STEREO

SEP HET and SIT CPU24 Processors

Flight Software Development Plan
April 17, 2003

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

Document Revision Record

Rev.	Date	Description of Change	Approved By
A	2001-Dec 06	Preliminary Draft	-
B	2002-May	Preliminary Draft	-
C	2003-Apr-14	Final Draft	-

Distribution List

Rick Cook, Caltech SRL
Alan Cummings, Caltech SRL
Andrew Davis, Caltech SRL
Branislav Kecman, Caltech SRL
Allan Labrador, Caltech SRL
Richard Mewaldt, Caltech SRL
Robert Radocinsky, Caltech SRL
Ed Stone, Caltech SRL
Mark Wiedenbeck, Caltech SRL
Donald Reames, GSFC
Tycho von Rosenvinge, GSFC
Kristin Wortman, CSC/GSFC
Dave Curtis, UCB
Glenn Mason, UMD
Peter Walpole, UMD

Table of Contents

1. Overview

1.1. *Introduction*

The IMPACT SEP instrument consists of four sensors – LET, HET, SEPT and SIT. Four microprocessors are dedicated to controlling, interfacing, and acquiring data from these sensors. This document defines the development plan for the HET and SIT microprocessors.

Flight software for the HET and SIT microprocessors will be developed at the Goddard Space Flight Center (GSFC). The GSFC group developed the flight software for the EPACT instrument suite, which was flown onboard the WIND spacecraft launched in 1994.

1.2. *Document Conventions*

In this document, TBD (To Be Determined) means that no information currently exists. TBR (To Be Resolved) means that a statement is preliminary. In either case, the initials of those responsible for providing the information typically follow the acronym.

1.3. *Applicable Documents*

Some of these documents and drawings can be found on the Berkeley STEREO/IMPACT website: http://sprg.ssl.berkeley.edu/impact/dwc/
Others are currently available from Caltech SRL, GSFC or UMD.

1. Phase A Report/PAIP (Performance Assurance Implementation Plan)
2. HET Flight Software Requirements (TBD-GSFC)
3. SIT Flight Software Requirements (TBD-UMD)
4. CPU24 Processor Manual
5. P24 MISC Processor Manual
6. SEPT FPGA Data Sheet
7. IMPACT Intra-Instrument Serial Interface ICD
8. SEPT Science Requirements Document
9. SEP Sensor Suite ICD (TBD-Kecman/WRC)
10. P24 MISC G-Buss I/O Interface Document (TBD-WRC)
11. IMPACT/Spacecraft ICD
12. HET-SEP Interface Control Document
13. SIT-SEP Interface Control Document

1.4. *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.

