Command & Data Handling Software

Steve Williams

The Johns Hopkins University

Applied Physics Laboratory

Laurel, MD 20723-6099

Tel: (301) 953-6000 x8883

Fax: (301) 953-6556

Email: Stephen.Williams@jhuapl.edu

This page intentionally left blank

Overview

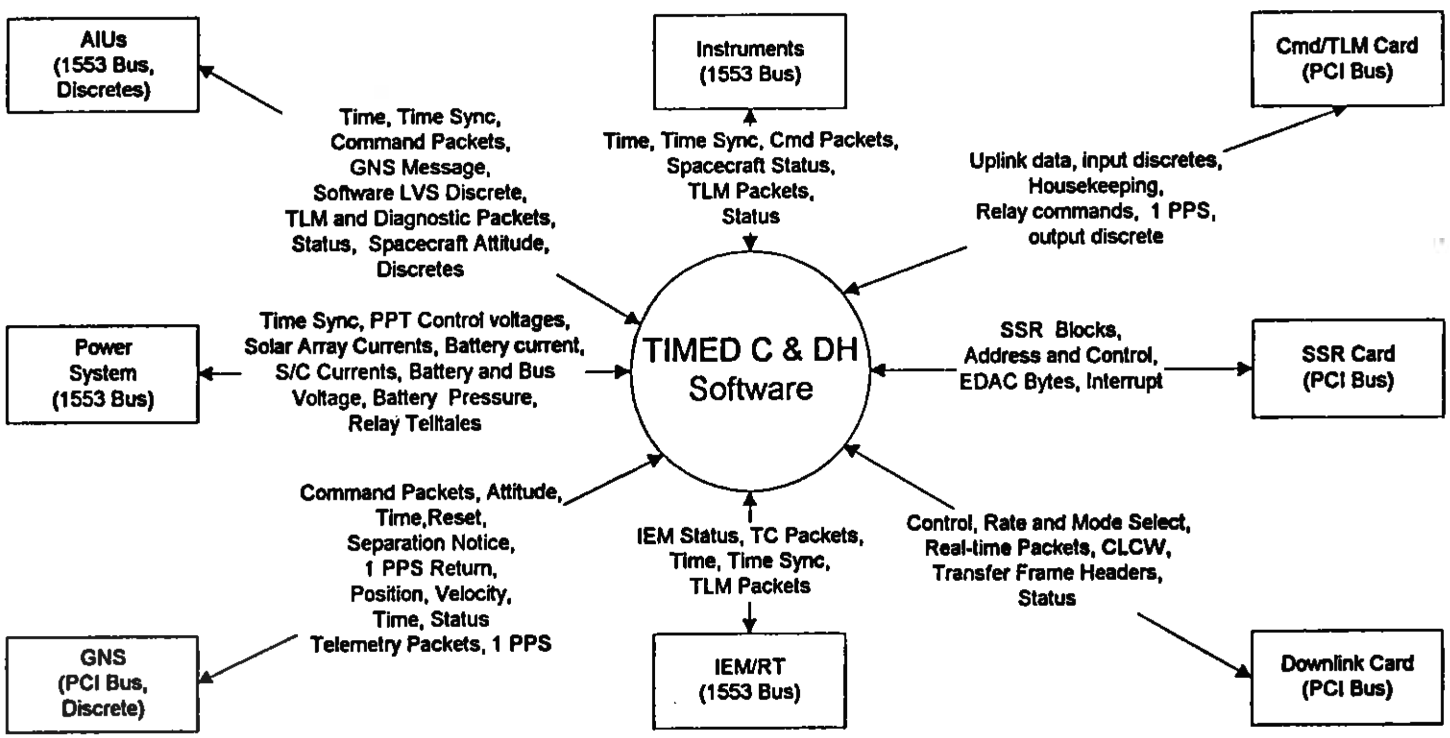
- Software Context Diagram
- Top Level Requirements
- Major Requirements Changes since PDR
- Requirements Reviews and Documents
- Development Environment
- Design Details
- Design Reviews and Documents
- IV & V

Acronyms
- IEM - Integrated Electronics Module
- PCI - Peripheral Components Interconnect

1. The primary communication path outside of the IEM is the 1553 bus. The primary communication path within the IEM is the PCI bus.

