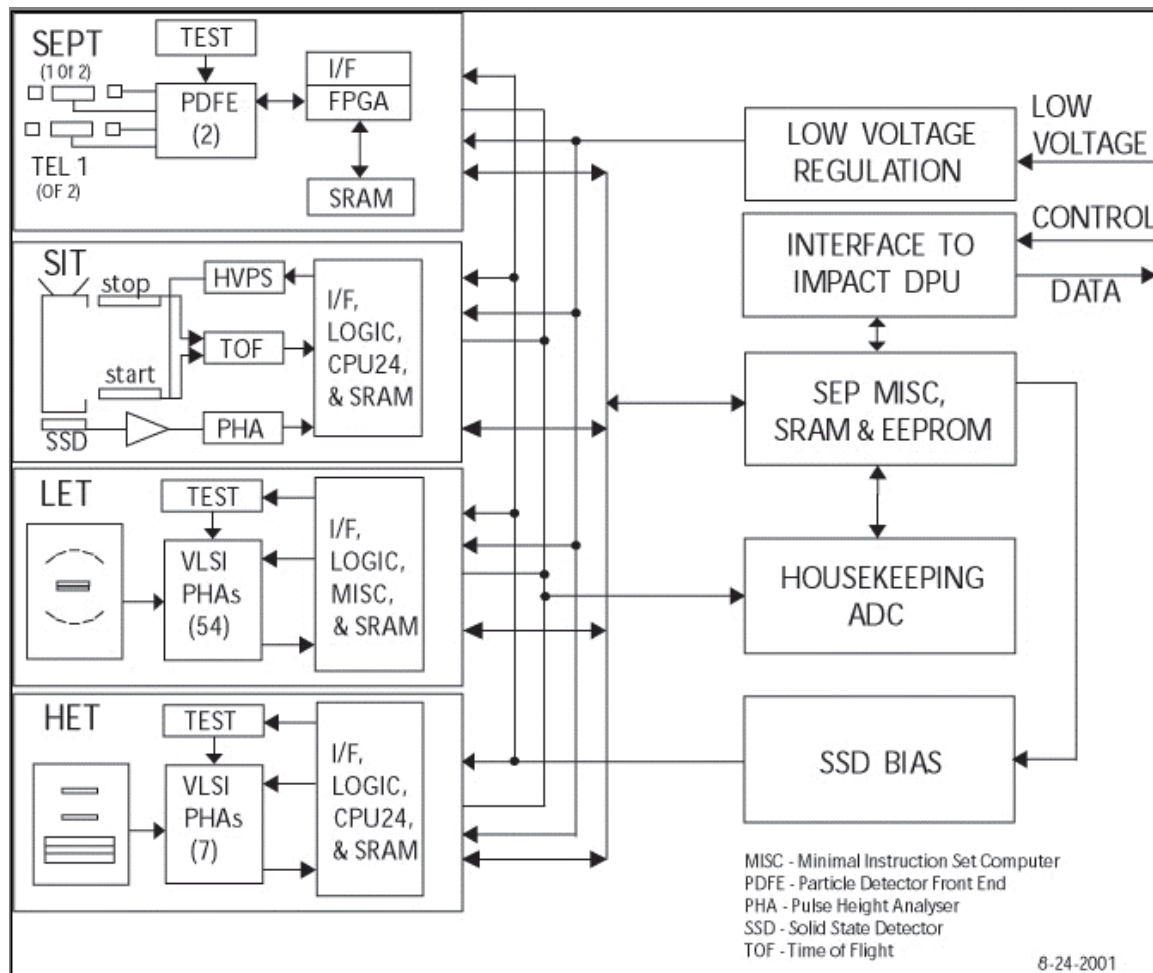


Figure 2.1: SEP Instrument Suite Block Diagram

2.2. CPU24 Microprocessor

The CPU24 has a 24-bit core with dual stack architecture. The processor design is simple to allow implementation within field programmable gate arrays. For STEREO SEP applications, the CPU24 is implemented in the ACTEL 54SX72A FPGA. The current implementation runs at 10 million instructions per second.

2.2.1. **Code Memory**

Current CPU24 development boards are provided with 128K×24 SRAM. The CPU24 boots over a serial link. During boot, system software is copied from the serial link to RAM and the software runs from RAM.

2.2.2. **Memory Map**

The CPU24 is capable of addressing a memory page of 256Kwords (each word is 24 bits). Other pages can be reached by pushing a 24-bit address on the return stack and executing the RET instruction, as described in Reference 4, but this capability is not expected to be required for STEREO (128K words of SRAM per CPU24 should be adequate).