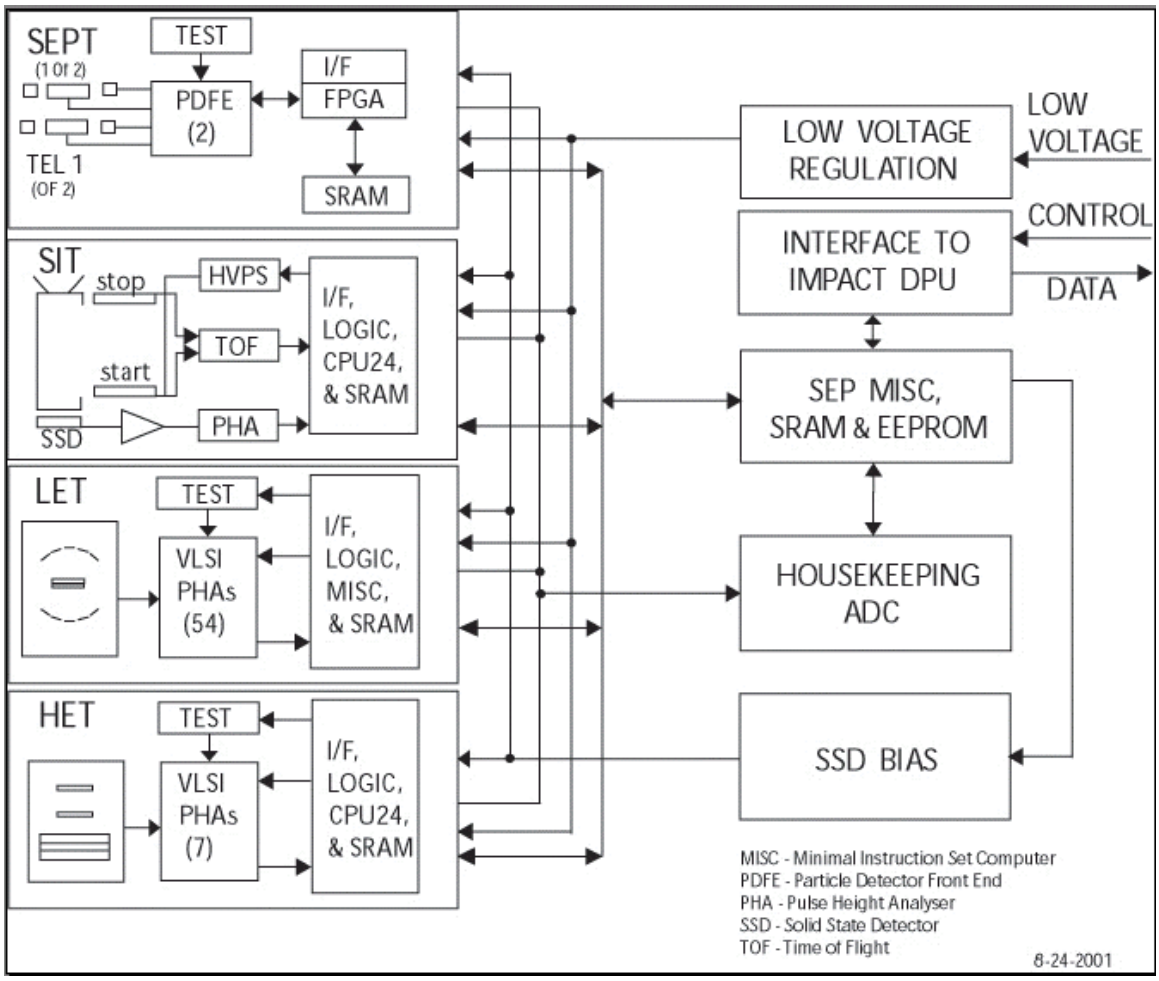

Figure 2.1: SEP Instrument Suite Block Diagram

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 interface shall be bi-directional, for transferring boot-code, commands, and command responses. The second interface shall be uni-directional, for transferring data from the HET and SIT sensors to the SEP. These interfaces are defined in the following documents: IMPACT Intra-Instrument Serial Interface ICD, HET-SEP Interface Control Document and the HET-SEP Interface Control Document and the SIT-SEP Interface Control Document.

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. Software Requirements

3.1. *Top Level Requirements*

The HET and SIT flight software shall run on two CPU24 processors, one dedicated to each sensor. Each sensor shall interface with its front-end electronics and the SEP Central MISC through serial communication ports.

The GSFC software team shall develop the software for the HET and SIT sensors in CPU24 assembly language.

3.2. *HET CPU24 Software Requirements*

The HET CPU24 software shall be developed at GSFC and shall interface with the SEP Central MISC. The HET CPU24 software shall boot via a serial link from the SEP Central MISC.

The HET CPU24 software shall be responsible for: science data acquisition and processing, beacon data processing, housekeeping and status data acquisition, and science data packet formatting and command processing.

Details of the HET CPU24 software requirements shall be outlined in the Flight Software Requirements for the HET CPU24 Processor document. Refer to figure 3.2 for a diagram of data flow through the HET CPU24.