2. The spacecraft status message is sent once per second to all instruments. It includes position, velocity, attitude (roll, pitch and yaw), sun vector, data validity flags and indicators of warnings and events. Warnings and events indicators are the day/night indicator, the polar region indicator, the South Atlantic Anomaly indicator, the yaw maneuver indicator, the solar panel rotation indicator, two attitude validity indicators, the soft LVS indicator, and power down indicators.

Software Context Diagram



Acronyms

CCSDS - Consultative Committee for Space Data Systems

C & DH - Command and Data Handling

1. When both IEM's are on, one C & DH will be the 1553 bus controller, and second C & DH will be a remote terminal on the bus. A relay contact,brought into the IEMs, will be examined by the software, which will then set up the 1553 bus protocol chip appropriately.
2. The downlink card can also be a bus master on the IEM PCI bus.

Major Requirement Areas - 1

- Deliver CCSDS telecommand packets
- Execute C & DH commands
- Provide spacecraft safing and autonomy
- Be bus controller and remote terminal on the Spacecraft 1553 Bus
- Be a bus master on the IEM PCI bus
- Implement peak power tracking and coulometry algorithms

Acronyms
TLM - telemetry

1. Management of the SSR includes maintaining two distinct segments on the SSR, checking for bad blocks on the SSR and mapping around them, mapping around bad memory chips, and commanding the SSR refresh rate and power mode.

Major Requirement Areas - 2

- **Build TLM packets from instrument and sub-system status and spacecraft housekeeping data**
- **Collect TLM packets from instruments and sub-systems**
- **Record and downlink TLM packets**
- **Manage the Solid State Recorder**
- **Support uploading of new code to flash memory**
- **Support storage of data structures (autonomy rules, command macros) to flash memory**

1. The peak power tracking rate has been reduced from 40 Hz to 16 Hz.
2. Specifically, there is no longer a high speed serial bus between the two IEM's, and no cross-strapping of 1 PPS signals from the GNS card of oneIEM to the C & DH processor of the second IEM card. Also, the software will not attempt to record to the SSR in the other processor, which was a possibility at one time.

3a. A check is of the form x&M = A, x&M /= A, x&M < A, x&M > A, where x is a telemetry value, M is a mask, and A is a constant. The checks are combined into a logical expression from left to right, using AND and OR operators.

3b. N true evaluations of the rule are required in M total evaluations for the rule to trigger. A rule is evaluated once per second.

3c. The use of functional groups provides for easier management of related autonomy rules.

3d. Arithmetic checks are of the form:

A(x&M1)(y&M2) + B(x&M1) + C(y&M2) + D <, > E, where A,B,C,D and E are 32 bit floating point numbers, x and y are telemetry points, and M1 and M2 are masks. The result of an arithmetic check is a flag, which can be used in an autonomy rule. It is expected that these checks will be used to compute power dissipation, based on both current and bus voltage.

3e. Relative time-tagged rules "time-out" at a time relative to when they are loaded, as opposed to an absolute time. It is anticipated that they will be used in conjunction with autonomy rules.

3f. New autonomy rules and their associated command macros can be uploaded to RAM, then saved to flash memory so that they are present in case the C & DH processor resets.