2.2.3. **I/O Bus (G-Buss) Peripherals**

G-bus peripherals are described in detail in Reference 4. Currently, G-bus peripherals include:

```
g@0  read interrupt control and status register
g@1  read UART receiver data
g@2  read zero
g@3  read one

g!0  write interrupt control and status register
g!1  write UART transmit data
g!2  write second UART transmit data
g!3  write to test board LED's (lower four bits)
```

G-bus functions will be added to interface to the PHA ASICS and perform other functions as needed.

2.2.4. **Boot ROM**

The CPU16 contains 16 words of ROM that holds a small program to boot over the serial link. After a reset, the CPU24 starts execution at location 0 which is mapped to the internal ROM. The instruction at location 0 performs a jump to location 20001 (hex) where the remainder of the boot code is located. A fixed number of bytes are received over the serial link. Every three bytes received are packed into a 24-bit word, with the first byte going into the most significant slot and the third byte going into the least significant slot. Words are stored beginning at address 1 in SRAM. Execution begins at address 1 (in SRAM) following the serial transmission.

2.3. **External Interfaces**

Interface between the SEP and the HET and SIT sensors

There shall exist two serial interfaces between the SEP and the HET and SIT sensors. The first serial interface shall be bi-directional, for transferring boot-code, commands, and command responses. The second serial interface shall be uni-directional, for transferring

data from the HET and SIT sensors to the SEP. These interfaces shall be defined in Reference 7 (TBD-Kecman/WRC).

2.4. *Hardware/Software Interfaces*

The hardware/software interfaces for the HET and SIT processors shall be defined in the CPU24 processor manual (reference 4).

3. HET CPU24 Software Requirements

3.1. *HET CPU24 Software Overview*

The HET CPU24 shall operate as a multi-tasking, interrupt-driven processor. It shall be coded using the CPU24 native assembly language. The HET CPU24 main software logic shall accomplish the following tasks:

- Process next particle in next queue
 - a. increment events-processed counter
 - b. increment appropriate SW bin
 - c. increment appropriate SW bin
 - d. select and buffer PHA sample events
- If Cmd flag – process immediate command or queue it for EOF
- If EOF:
 - a. update hardware count rates, reset counters
 - b. begin formatting output packets for HW & SW rates and PHAs.
 - c. process any EOF commands to affect next frame.
 - d. run algorithm enable/disable H1 singles processing at high rates
 - e. run algorithm to raise/lower thresholds at high rates
 - f. initialize and clock out any HW setup serial strings
 - g. clear all SW & HW counters, clear event queues.
 - h. clear EOF flag
- Continue formatting output packets, if any. This is done in pieces so that particle processing can continue, *i.e.* to prevent queues from filling.

The HET CPU24 shall consists of the following software modules to perform the above logic:

- Science data acquisition
- Science data processing
- Beacon data processing
- Housekeeping and status data acquisition
- Command processing
- Science data formatting

3.2. *Interrupt Processing Overview*

The HET CPU24 software system shall be responsive to servicing several interrupts, which are generated by the SEP Central MISC and the HET front-end electronics. These interrupts are defined as the following:

- **1-second timer:** Time synchronizing pulses from the SEP Central MISC will be 1 ms wide positive pulses with a 1 second period. Use to increment second clock (0-59).
- **1-minute timer:** Every 60th 1-sec pulse will be followed, after a 1 ms delay, by an extra 1 ms wide pulse (the 1-minute timer pulse). The basic sensor data accumulation interval will be one minute, starting at the first 1-second timer pulse following the 1-minute timer pulse (i.e. reset the second clock to 0 at this time). The HET CPU24 will send data to the SEP Central MISC during a 100 ms window following 1-second timer pulses 0, 3, 6,, 57. HET may send no packets or more than 1 packet during a window, but packets should be spread out over the 60 second cycle. The 1-minute timer pulse may also be referred to as a frame synch pulse or end-of-frame (EOF) pulse since rate counters should be read out and reset at this time, packets should be queued for readout into the telemetry stream, and any status bits (e.g. the high-rate/low-rate bit) should be captured and readout.
- **Serial Command In:** There is a serial line which sends command information from the SEP Central MISC to the HET CPU24 via the receive portion of the Command UART. An interrupt is set by the UART to indicate that the next command byte has been received by the UART and is ready to be read by the HET MISC to the command buffer. Boot-up of the HET MISC also uses the Command UART.
- **Serial Command Out:** The Command UART also transmits Command Response information to the SEP Central MISC. This response information is to be printable ASCII text. An interrupt from the Command UART signals the MISC that it is ready to get the next byte to be transmitted from the command response buffer.
- **Serial Data Out:** The 'transmit' half of a UART (referred to as the Data UART) sends formatted data from the HET CPU24 to the SEP Central MISC. The Data UART sends the Serial Data Out interrupt to the MISC when it is ready to get another byte of data to send out.
- **PHA:** The HET front-end logic is in the same ACTEL chip as the HET MISC. When the front-end logic detects that a pulse height event has been collected, the front-end logic sends an interrupt to the HET CPU24 to go read the event out of the PHA ASICs. The pulse-height values for an event

are read out using a 24-bit wide bus (the G-bus). The interface for this is described in the PHA User's Manual (Rick Cook, Caltech internal document). When an event has been read it should be queued into one of 4 queues:

- Stopping in H1 (H1inner or H1outer)
- Stopping in H2-H5
- Penetrating
- Pulser stimulation event

The PHA-in counter and the appropriate PHA-queued counter (if queued) should also be incremented correspondingly.

