SOFTWARE REQUIREMENTS SPECIFICATION / INTERFACE REQUIREMENTS SPECIFICATION

删除[Alex Edsall]: 10 August 2001

for the

X-38 Fault Tolerant System Services

Contract No. NAS 9-97216

DRL Sequence Nos. 12/14

12 April 2000

Prepared for:

National Aeronautics and Space Administration
Lyndon B. Johnson Space Center
2101 NASA Road 1
Houston, Texas 77058-3696

Prepared by:

The Charles Stark Draper Laboratory, Inc.
555 Technology Square
Cambridge, Massachusetts 02139
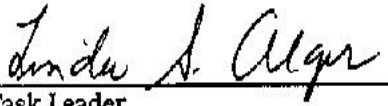**Cage Code: 51993**

DISTRIBUTION STATEMENT [A]

[Approved for public release; distribution is unlimited]

Total pages: 97

SOFTWARE REQUIREMENTS SPECIFICATION / INTERFACE REQUIREMENTS SPECIFICATION

删除[Alex Edsall]: 10 August 2001

for the

X-38 Fault Tolerant System Services

Approved by:
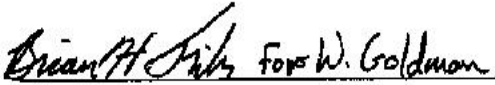
_____   _5/4/2000_
Task Leader   Date
Linda S. Alger

_____   _5/4/2000_
Engineering Director/Associate Director   Date
Linda S. Alger

_____   _5/4/2000_
Quality Assurance   Date
Wade M. Goldman

_____   _5 MAY 00_
Technical Director   Date
Alexander C. Edsall

_____   _5/5/00_
Program Manager   Date
Roger E. Medeiros

_____   _____
Sponsor   Date

删除[Alex Edsall]: 10 August 2001

RECORD OF REVISIONS

Rev	Result of	Pages Affected	Approval/Date
–	ECR 0079A	Initial Release	L.S.A. 5/4/00
A	ECR 0112	Revision due to Updated FTTP Specifications	RR 24 Aug 2000
B	ECR 134	Revision due to Updated FTTP Specifications	RR 22 Dec 2000
C	ECR148	Revision due to NASA Comments and Updated FTPP Requirements Document	RR 14 Mar 2001
D	ECR182	Revision due to NASA Comments and Updated FTPP Requirements Document	RR 2 Jul 2001
E	ECR190	Revision due to NASA Comments	RR 10 Aug 2001
F	ECR0226	Updated Requirements Traceability table, pages 65, 67, 68, 74, 83, 84, 86	

TABLE OF CONTENTS

Figure   **Page**

删除[Alex Edsall]: 10 August 2001

LIST OF TABLES

Table   **Page**

1. SCOPE

1.1 Identification

This Software Requirements Specification/Interface Requirements Specification (SRS/IRS), Draper document number 297749, defines the software requirements and the external interface requirements for the Fault Tolerant System Services (FTSS) Computer Software Configuration Item (CSCI).

1.2 System Overview

The central part of the avionics architecture of NASA's X-38 Crew Return Vehicle is a quad-redundant Flight Critical Computer (FCC) which is based on Draper's Fault Tolerant Parallel Processor (FTPP) architecture. The FCC consists of four Flight Critical Processors (FCPs) operating as a quad-redundant Virtual Group (VG), five simplex Instrument Control Processors (ICPs) running as five separate VGs, five Draper Network Elements (NEs), four Multi-protocol/RS-422-cards, sixteen Digital I/O (DIO) cards, four Analog I/O cards, and four Decomm cards.

The FCPs, operating as a single, quad-redundant set, function as the main application processor. A complete suite of Fault Tolerant System Services (FTSS) software will be loaded onto the FCPs and provide an Application Programming Interface (API) between NASA's application code and the underlying hardware (Motorola Power PCs) and a COTS operating system (VxWorks). The FTSS software provides Scheduling Services, Communication Services, Time Services, Memory Management Services, Fault Detection and Isolation, Redundancy Management, System Support Services, and a Mission Management template. A reduced set of FTSS Communications Services will be loaded onto each ICP and will provide an API between the I/O software running on the ICPs and the NEs.