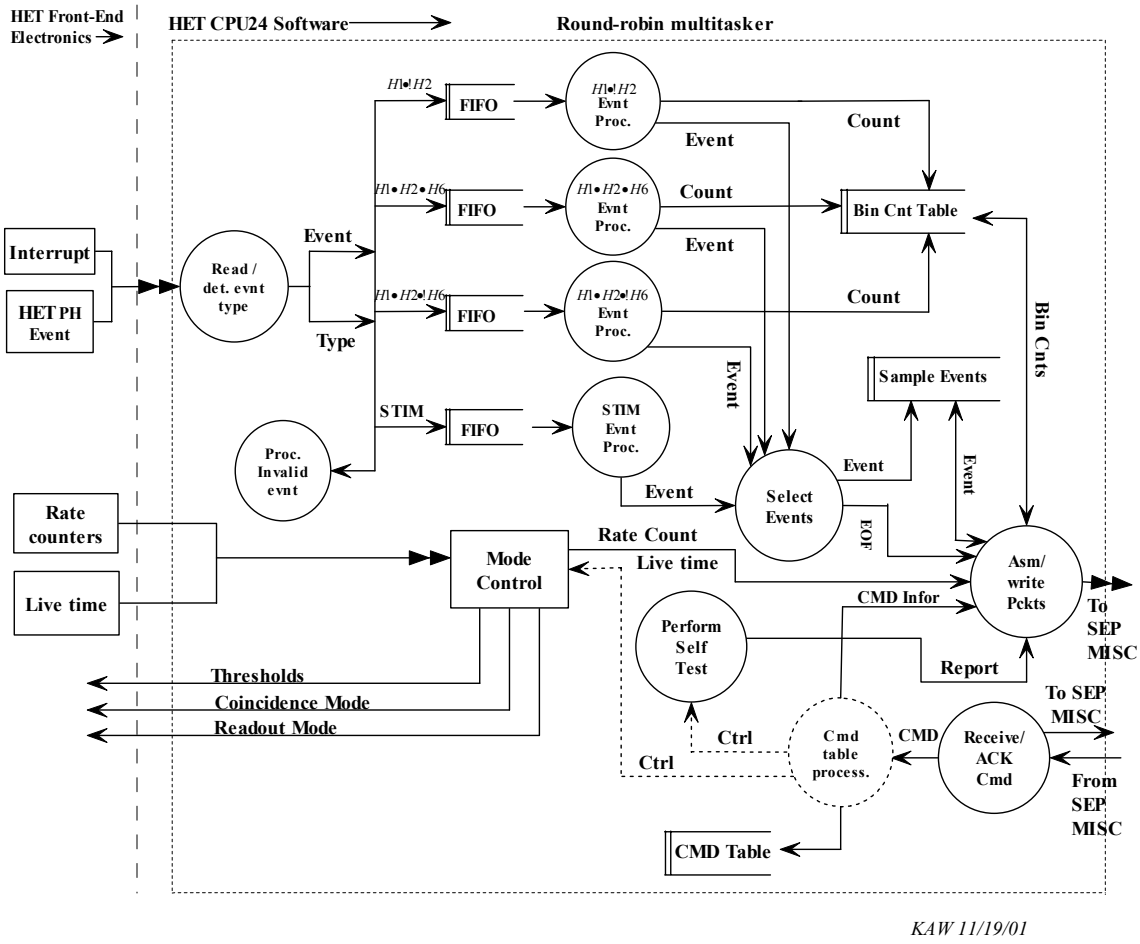

Figure 3.2 HET Data Flow thru HET CPU24

3.3. *SIT CPU24 Software Requirements*

The SIT CPU24 software shall be developed at GSFC and shall interface with the SEP Central MISC. The SIT CPU24 software shall boot via a serial link from the SEP Central MISC.

The SIT CPU24 software shall be responsible for: science data acquisition and processing, beacon data processing, housekeeping and status data acquisition, and science data packet formatting and command processing.

Details of the SIT CPU24 software requirements shall be outlined in the Flight Software Requirements for the SIT CPU24 Processor document. Refer to figure 3.3 for a diagram of data flow through the SIT CPU24.