3.3. *HET CPU24 Stimulus Events Overview*

The HET CPU24 has the capability to stimulate the ASIC on-board stimulus system once per second. The sizes of the stimulus pulses are under the control of the CPU24. Specifically, there is a DAC accessible by the CPU24, which sets the pulser reference voltage, and there are ASIC configuration control bits which set the size of the coupling capacitors from the test pulser to the inputs of the preamplifiers. In the background task the stimulus amplitude shall be slowly sequenced from the smallest pulse size to the largest, providing a long-term history of PHA performance for each PHA over its entire dynamic range.

3.4. *HET CPU24 Power-On Initialization*

In flight configuration, it is expected that the HET CPU24 will boot via the serial link from the SEP Central MISC. Therefore, only the SEP Central MISC will need EEPROM during flight.

After the initial boot code has loaded, the HET flight software shall be transferred over the serial link and loaded from the SEP Central MISC to the HET CPU24.

Once the boot process has completed, the HET CPU24 shall configure the HET front-end electronics into a default state. The default state shall be defined in tables loaded into SRAM during the boot process.

3.4.1. *HET CPU24 Boot-up Procedure Requirements*

3.4.1.1 The HET CPU24 boot-up shall be able to download code and tables from the EEPROM in the SEP Central MISC.

3.5. *1-minute Timer Interrupt*

The 1-minute timer pulse shall invoke the rate counters to be read out and the PH event buffer to be formatted into science packets and be prepared for output. The End-Of-Frame flag shall be set in response to the 1-minute timer interrupt.

3.5.1. 1-minute Timer Interrupt Software Requirements

3.5.1.1 The 1-minute timer interrupt shall respond by setting the End-Of-Frame flag.

3.6. *HET CPU24 Soft Reset*

The HET CPU24 soft reset shall respond to command from the SEP Central MISC. The soft reset shall initialize all the hardware, and software counters and memory buffers; this includes resetting the End-Of-Frame (EOF) flag.

3.6.1. HET CPU24 Soft Reset Requirements

3.6.1.1 The HET CPU24 soft reset shall initialize all the UARTs.

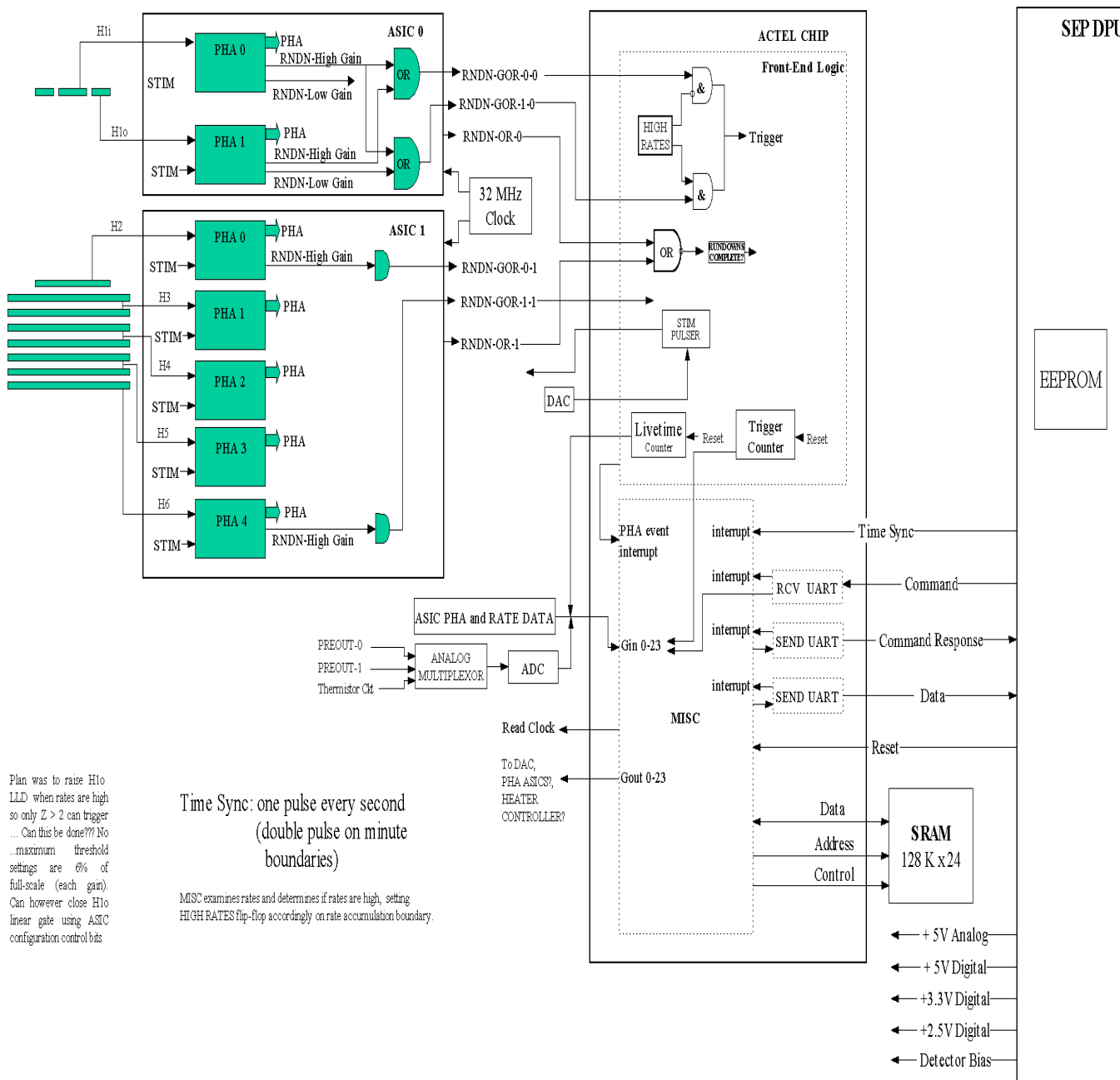
3.6.1.2 The HET CPU24 soft reset shall initialize and clock out all the hardware setup serial strings.