Requirements Changes Since PDR

- Peak power tracking rate reduced
- Cross-strapping requirements between IEMs reduced
- Autonomy capabilities have been determined
  - four logical checks/per rule
  - N out of M criteria for rule firing
  - creation of functional groups
  - use of arithmetic checks
  - addition of relative time-tagged commands
  - storage of autonomy rules and macros to flash memory

This page intentionally left blank

Requirements Review and Documents

- Interface requirements review held on 04/25/97
  - action items answered in SEE-97-0094
- Application code requirements review held on 06/7/97
  - action items answered in SEE-97-0131
- Software Requirements Specification document JHU/APL 7363-9110

1. The console boot program will be disabled when the console connector is removed. The flight boot code occupies the first 256 Kbytes of the 4 Mbyte flash memory area, and it will be write-protected when the console connector is removed.

Software Development Environment

- Target - Mongoose V, 32-bit RISC processor, 1Mbyte Flash (console boot), 4 MBytes Flash, 2 Mbytes SRAM, 64Kbytes shared RAM (1553), 8 Kbytes dual-port RAM (PCI)
- Development Platform - PC, running Windows NT
- Development Boards - Turbo Rocket, uses LSI Logic 33300, plus breadboard, engineering model
- Real Time Operating System - Nucleus Plus
- Compilers - Tasking C
- Debuggers -Tasking

Acronyms
PPS - pulse per second
GNS - GPS Navigation System

1. There are 8 minor frames per 1553 bus major frame.
2. The 1 PPS from GNS is the most important timing signal. The 1 Hz and 16 Hz timers are provided by the operating system, and are synchronized to the 1 PPS. In normal operation, the 1 Hz timer is used as a backup to the 1 PPS, and is set to a period slightly longer than one second. 16 Hz was chosen for a timer frequency because it is a multiple of the bus minor frame rate, and higher resolution than 8 Hz was needed.
3. Modules that manage the 1553 bus are among those that execute at specific counts of the 16 Hz timer.
4. Receipt of uplink data readies the various modules that form and process CCSDS telecommand packets. Receipt of telemetry packets readies the modules that store telemetry to the SSR or downlink it in real time.
5. Peak power tracking and the inputting of an uplink code block must each complete in roughly 30 msec. Modules that manage the 1553 bus must complete in roughly 62 msec.

Functional Design - Top Level

- Basic software "cycle" is 1 Hz (1553 major frame)
- 1 PPS, 1 Hz, and 16 Hz timers provide timing
- Many software modules execute after specified interrupts of the 16 Hz timer
- Many modules are driven by events
  - receipt of uplink data, receipt of telemetry packets
- Certain modules must complete within a specified period of time
  - peak power tracking, reading of uplink data from the hardware, 1553 bus management

This page intentionally left blank

Software Modules - Bus Controller

Module Name	Description, Comments
1 PPS ISR	Responds to 1 PPS; used to synchronize timing in normal operations; used to start 1553 major frame; activates several modules
1 Hz Timer ISR	Responds to internal 1 Hz Timer; used as a backup to the 1 PPS
16 Hz Timer ISR	Responds to internal 16 Hz Timer; readies many tasks
Power System Timer ISR	Responds to internal timer that is started in the 16 Hz Timer ISR; readies Power System task
SSR ISR	Responds to SSR interrupt; readies task SSR Playback Continue
Command Buffer Ready ISR	Responds to command buffer ready interrupt; readies Uplink Data task
Relay Command Complete ISR	Responds to relay command complete interrupt; clears the busy flag associated with the relay command interface
PCI Transaction Complete ISR	Responds to interrupt from PCI controller chip; sets a Nucleus Plus event flag to allow PCI processing to continue
Correctable Error ISR	Responds to an interrupt from the Mongoose EDAC circuitry indicating that a single bit error has occurred; writes back corrected value to memory, and verifies that corrected value can be read back OK
Perform Peak Power Tracking	Does peak power tracking; must execute within 30 msec.
Perform Coulometry	Does battery coulometry
Build 1553 Minor Frames	Builds Summit command blocks and outputs them to 1553 memory; Must complete within 62 msec.
Process 1553 Minor Frame Results	Moves data collected from 1553 memory to internal memory; in worst case, must complete within 62 msec.