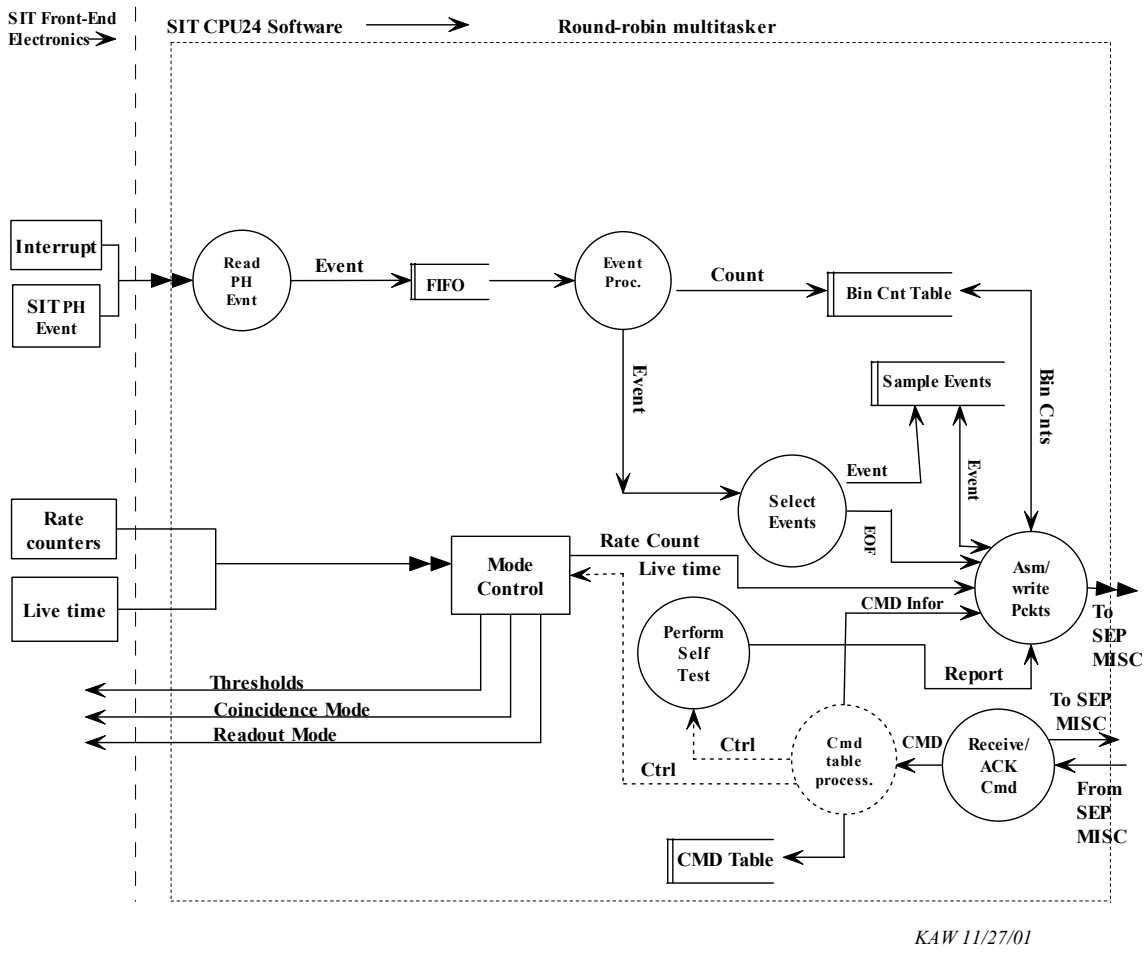

Figure 3.3 SIT Data Flow thru SIT CPU24

3.4. *Design*

Basic information flow modeling shall be used as the design methodology for the development of HET and SIT CPU24 software. A series of data flow diagrams shall be used to depict the refinement of information flow between the front-end electronics, the CPU24 and the SEP Central MISC for the HET and SIT sensors. Ward and Mellor extensions shall be used to extend the basic flow notation to accommodate a real-time system that responds to interrupts. Hatley and Pirbhai extensions shall also be used to depict behavior modeling of the system using state transition diagrams. These mapping techniques shall enable the software team to derive a conventional program structure from data flow characteristics.

3.5. *Coding*

The software development approach shall be modular and structured. HET and SIT CPU24 software modules shall be developed to accomplish science data acquisition and

processing, beacon data processing, housekeeping and status data acquisition, and science data packet formatting and command processing.

The software team at GSFC using the assembly language shall develop the HET and SIT CPU24 software modules. Algorithms for the data processing routines shall originally be developed using the C programming language. These C routines shall be tested and delivered to the software team, who shall translate them into assembly language.

The command processing, interrupt servicing and data packet formatting modules shall be originally developed in assembly language.