3.6.1.3 The HET CPU24 soft reset shall clear all the software and hardware counters, queues and output packet buffers.

3.6.1.4 The HET CPU24 soft reset shall wait for the End of Frame (EOF) flag, then clear it and start the main processing loop.

3.7. *Science Data Acquisition*

The HET flight software shall be responsible for acquiring science data from the HET front-end electronics. Event control logic on the MISC FPGA shall interrupt the MISC processor when event data is ready to be read into memory. Refer to the following HET data flow block diagram.



HET Logic/Data Flow Block Diagram

3.7.1. Science Data Acquisition Software Requirements

3.7.1.1 The HET data acquisition software shall respond to the PHA event interrupt by reading a PHA event from the ASICS using a 24-bit wide bus (G-bus).

3.7.1.2 The HET data acquisition module shall determine the class type based upon the event coincidence tags, the IDs of the hit detectors, and the raw pulse-height-data.

3.7.1.3 The HET data acquisition module shall store each class type of PH events in a separate queue using first-in-first-out (FIFO) ordering.

3.7.1.4 The HET data acquisition module shall count all PH events read from the PHA ASICs. This count shall include all defined valid and invalid PH events.

3.7.1.5 The HET data acquisition module shall count each PH event for each class type. These counts shall include all the PH events, which are abandoned when the maximum queue length is reached.

3.7.1.6 The HET data acquisition module shall count all PH events, which are defined as having duplicates of H1 (inner and outer) pulse heights.

3.7.1.7 The HET data acquisition module shall count all PH events, which are defined as not having H1 (inner or outer) pulse height as the first one being read out of the ASICs.