Figure 1-1 is a high-level block diagram of the FCC virtual hardware configuration.

Figure 1-2 is a high-level block diagram of the FCC software architecture.

删除[Alex Edsall]: 10 August 2001
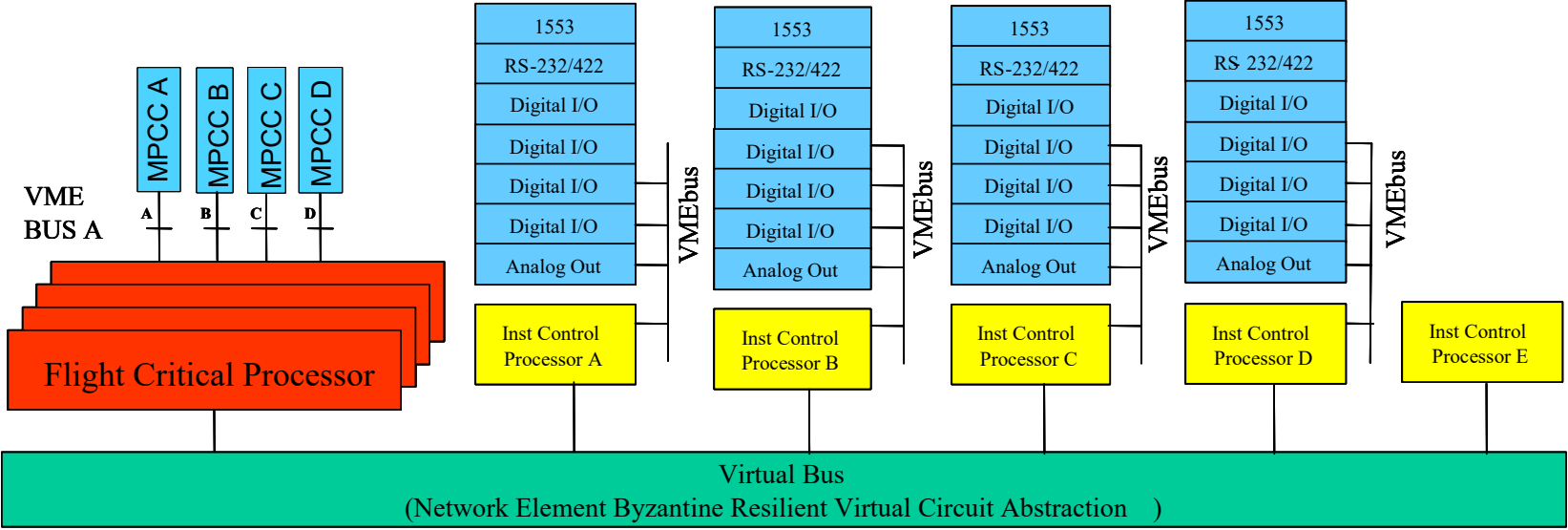
Figure 1‑1 FCC Virtual Architecture.

删除[Alex Edsall]: 10 August 2001

FCP

ICP

Applications

Sensor, Appl, Effector, Vehicle FDI & RM	User Applications Tasks	Mission / Vehicle Management

API →

FTSS

Scheduling Services	Memory Mgmt. Services	Communications Services	Support Services	Time Services	FDIR (FCP, NE)	MPCC

User I/O Tasks

FTSS SW (partial)	ICP Custom System Software

OS

VxWorks		BRVC

Hardware

FCP CPU and Resources	FCP I/O Hardware	

BRVC	ICP CPU & Resources	VxWorks / ICP I/O Hardware

Legend:
- ■ User Software
- ■ Fault-Tolerant System Services
- ■ COTS Software
- ■ FCP Hardware
- ■ ICP Hardware

Figure 1‑2. FCC Software Architecture.

1.3 Document Overview

This specification defines the software requirements and the interface requirements for the FTSS CSCI. It has been prepared using MIL-STD-498 and DI-IPSC-81433 and DI-IPSC-81434 for guidance. This SRS/IRS is organized as follows:

1. Section 1 - Scope: identifies the CSCI that this specification pertains to, provides an overview of FTSS, and provides an overview of this specification.