Good coding practices shall be used in the development of these modules, such as:

- Software modules shall be structured into subroutines according to functionality.
- Software subroutines shall be compact in size, a minimal number of instructions used to accomplish the task.
- Software subroutines shall be written to use memory, stacks, and registers efficiently.
- Source code shall be documented internally with comments.
- Software shall contain error-handling subroutines for abnormal conditions and behavior.
- Software subroutines containing a looping structure shall have well defined ending conditions.
- Interactions between modules and subroutines shall be well defined.

Source code developed by the software team shall be retained at GSFC, binary executables shall be delivered to CIT for loading into the EEPROM which is attached to the SEP Central MISC.

3.6. *Development Environment and Equipment*

The flight software for the HET and SIT CPU24 processors shall be developed using the following packages:

- Assembler (24-bit)
- CPU24 Debug Monitor
- Windows Debug Monitor Interface
- Visual C++ compiler
- Windows 98/2000 operating system
- CPU24 boot code

The flight software for the HET and SIT CPU24 processors shall be developed in a hardware environment requiring the following equipment:

- PC equipped with an Intel Processor, hard disk, available serial port, and a monitor
- CPU24 processor with serial connector

- RS232 serial cable
- Power supply
- Power supply cables

Refer to figure 3.6. for the software and hardware requirements for the development of HET and SIT CPU24 processor software.

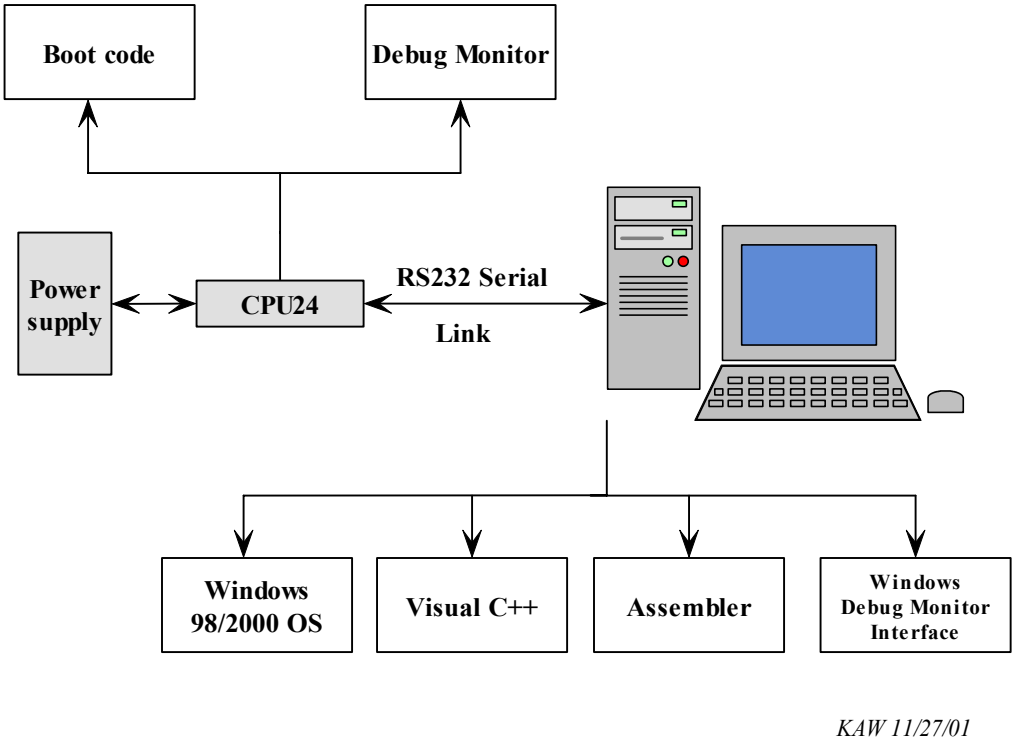

Figure 3.6 Software and hardware requirements for the development environment for the HET and SIT CPU24 processor software.

All software packages and equipment are currently available. The assembler software was developed and delivered by Phong Le and the debug monitor and Windows debug monitor interface were developed and delivered by Tom Nolan.

The debug monitor and Windows debug monitor interface shall be used for debugging and verification of the assembly code modules during development. The Windows debug monitor interface executes on a PC equipped with an Intel processor running the Windows 98/2000 operating system. This configuration requires the CPU24 processor to be attached to the PC via a RS232 serial link.