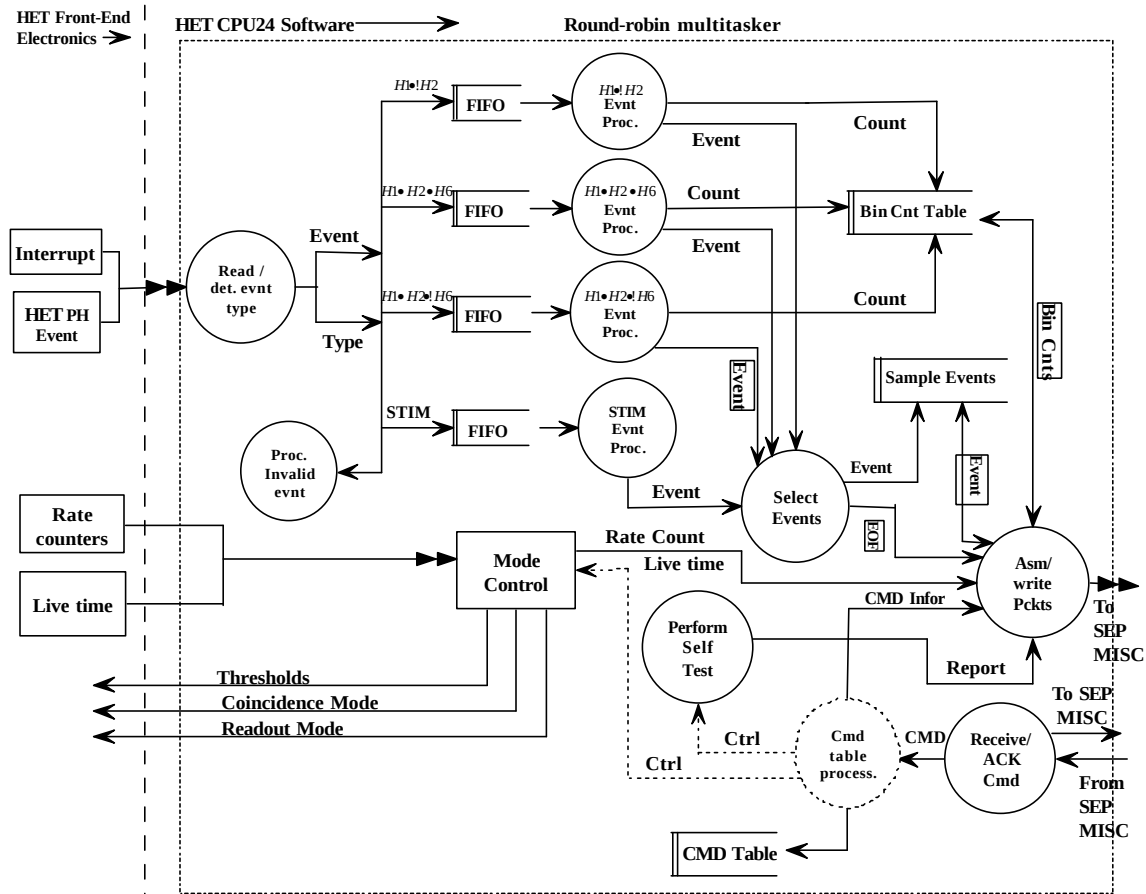
3.7.1.8 The HET data acquisition module shall count all PH events, which are defined as not being in layer sequence.

3.8. Science Data Processing

The telemetry bandwidth allocated to HET is adequate to telemeter only a fraction of the events recorded by the instrument. Therefore the HET software is required to process most events onboard. The objective of the onboard event processing software is to analyze the data gathered during each event and assign a species and energy to the particle that generated the event. The species and energy determination shall be carried out using a 2-dimensional detector response matrix appropriate for the class of event being processed.

Once the software has determined the position of an event in the response matrix, the appropriate counter shall be incremented. It is expected that the size of a response matrix shall be of 128×128, with 100 software bins overlaid on the matrix. Each software bin shall have an associated counter. Note that this means that many cells of the response matrix shall lie within the same software bin and shall be associated with the same counter. The counters associated with the software bins shall be read out periodically, and this constitutes the data, which shall be telemetered.

When the HET CPU24 receives an event interrupt, the interrupt service routine shall read the pulse-height data from the HET ASICs. A software routine shall read the event and assign a class to each event. The HET event types shall be determined based upon the event coincidence tags, the IDs of the hit detectors, and the raw pulse-height-data. After the event type has been determined the event shall be passed into one of a series of queues, which has been designated for that type. Higher-level event processing routines shall read and process events from these event queues. Refer to the following diagram, which shows the HET data flow through the HET CPU24.



KAW 11/19/01

3.8.1. Science Data Processing Software Requirements

3.8.1.1 The HET event processing module shall select the PH events from each class type queue and store them in a sample event buffer. This buffer shall be used for output in the science data packet.

3.8.1.2 The HET event processing module shall determine species and energy by using a 2-dimensional detector response matrix appropriate for the class of event being processed.