2. Section 2 - Referenced Documents: provides a list of documents referenced in this specification.

3. Section 3 - Requirements: specifies the engineering requirements for the FTSS CSCI

   a)   Section 3.1 describes the CSCI required states and modes.

   b)   Section 3.2 specifies the CSCI software requirements for each capability as follows:

         i)   3.2.1 System Initialization

         ii)   3.2.2 Scheduling Services

         iii)   3.2.3 Memory Management Services

         iv)   3.2.4 Communication Services

         v)   3.2.5 Fault Detection and Isolation

         vi)   3.2.6 Redundancy Management

         vii)   3.2.7 Time Services

         viii)   3.2.8 System Support Services

   c)   Section 3.3 describes the CSCI external interface requirements.

   d)   Section 3.4 identifies internal interface requirements.

   e)   Section 3.5 identifies internal data requirements.

   f)   Section 3.6 identifies the adaptation requirements.

   g)   Section 3.7 presents safety requirements.

   h)   Section 3.8 presents security and privacy requirements.

   i)   Section 3.9 discusses environment requirements.

   j)   Section 3.10 identifies computer resource requirements.

k) Section 3.11 describes software quality factors.

l) Section 3.12 identifies design and implementation constraints.

m) Section 3.13 identifies personnel requirements.

n) Section 3.14 identifies training-related requirements.

o) Section 3.15 identifies logistics-related requirements.

p) Section 3.16 identifies other requirements.

q) Section 3.17 presents packaging requirements.

r) Section 3.18 identifies precedence and criticality requirements.

4. Section 4 - Qualification provisions: defines a set of qualification methods and specifies for each requirement in Section 3 the method(s) to be used to ensure that the requirement has been met.

5. Section 5 - Requirements Traceability: provides a summary of traceability between system requirements expressed in the X-38 Fault Tolerant Parallel Processor Requirements document and the requirements elaborated in Section 3 of this document.

6. Section 6 - Notes: provides a list of acronyms and a glossary of terms used throughout this document.

2. REFERENCED DOCUMENTS

The following documents of the exact issue shown, or current issue if not shown, form a part of this specification to the extent specified herein. This document is directly traceable to the X-38 Fault Tolerant Parallel Processor Requirements document. In the event of conflict between that document and the contents of this specification, Draper will propose resolution of the conflict to NASA for approval.

2.1 Government Documents

Document No.	Date	Title
MIL-STD-498	5 December 1994	Software Development and Documentation
DID DI-IPSC-81433	5 December 1996	Data Item Description - Software Requirements Specification.
DID DI-IPSC-81434	5 December 1996	Data Item Description - Interface Requirements Specification.
JSC 28671	13 April 2001	X-38 Fault Tolerant Parallel Processor Requirements, Rev 6.2, National Aeronautics and Space Administration Lyndon B. Johnson Space Center 2101 NASA Road 1 Houston, Texas 77058-3696

2.2 Non-Government Documents

Document No.	Date	Title
297752		Application Programming Interface, The Charles Stark Draper Laboratory, Cambridge, Massachusetts
297746		Certification Test Procedure for the Network Element for the NASA X-38 Flight Critical Computer, The Charles Stark Draper Laboratory, Cambridge, Massachusetts
Publication No. YD681MPCC1	October 1998	MPCC01 Firmware Manual Rev A Radstone Technology PLC
Publication No. HH681MPCC1	October 1998	MPCC01 Hardware Manual Rev B Radstone Technology PLC
DOC-12068-ZD-00	4 Apr. 1999	VxWorks Reference Manual, 5.4 Edition 1 Wind River Systems, Inc.

All references to API in this document refer to Draper document number 297752, Application Programming Interface for the X-38 Fault Tolerant System Services.

3. REQUIREMENTS

3.1 Required States and Modes

Fault Tolerant System Services CSCI states are shown in Figure 3-1.