This page intentionally left blank

Software Modules - Bus Controller

Manage GNS Interface	Performs the PCI-bus transactions that transfers data to and from GNS.In worst case, must complete within 62 msec.
Output 1 PPS Messages	Sends the 1 PPS return message to GNS and the 1 PPS indicator to the CMD/TLM card
Output Downlink Headers	Reads the hardware BC/RT telltale, unmasks the 1 PPS interrupt and outputs downlink transfer frame header information and CLCW to the downlink card
Parse Telemetry Packets	Writes TLM packets to the SSR TLM Packet Buffer if recording is enabled, and writes TLM packets to the Real-timeTLM Packet Buffer if a real-time downlink is enabled. Maintains knowledge of instrument packet counts and packet limits
Manage Real-time Downlink	Verifies that the real-time telemetry packets are consistent with the selected downlink mode, and writes groups of four telemetry packets to the downlink hardware
Manage Segment 1 Recording	Forms SSR data blocks from telemetry packets that will go on SSR segment 1, and writes SSR data blocks to SSR segment 1
Manage Segment 2 Recording	Forms SSR data blocks from telemetry packets that will go on SSR segment 2, and writes SSR data blocks to SSR segment 2
Start SSR Playback	Responds to the execution of a Downlink Format Select command that specifies an SSR playback. Outputs the next location to read from and the number of blocks to read; must have knowledge of bad blocks and bad chips.
Continue SSR Playback	Responds to an interrupt from the SSR hardware that occurs when the specified number of blocks have been read out. Outputs the next location to read from and the number of blocks to read; must have knowledge of bad blocks and bad chips.
Input Housekeeping Data	Reads housekeeping data from the Command and Telemetry card, and stores it to the Data Collection Buffer.
Form C & DH Telemetry Packets	Forms packets from internal C & DH variables and structures.

This page intentionally left blank

Software Modules - Bus Controller

Form Housekeeping Telemetry Packets	Forms packets from instrument and sub-system status, spacecraft currents, voltages and temperatures
Update Data Summary Table	Update the data summary table with the latest housekeeping data.
Evaluate Arithmetic Functions	Evaluates up to 64 arithmetic functions that are part of the autonomy system; these must be completed before evaluating the autonomy rules
Evaluate Autonomy Rules	Evaluates up to 512 autonomy rules.
Evaluate Time-tagged Rules	Evaluates up to 512 time-tagged rules
Process and Execute C & DH Commands	Executes the highest priority C & DH command that is ready, and updates command pointers
Check C & DH Commands	Checks the validity of all commands in a telecommand packet that has been addressed to the C & DH
Check for TC Packet Time-out	Checks to see if any partially completed packets built using segmentation have exceeded their time-out period
Perform TC Packetization Layer Checks	Check the validity of completely assembled TC packets.
Perform TC Segmentation Layer Checks	Assembles and extracts TC packets from segments.
Perform TC Transfer Layer Checks	Performs the specified CCSDS transfer layer checks on a TC transfer frame, and updates the CLCW
Perform TC Coding Layer Checks	Computes the BCH code for a CCSDS code block, and compares to the received data, then de-randomizes the data. Sends a completed transfer frame to the next higher layer.
Input Code Block	Reads a CCSDS code block from the Command and Telemetry card.
Check SSR Memory	Reads SSR data blocks from the SSR, and checks for uncorrectable errors.
Scrub C & DH Memory	Reads from C & DH RAM to correct any single bit errors.