3.8.1.3 The HET event processing module shall respond to the EOF reset by resetting the software counters.

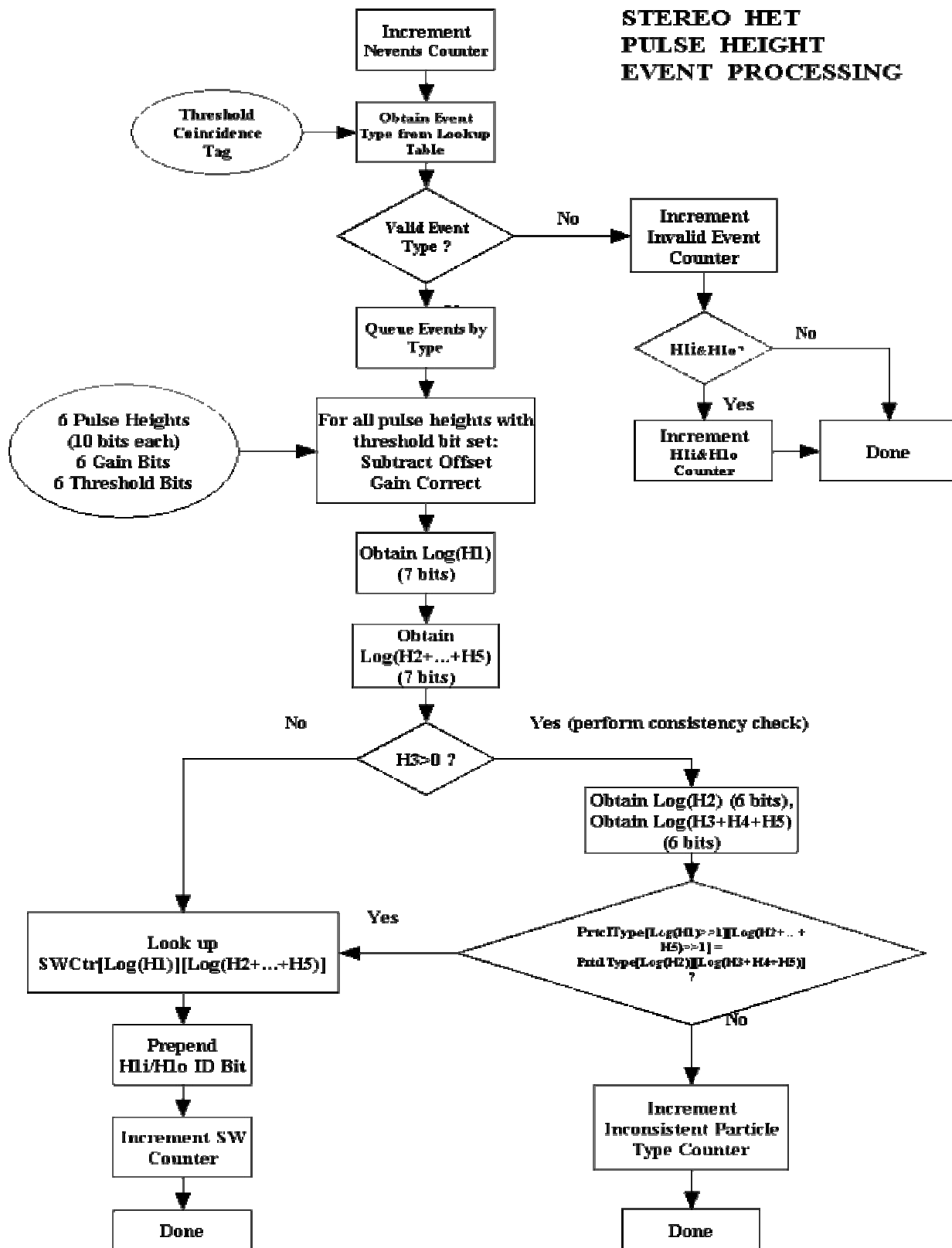
3.8.1.4 The HET event processing module shall respond to the EOF reset by resetting the hardware counters.

3.8.1.5 The HET event processing module shall respond to the EOF reset by clearing all the event queues.

3.8.2. HET Event Interrupt Service and Processing DFD

The following data flow diagram shows the logic flow for the HET event interrupt serving and processing.

STEREO HET PULSE HEIGHT EVENT PROCESSING



TvR, 26 August 2001

3.9. ***Beacon Data Processing***

Beacon data shall be transmitted to the ground by the SEP Central MISC and are not to be included in the 6 packets per minute allocation. Beacon packets shall be generated once per minute by the SEP Central MISC using sub-packets sent each minute from the HET sensor.

3.9.1. **Beacon Data Processing Requirements**

3.9.1.1 The HET beacon data processing module shall calculate beacon data every 60 seconds.

3.9.1.2 The HET beacon data processing module shall calculate beacon data for the following species and energy ranges:

Species	Energy Range (MeV/nucleon)
Protons	13 - 30
Protons	30 - 50
Protons	50 – 100
He	13 - 30

3.9.1.3 The HET beacon data processing module shall format the beacon data into a packet of 272 bytes.

3.9.1.4 The HET beacon data processing module shall transmit the beacon data packet every minute to the SEP Central MISC via the data-out UART.

3.10. ***Housekeeping and Status Data Acquisition***

The SEP Central MISC shall be responsible for gathering housekeeping data (voltages, currents, temperature, checksums etc.). Housekeeping data are transmitted to the ground by the SEP Central MISC and are not be included in the 6 packets per minute allocation.

3.10.1. Housekeeping and Status Data Acquisition Requirements

3.10.1.1 The HET housekeeping and status data acquisition module shall read HET-specific software status information from the SEP Central MISC.

3.10.1.2 The HET housekeeping and status data acquisition module shall read HET-command table information from the SEP Central MISC.