Available boot code and debug monitor software shall be loaded from the PC to the CPU24 via the serial link during development. The assembler, Visual C++ and the Windows debug monitor interface shall be executed on the PC. Assembler executables shall be loaded and executed on the CPU24 processor.

Supplemental PC's shall be used for development of the data processing algorithms in the C programming language using the Visual C++ compiler.

3.7. *Product Assurance*

3.7.1. **Walkthroughs and Reviews**

Regular in-house walkthroughs and reviews shall be conducted throughout the development cycle on all software requirements, data flow diagrams and source code modules.

During the HET and SIT software requirements phase, the software team shall meet with the HET and SIT team members for an in-house requirements review of the flight code. Project personnel will be invited to this review. Once the requirements have been reviewed and accepted, the software team shall finalize the Flight Software Requirements for the HET and SIT CPU24 Processor.

During the HET and SIT software design phase, the software team shall meet with the HET and SIT team members for an in-house design review of the flight code. Project personnel will be invited to this review. Once the design has been reviewed and accepted, the software team shall finalize the Flight Software Design for the HET and SIT CPU24 Processor.

Software shall be developed in a modular fashion. These modules shall consist of the data (event and beacon) acquisition and processing, housekeeping and status data acquisition, science data packet formatting and command processing. Internal walkthroughs shall be conducted on each of these major software components.

HET and SIT team members shall develop the data processing algorithms using the C programming language. The software team and interested team members shall review the data flow diagrams and C modules with the originator. The intent of these reviews is to ensure understanding and to verify correctness of the algorithms prior to conversion into assembly language. Once the modules have been coded in assembly language, the responsible software team member shall also conduct an informal walkthrough with interested HET and SIT team members.

The software team members shall conduct regular in-house walkthroughs for HET and SIT team members on all data flow diagrams and assembly code modules for accomplishing the command processing, data acquisition and packet formatting tasks. The intent of these reviews is to ensure understanding, correctness and efficiency of the algorithms and the source code.

Action items shall be identified from all in-house walkthroughs and peer reviews and their status shall be provided at PDR, CDR and any other project reviews.

3.7.2. **Test Plan**

Continuous software component testing shall be conducted throughout the development phase. Software modules originating in the C programming language shall be tested before delivery to the software team. Once the source code module has been developed in the assembly language, parallel testing shall be conducted to compare the results produced by the C and assembly software versions. The Visual C++ compiler and debugger shall be used to verify and validate the C source code modules. The debug monitor and the Windows debug monitor interface software shall be used to verify and validate the assembly software.

A software requirements traceability matrix shall be developed and maintained throughout the development phase to ensure all requirements are traceable to software components and are tested for acceptance. The RTM shall be made available for review by the Project Office, upon request.

3.7.3. **Software Configuration Management / Backup Plan**

A designated configuration manager (CM) shall be assigned and responsible for version control, distribution, tracking and record keeping during the HET and SIT CPU24 flight software development at GSFC.

The CM shall be responsible for the following tasks:

- Version control of all GSFC developed software components and executables (C and assembly language).
- Version control of all software (GSFC developed) and related (developed by IMPACT team) documents.
- Distribution of flight software components to the GSFC software team.
- Distribution of software and related documents to the GSFC software team and HET and SIT team members.
- Distribution of GSFC software documents to IMPACT team members and project personnel.
- Distribution of the executables and software documentation (developed by GSFC) to the lead flight engineer at Caltech.
- Ensure that all source code modules and documents are recoverable in a systematic fashion by using both removable and static media.

To automate and aid the CM in accomplishing the above tasks, the WinCVS software package is being proposed and researched. WinCVS software is a free package available at www.cvshome.org. Use of the WinCVS software shall be adequate to ensure source code and document version and change control at GSFC.

For added security, backup copies of all source code modules and documents shall be stored in several different physical locations.

3.7.4. **Software Problem Reporting and Tracking**

Each software and requirement change shall be recorded on a Software/Requirement Change Request form and submitted to the HET and SIT team members for approval. The CM shall use MS Access to store and track the software or requirement change from initiation through the development cycle.

The CM shall be responsible for the following tasks:

- Record all additions/changes to the software requirements.
- Track all additions/changes to the software requirements through the development cycle.
- Record all changes/fixes to the software components.
- Track all software component changes/fixes through the development cycle.
- Ensure all software and requirements changes are reflected in the corresponding documentation.