1. Interrupts will not be nested. Interrupt service routines (ISRs) will be kept very short. Many ISR's will clear the interrupt, then set an event flag, which will have the effect of causing a task to be made ready. Measured interrrupt latency is 40 usec.
2. The Mongoose has interrupts numbered 0-5, with 0 being the highest priority. Interrupt 5 Is used for expansion, with up to 10 external interrupts, and several internal interrupts, being tied to interrupt 5. The internal timers use interrupts 0 and 1, and the floating point unit uses interrupt 3. The 1 PPS is tied to interrupt 2, and the 1553 interrupt is tied to interrupt 4. All other interrupts use interrupt 5.
3. Interrupt rates are:
    - 1 PPS - 1/second
    - 1 Hz - 0 in normal operation, 1/second worst case
    - 16 Hz - 15/second (timer disabled after interrupt 15, re-enabled after 1 PPS)
    - Power System - 16/second
    - Command buffer ready - 31/second, maximum
    - SSR Playback - 1 every 16 seconds, playback only, nominal operation
    - Relay Command Complete - 9/second, maximum
    - PCI Transaction Complete - 50/second (1 for each PCI bus transaction)
    - Correctable Error - 1.3/second (based on peak error rate due to protons in the South Atlantic Anomaly)

Functional Design - Interrupts

- 1 PPS (from GNS)
- Processor/Operating system timers (1 Hz, 16 Hz, Power system)
- Command buffer ready
- SSR playback
- Relay command complete
- PCI transaction complete
- 1553 bus (remote terminal only)
- Correctable error (single-bit error on memory transactions)

1. A pre-emptive, multi-tasking operating system (Nucleus Plus) is used. Modules that can execute at the same time are placed in the same task. Otherwise, most modules are placed in separate tasks. The goal is to keep the tasks simple, and let the operating system do the work.
2. The remote terminal software has 27 modules in 15 tasks. Five modules and two ISRs are different in the remote terminal than in the bus controller.
3. This is estimate of software loading includes the overhead associated with the operating system.
4. The Power System task contains the module to do peak power tracking. The Uplink Data task contains the module to input uplink code blocks. The 16 Hz task contains modules that manage the 1553 bus.
5. The priority of the SSR Playback task will be raised once a playback has started.
6. The priorities of the 1 Hz task and the Autonomy task will be raised if they have not executed by certain times within a one second interval. Both contain software modules that must complete within one second.
7. The 1 Hz, Autonomy and Command Execute tasks contain modules that may take a relatively long time to execute, and thus may be time-sliced, I.e., set up to run for a certain period of time, and then suspending, to allow other tasks of equal priority to run.
8. This is a very typical architecture for real-time systems. All tasks are set up with the first statement inside of the loop a request for a Nucleus Plus service. If the service can't be satisfied, the task will suspend. When the service is satisfied, the task is placed in a ready state. The operating system executes the highest priority ready task. Typical service requests that the tasks will require are an event flag being set, or data being present in a Nucleus Plus data structure such as a queue, pipe or mailbox.

Functional Design - Tasks

- Bus controller has 30 modules organized into 16 Nucleus Plus tasks
- Analysis indicates the processor is lightly loaded (~20 %)
- 3 "hard" real-time tasks given highest priority
- Remaining tasks given equal priority
  - priority of one task will be dynamically raised, and priorities of two others may be raised
- Time slicing of longer tasks will be done
- All tasks run as infinite loops

This page intentionally left blank

Task Priorities - Bus Controller

Task Name	Priority (0 = highest priority)	Software Modules Included
Power System	0	Perform Peak Power Tracking, Perform Coulometry
Uplink Data	0	Input Code Blocks
16 Hz	2	Build 1553 Minor Frames, Process 1553 Minor Frame Results, Manage GNS Interface, Check SSR Memory, Scrub C & DH Memory, Output 1 PPS Messages, Output Downlink Headers
SSR Playback	7/3	Start SSR Playback, Continue SSR Playback
Telemetry Packet	7	Parse Telemetry Packets
Segment 1 Recording	7	Manage Segment 1 Recording
Segment 2 Recording	7	Manage Segment 2 Recording
Real-Time Downlink	7	Manage Real-time Downlink
Command Execute	7	Process and Execute C & DH Commands
Command Check	7	Check C & DH Commands
Packetization Layer	7	Perform Telecommand Packetization Layer Checks
Segmentation Layer	7	Perform Telecommand Segmentation Layer Checks
Transfer Layer	7	Perform Telecommand Transfer Layer Checks
Coding Layer	7	Perform Telecommand Coding Layer Checks
1 Hz	7/4	Check for Telecommand Packet Time-out, Form C & DH Telemetry Packets, Evaluate Time-tagged Rules
Autonomy	7/4	Input Housekeeping Data, Evaluate Arithmetic Functions, Evaluate Autonomy Rules, Form Housekeeping Telemetry Packets, Update Data Summary Table