3.10.1.3 The HET housekeeping and status data acquisition module shall read HET-specific software housekeeping information from the SEP Central MISC.

3.10.1.4 The HET housekeeping and status data acquisition module shall format HET-specific software status information for the science data packet.

3.10.1.5 The HET housekeeping and status data acquisition module shall format HET-specific software command table information for the science data packet.

3.10.1.6 The HET housekeeping and status data acquisition module shall format HET-specific software housekeeping information for the science data format.

3.10.1.7 The HET housekeeping and status data acquisition module shall determine when the HET is to be switched from high-rate mode to low-rate mode. This switching can only occur on 1-minute boundaries.

3.10.1.8 The HET housekeeping and status data acquisition module shall determine when the HET is to be switched from low-rate mode back to high-rate mode. This switching can only occur on 1-minute boundaries.

3.10.1.9 The HET housekeeping and status data acquisition module shall be able to force the mode into either high-rate mode or low-rate mode (i.e. by-pass the mode-determining algorithms). This switching can only occur on 1-minute boundaries.

3.10.1.10 The HET housekeeping and status data acquisition module shall be able to monitor the output voltage of each preamplifier.

3.10.1.11 The HET housekeeping and status data acquisition module shall be able to adjust the corresponding ASIC on-board current source in order to maintain an output voltage within a specified range.

3.10.1.12 The HET housekeeping and status data acquisition module shall be able to stimulate the ASIC on-board stimulus system once per second.

3.10.1.13 The HET housekeeping and status data acquisition module shall be able to adjust the sizes of the stimulus pulses.

3.10.1.14 The HET housekeeping and status data acquisition module shall format the housekeeping and status data into a packet of 272 bytes.

3.11. *Science Data Formatting*

Science data is to be transferred from the IMPACT DPU (IDPU) in packets of fixed length of 272 bytes, which includes an 11-byte header. HET is allocated 6 packets per minute. Each packet shall include a checksum in order to be able to discriminate against data with bit errors.

Data packets collected from the instruments during a given accumulation minute (N) by the SEP Central MISC shall contain data accumulated during the previous minute (N-1) and shall be transmitted to the IDPU during minute N+1. The SEP Central MISC shall fill in the UT portion of all packet headers with the time corresponding to the beginning of the accumulation minute.

Special modes:

- **Accelerator testing mode:** In this mode the SEP Central MISC only pays attention to one sensor at a time. The sensor can use the full 57.6 kbaud bandwidth of the serial data line. The data format from the sensor can be anything in this mode, but could be in the form of multiple packets per second.
- **Variable packet rate:** The number of packets per minute generated by HET should be commandable. This allows one sensor to make use of available bandwidth should another be disabled. Currently, in addition to HET, SIT transmits 12 packets per minute in 100 ms windows following 1-second timer pulses 1, 4, ..., 58; LET transmits 16 packets per minute following 1-second timer pulses 2, 5, ..., 59; and SEPT has 2 packets per second handled separately by the SEP Central MISC.

The science data formatting software shall be responsible for formatting the HET science data into minor frames, in response to the End of Frame Flag (EOF) being set. The SEP Central MISC shall perform formatting of HET science frames into CCSDS packets. Checksum values shall be calculated for each packet in order to be able to discriminate against data with bit errors.

3.11.1. Science Data Formatting Requirements

- 3.11.1.1 The HET data formatting module shall execute in response to the End of Frame (EOF) flag being set.
- 3.11.1.2 The HET data formatting module shall retrieve the rate counters and the live time from the SEP buffers.
- 3.11.1.3 The HET data formatting module shall read the ASIC single rates using the G-buss.
- 3.11.1.4 The HET data formatting module shall retrieve the sample PH event data from a buffer.
- 3.11.1.5 The HET data formatting module shall format the HET science data (rates and PH events) into a science data packet.
- 3.11.1.6 The HET data formatting module shall format 6 science packets accumulated over one minute.
- 3.11.1.7 The HET data formatting module shall format each science packet to be 272 bytes in length, which includes an 11-byte header.
- 3.11.1.8 The HET data formatting module shall calculate a checksum value for each science data packet.
- 3.11.1.9 The HET data formatting module shall write the checksum value in the science data packet.
- 3.11.1.10 The HET data formatting module shall transmit the science data packet to the SEP Central MISC via the data-out UART at the rate of 6 packets per minute.
- 3.11.1.11 The HET data formatting module shall reset the End-Of-Frame (EOF) flag.

3.12. Command Processing

Commands to the HET CPU24 shall be used to set configuration control bits for each of two PHA ASICS. The two ASICS each contain a serial shift register of roughly 1200 bits to acquire the corresponding configuration control bits. The shift registers in the two ASICS are coupled together to form one long shift register. The contents of this shift register shall be refreshed once every minute. A default state for this shift register shall reside in EEPROM. Reverting to the default state and making changes to this state shall be possible without having to specify the whole state from scratch. The state of this register shall be trickled to the ground every TBD minutes as part of the HET housekeeping data.