3.7.5. **Risk Assessment**

The software team for the HET and SIT CPU24 Processors shall consist of several members working at or near GSFC. The communication level, during and after development, between all software team members and scientists is expected to be high, therefore reducing the risk of a single point of knowledge. Any risks identified shall be communicated with the HET and SIT software and management teams.

3.7.6. **Software Maintenance**

The HET and SIT software team is expected to produce adequate documentation describing the software components' design, source code and usage. The original/ replacement programmer can use this documentation for reference, in the event the software needs to be modified.

4. Management Plan

4.1. *Build Plan*

4.1.1. **HET CPU24 Processor Software**

The HET CPU24 processor software shall consist of several major components. Each software component shall be developed and tested independently. The following major builds shall be identified as milestones in the development phase:

- The HET Data Processing Build (HDPB) shall be designed to test the HET data processing (event and beacon) components. These software components support/require the CPU24.
- The HET Data Acquisition and Command Processing Build (HDACPB) shall be designed to test the science, housekeeping and status data acquisition and command processing components. These software components support/require the HET front-end electronics and the CPU24.

- The HET Data Formatting and Command Processing Build (HDFCPB) shall be designed to test the HET science and beacon data formatting and the HET sensor and SEP Central MISC command processing. These software components support/require the SEP Central MISC, HET front-end electronics and the CPU24.
- The HET Final Integration Build (HFIB) shall be designed to test all the software components as a fully integrated and functional HET flight software system.

4.1.2. **SIT CPU24 Processor Software**

The SIT CPU24 processor software shall consist of several major components. Each software component shall be developed and tested independently. The following major builds shall be identified as milestones in the development phase:

- The SIT Data Processing Build (SDPB) shall be designed to test the SIT data processing (event and beacon) components. These software components support/require the CPU24.
- The SIT Data Acquisition and Command Processing Build (SDACPB) shall be designed to test the science, housekeeping and status data acquisition and command processing components. These software components support the SIT front-end electronics and the CPU24.
- The SIT Data Formatting and Command Processing Build (SDFCPB) shall be designed to test the SIT science and beacon data formatting and the SIT sensor and SEP Central MISC command processing. These software components support the SEP Central MISC, SIT front-end electronics and the CPU24.
- The SIT Final Integration Build (SFIB) shall be designed to test all the software components as a fully integrated and functional SIT flight software system.

4.2. ***Reviews***

The following reviews shall be conducted throughout the HET and SIT flight software development phase.

- Flight Software Requirements In-house Review for the HET and SIT CPU24 Processors
- Flight Software Design In-house Review for the HET and SIT CPU24 Processors
- Flight Software Acceptance In-house Review for HET and SIT CPU24 Processors

4.3. ***Documents and Source Code***

Version Description Documents (VDDs) shall accompany all software releases. The VDD shall contain the functionality of the build/release, list of closed software problem reports (SPRs), list of open SPRs (found during testing) , list of liens/workarounds, installation instructions, list of deliverable media or files (with version or date identifiers), the installation site-map, and any special instructions.

1. Flight Software Development Plan (Preliminary) for HET and SIT CPU24 Processors
2. Flight Software Development Plan (Final) for HET and SIT CPU24 Processors
3. CPU24 Processor Manual
4. Flight Software Requirements for the HET CPU24 Processor
5. Flight Software Requirements for the SIT CPU24 Processor
6. Software Requirements Traceability Matrix
7. Flight Software Design for the HET CPU24 Processor
8. Flight Design Document for the SIT CPU24 Processor
9. Flight Software User's Guide for the HET CPU24 Processor (updated for each build)
10. Flight Software User's Guide for the SIT CPU24 Processor (updated for each build)
11. Data Frame Format Specifications for the HET Sensor
12. Data Frame Format Specifications for the SIT Sensor
13. Flight Software Test Procedures for the HET CPU24 Processor
14. Flight Software Test Procedures for the SIT CPU24 Procesor

4.4. *Heritage and Reuse*

The GSFC group developed the flight software for the EPACT instrument suite, which was flown onboard the WIND spacecraft launched in 1994. Some algorithms and software routines developed for this project shall be reused for the HET and SIT CPU24 software.