Power-on
Reset

Reset

System
Initialization

Timer
Interrupt
Enabled

Normal
Operation

Fault
Detected

Fault
Recovery
Complete
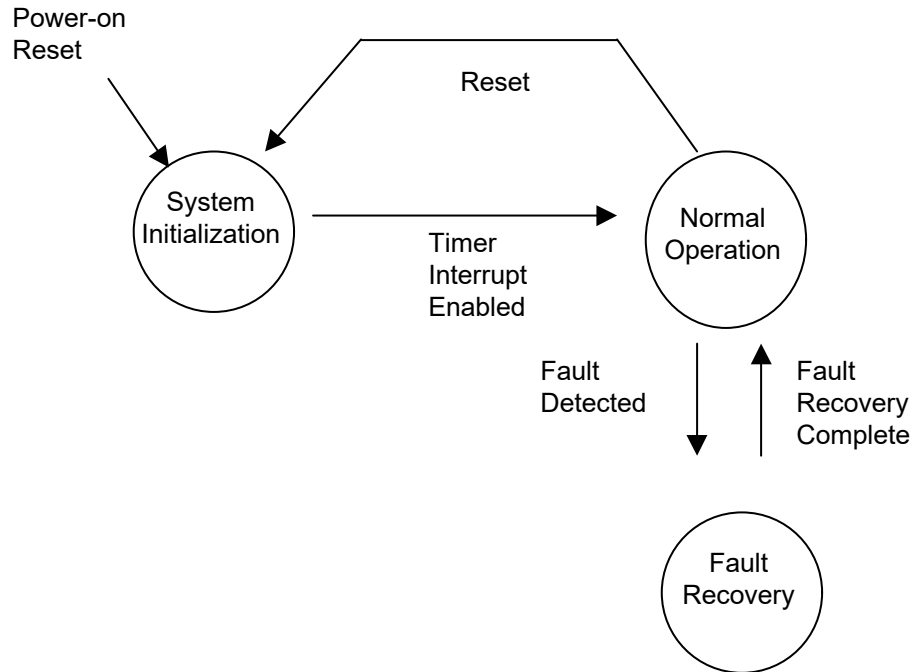
Fault
Recovery

Figure 3- 1. Fault Tolerant System Services States.

System Initialization is entered when the system is powered up for the first time, or when a power-on reset exception is received by the software. Section 3.2.1 gives the requirements for this state. The system transfers to the Normal Operation state after the FCP has been configured into a fault-tolerant computer and enables the timer interrupt.

In the Normal Operation state the software meets the performance and functional requirements (other than those listed as System Initialization requirements) in the no-fault case. The system will transfer to the System Initialization state if a reset exception is received. The system will transfer to the Fault Recovery state if a fault is detected.

In the Fault Recovery state the system is reconfigured. If a single permanent fault has occurred, for example, the system will, when the transfer is made back to Normal Operation state, be capable of handling another fault. The requirements for this state are found in Section 3.2.6.2 and its subsections.

3.2 CSCI Capability Requirements

3.2.1 System Initialization

System Initialization performs those functions necessary to transform the hardware consisting of the FCP processors, network elements, and on-board I/O devices into a real time system executing tasks with fault tolerant message exchanges.

1. Whenever a power-on reset occurs, System Initialization shall [SRS194] perform the following functions.

2. As part of System Initialization , the Boot ROM shall [SRS234] be configured to, after completing IBIT, call the manufacturer-supplied VxWorks Board Support Package (BSP) initialization software followed by a call to the FTSS System Initialization software.

3. System Initialization shall [SRS014] initiate the watchdog timer.

4. System Initialization shall [SRS292] enable and reset the processor's watchdog timer such that, in the absence of a fault, the watchdog timer does not expire and reset the processor..

5. System Initialization shall [SRS008] synchronize the FCP virtual group in the presence of a power on skew of 2.5 seconds.

6. System Initialization shall [SRS010] configure the FCP virtual group to use all available synchronized processors, if at least 3 of the 5 FCRs are active.

7. If any of the FCP processors are not synchronized, System Initialization in the surviving triplex shall [SRS177] attempt to sync with the failed FCP.