Commands for HET shall be received via the command data-in UART connection to the SEP Central MISC. Command responses from the HET CPU24 shall be sent via the command data-out UART connection to the SEP Central MISC, where they will be formatted for telemetry. Flags shall be used in the HET science data packets to indicate error conditions. Refer to the HET Event Data Flow diagram.

Command responses are transmitted to the ground by the SEP Central MISC are are not included in the 6 packets per minute allocation.

3.12.1. Command Processing Software Requirements

3.12.1.1 The HET command processing module shall respond to the serial command in interrupt by reading the command byte from the UART connection to the SEP Central MISC and storing it in a buffer.

3.12.1.2 The HET command processing module shall use the command buffer and lookup and retrieve the associated commands in a table.

3.12.1.3 The HET command processing module shall respond to commands to set data table values.

3.12.1.4 The HET command processing module shall execute the commands producing a status flag to indicate error conditions.

3.12.1.5 The HET command processing module shall write the command status flag to the science data packet.

3.12.1.6 The HET command processing module shall respond to the serial command out interrupt by transmitting the command byte to the UART connection to the SEP Central MISC.

3.12.1.7 The HET command processing module shall be able to send a command request for a burst of N pulses of given amplitudes and combinations of PHA channels.

3.13. Reliability

Abnormal conditions shall be detected by the HET software and reported in the science data packet. It is not expected that automatic onboard correction of error conditions shall be implemented, but whenever possible (consistent with instrument safety) the system should remain in an operational state following an error. Error conditions shall be flagged in the science telemetry and corrective action shall be initiated from the ground.

3.13.1. Reliability Requirements

- 3.13.1.1 The HET software shall be able to detect error conditions.
- 3.13.1.2 The HET software shall set flags to indicate error conditions.
- 3.13.1.3 The HET software shall write the error flags to the science data packet.
- 3.13.1.4 The HET software shall remain in an operational state following an error condition.
- 3.13.1.5 The HET software shall set flags to indicate error conditions.
- 3.13.1.6 The HET software shall maintain a command table containing all variable settings (thresholds, gains, modes, etc).
- 3.13.1.7 The flight software shall calculate checksums on the variable setting command table.
- 3.13.1.8 The HET software shall write the command table checksums to the science data packet.
- 3.13.1.9 The HET software shall monitor the command tables for changes in status, if a change occurs, a flag shall be set.
- 3.13.1.10 The HET software shall write the command table status flag to the science data packet.

4. Ground Support Equipment Software

The HET CPU24 is being built by GSFC and needs to be tested and monitored using the ground support software. This software shall support the HET CPU24 by itself and when it is connected to the SEP Central MISC, which is being built by Caltech. This support shall include the following tasks:

- Boot code loading via the UART.
- Display of data received via the UART.
- Plotting of the data received via the UART.
- Command processing via the UART.

4.1. Ground Support Software Requirements

4.1.1. Boot Code Loading

- 4.1.1.1 The HET GSE software shall be able to load download boot-up code and tables.

4.1.2. Display of HET CPU24 Data Bytes

- 4.1.2.1 The HET GSE software shall be able to retrieve data packets via the UART.
- 4.1.2.2 The HET GSE software shall be able to retrieve data packets via the TCP/IP protocol and use of Windows sockets.
- 4.1.2.3 The HET GSE software shall be able to format the data packets into the LECR format.
- 4.1.2.4 The HET GSE software shall be able to display the data packets in a readable a format.
- 4.1.2.5 The HET GSE software shall allow selection of quantities for plotting.
- 4.1.2.6 The HET GSE software shall be The HET GSE software shall be able to write the LECR format data to an output file. This shall promote archiving of the test data and allow the user to review the data at a later time.
- 4.1.2.7 The HET GSE software shall allow command file setup and processing.

4.1.3. Plotting of the HET CPU24 Data Bytes

- 4.1.3.1 The HET GSE software shall be able create 2 dimensional plots of the formatted data bytes.
- 4.1.3.2 The HET GSE software shall allow selection of quantities for plotting.
- 4.1.3.3 The HET GSE software shall allow command file setup and processing.

4.1.4. Command Processing To and From the HET CPU24

- 4.1.4.1 The HET GSE software shall be able to retrieve command bytes from the CPU24. This shall allow for simulation of commands being sent from the HET CPU24 to the SEP Central MISC.
- 4.1.4.2 The HET GSE software shall be able to send command bytes to the HET CPU24. This shall allow for simulation of commands being sent from the SEP Central MISC to the HET CPU24.