4.5. *Staffing Plan*

GSFC /UMD – HET and SIT CPU24 Team

- Bob Baker
- Joe Dwyer
- Phong Le
- Tom Nolan
- Don Reames
- Tycho von Rosenvinge
- Peter Walpole
- Kristin Wortman

4.6. *Interaction between Caltech and GSFC*

GSFC shall develop flight software for the HET and SIT CPU24 processors. The ICD (Reference 7) shall be available to describe the communications interface between the SEP sensors and the SEP Central MISC. This should be adequate information to allow GSFC to design and build hardware and software that utilizes the interface.

Prior to SEP integration, Caltech personnel shall visit GSFC with a SEP Central

MISC test unit. This test unit shall provide the interface between the HET and SIT CPU24 processors and the SEP Central MISC. This visit shall occur after SEP Central software build 2, which is scheduled for 04/01/03.

4.7. *Schedule*

4.7.1. **Documentation**

Task Description	Start Date	End Date	Responsible Person(s)
Flight Software Dev. Plan (Preliminary), HET & SIT Processors	11/19/01	12/07/01	Baker, Reames, von Rosenvinge, Wortman
Flight Software Dev. Plan (Final), HET & SIT Processors	10/01/02	10/25/02	Reames, von Rosenvinge, Wortman
Flight Software Req. for HET CPU24 Processor	12/10/01	01/15/02	Reames, von Rosenvinge Wortman
Flight Software Req. for SIT CPU24 Proc.	12/10/01	01/15/02	Walpole
Flight Data Frame Format Spec. for HET Sensor	01/01/02	05/01/02	Reames von Rosenvinge, Wortman
Data Frame Format Spec. for SIT Sensor	01/01/02	05/01/02	Walpole
CPU24 Processor Manual	10/15/01	06/01/02	Bob Baker
Flight Software Design for HET CPU24 Processor	01/16/02	05/01/02	Nolan, Reames, von Rosenvinge, Wortman
Flight Software Design for SIT CPU24 Processor	01/16/02	05/01/02	Dwyer, Nolan, Reames, Walpole, Wortman
Flight Software User's Guide for HET CPU24 Processor	03/01/02	12/02/03*	Nolan, Wortman

Flight Software User's Guide for SIT CPU24 Processor	03/01/02	12/17/03*	Nolan, Wortman

Schedule considerations:

* Final delivery date for flight software

4.7.2. **HET CPU24 Processor Software**

Task Description	Start Date	End Date	Responsible Persons
HDPB	03/01/02	10/15/02	Reames, von Rosenvinge, Wortman
Event Data Processing	03/01/02	06/15/02	
Beacon Data Processing	09/16/02	10/15/02	
HDACPB	06/01/02	11/15/02 *	Nolan, Reames, Wortman
Command Processing	06/01/02	08/01/02	
Data Acquisition	10/02/02	11/15/02	
HDFCPB	11/16/02	04/15/03 **	Nolan, Reames, Wortman
Data Formatting	11/16/02	02/15/03	
HFIB	04/16/03	12/02/03	Nolan, Reames, von Rosenvinge, Wortman
Integration	04/16/03	12/02/03	

4.7.3. **SIT CPU24 Processor Software**

Task Description	Start Date	End Date	Responsible Persons
SDPB	06/16/02	11/15/02	Dwyer, Reames, Walpole, Wortman
Event Data Processing	06/16/02	09/15/02	
Beacon Data Processing	10/15/02	11/15/02	
SDACPB	03/01/02	11/15/02 *	Nolan, Reames, Walpole, Wortman
Command Processing	08/02/02	10/01/02	
Data Acquisition	10/02/02	11/15/02	
SDFCPB	02/16/03	04/15/03 **	Nolan, Reames, Walpole,

			Wortman
Data Formatting	02/16/03	04/15/03	
SFIB	04/16/03	12/17/03	Dwyer, Nolan, Reames, Walpole, Wortman
Integration	04/16/03	12/17/03	

Schedule considerations:
* CDR - 11/19/02-11/20/02, VLSI chip delivery - 07/11/02 (tentative)
** Caltech - SEP Central MISC visit to GSFC