Acronyms

COP - Command Operation Procedures

1. COP-1 insures an orderly delivery of CCSDS transfer frames. The received frame sequence number must match the expected value. Frame error control consists of a16-bit CRC computed over the entire transfer frame.
2. All C & DH commands have a 32-bit checksum. When the commands are received, the checksum and the command construction is checked. Before a command is executed, its checksum is again checked.
3. The software will scrub all of SRAM in 16 seconds. Scrubbing corrects correctable (single bit) errors before they become uncorrectable.
4. The software reads back SSR blocks. It checks the bytes generated by the EDAC-V chip to see if there are uncorrectable errors. It reports bad blocks in telemetry, and maps around them.
5. Autonomy and time-tagged rules are stored with their checksums, which are checked each time the rules or evaluated. Autonomy rules and command macros must first be disabled before they can be cleared or over-written. Memory management for command macros is done on board, minimizing the chances that the ground can inadvertently over-write a macro.
6. Several error history buffers are maintained to store information about errors found in the building or delivery of telecommand packets, the checking and execution of C & DH commands, and problems on the PCI and 1553 buses. The top entries from all of these buffers are placed in telemetry packets that are always produced. The entire buffers are available via memory dumps.
7. The resetting of the software watchdog timer will only occur if all regularly scheduled tasks run.

Functional Design - Fault Protection

- COP-1 protocol, frame error control on uplink
- Checksum, format checking on commands
- EDAC, memory scrubbing on SRAM
- Checking of SSR memory for uncorrectable errors
- One retry on PCI transactions
- One to three retries on 1553 bus messages
- Checksums, write protection on autonomy rules and command macros
- Extensive fault reporting in telemetry
- Software watchdog timer

TIMED

Thermosphere • Ionosphere • Mesosphere • Energetics and Dynamics

1. Software build 1 will support IEM testing in the summer of 1998. It will implement most of the basic functions: building and delivering telecommand packets, executing a subset of C & DH commands, collecting telemetry packets, building some of the C & DH telemetry packets, supporting the real-time downlink, and performing the peak power tracking algorithm.

2. Software build 2 will support initial engineering model IEM testing on the spacecraft in the fall of 1998. It will add support for most of the solid state recorder functions, implement additional C & DH commands, build additional C & DH telemetry packets, and perform the coulometry algorithm.

3. Software build 3 is the final build, and is scheduled to be completed in the spring of 1999. It adds support for autonomy and time-tagged rules, implements the remaining C & DH commands, builds the remaining C & DH telemetry packets, adds the remote terminal unique software, and adds checking and re-mapping of solid state recorder memory.

Design Reviews and Documents

- Software PDR held on 07/27/97
  - action items answered in SEE-97-0197
- Functional Design Document JHU/APL 7363-9113
- Detailed design reviews, code walk-throughs will be held for each of 3 builds
- 1553 Bus Specification JHU/APL 7363-9111
- PCI Bus Specification JHU/APL 7363-9112
- C & DH Command Definitions JHU/APL 7363-9115
- C & DH telemetry packet definitions

This page intentionally left blank.

IV & V

- **IV & V will be done internal to APL**
- **Individuals not on the S/W development team have primary responsibility for S/W qualification**
  - lead, with support from the mission ops team
- **IV & V effort will culminate in a software acceptance test on the final software build**
  - will consist of a lengthy series of scripts executing on the IEM testbed and mini-MOC.
- **A detailed list of tests is being developed by the IV & V lead; all tests are referenced to requirements**