8. If the failed FCP processor has not synced in 2.5 seconds after the surviving triplex has detected the loss of the FCP, then the surviving triplex shall [SRS178], within 1 second, send a single voted VMEbus reset through the NE to the failed FCP.

9. System Initialization shall [SRS011] align processor state and congruent aligned memory locations. Processor state includes all registers. It also includes those timers used by FTSS.

10. The FCP shall [SRS296] configure ICP simplex virtual groups for each channel in the FCP virtual group.

11. The FCP shall [SRS297] wait up to 15 seconds, after configuring the ICP virtual groups, for communication to start from the ICP. The application can use this time on the ICP to initialize I/O boards.

12. System Initialization shall [SRS215] call an application initialization function to allow the application to (at least) create tasks, create communication sockets, initialize the vehicle mode, and initialize memory alignment allowance.

13. The FCP shall [SRS221], after application initialization is complete, send an FCP Ready Sync message to the ICP

14. The FCP shall [SRS189] wait up to 2.5 seconds (from the sending of the FCP Ready Sync) for the ICP Ready signal. Note that FTSS will not fail the FCR if this signal is not received within this time. FTSS will wait until the normal ICP presence test fails.

15. The FCP shall [SRS243], if the NEFU ICP fails to send its ICP Ready signal, mask out that ICP, but continue to use the NE.

16. System Initialization shall [SRS199], when all other activities are completed, start the 50 Hz timer and enable the timer interrupt. This will allow the interrupt handler to initiate normal activities.

17. System Initialization, from hardware reset to starting of the 50 Hz timer, shall [SRS015] take no longer than 1.5 minutes.

3.2.2 Scheduling Services

3.2.2.1 Scheduling Execution

Whenever the 50 Hz timer interrupt occurs, the interrupt handler invokes the scheduler (there are various ways to implement this invocation, such as using a procedure call or by setting an event; no specific implementation is to be inferred). The scheduler allows the application to create lists of tasks that run during a given segment of time, at various rates. The application can create "vehicle modes" to designate a unique segment. The application can also set up "rate groups". Each rate group has some number of tasks associated with it, and it also has a rate for those tasks. Note that there may be some number of rate groups that have the same rate. These contain the tasks that will run at that rate in different vehicle modes. Some number of rate groups can be associated with a given vehicle mode. When an API call is made to change the vehicle mode, the scheduler will disable the tasks associated with all the rate groups in the old vehicle mode, and enable the tasks associated with all the rate groups in the new vehicle mode. The enabled tasks are then unblocked at the rate given in its associated rate group. An API call is available for the task to call to block itself when it is finished with its cyclic processing.

1. The scheduler shall [SRS017] provide an API call to install a task into a rate group. The API call is invoked during system initialization.

2. The scheduler shall [SRS196] support up to 20 tasks per rate group.

3. The scheduler shall [SRS018] provide an API call to install a rate group into a vehicle mode at system initialization.

4. The scheduler shall [SRS197] support up to 3 rate groups per vehicle mode.

5. The scheduler shall [SRS195] support up to 5 vehicle modes.

6. The FTSS software shall [SRS002] provide the identical services in all vehicle modes.

7. The scheduler shall [SRS019] provide an API call for an FCP application task to alert the scheduler of a vehicle mode change.

8. The scheduler shall [SRS020] complete the change from one vehicle mode to the next within 1.02 seconds. There is up to a full major frame from notification of an impending mode change to acting on it in minor frame 0 of the next major frame plus the time it takes during the next minor frame 0 to switch tasking.

9. The scheduler shall [SRS021] process vehicle mode changes during minor frame 49.

10. The scheduler shall [SRS022] execute cyclic tasks, providing an API call to allow the application to block until its next iteration.

11. The scheduler shall [SRS024] execute as the highest priority FTSS or application task in the system.

12. The scheduler shall [SRS025] keep a minor frame count from 0 to 49.

13. The scheduler shall [SRS027] give tasks priority values according to their rate - the higher the rate, the higher the priority.

14. The scheduler shall [SRS028] detect 50 Hz, 10 Hz and 1 Hz rate group over-runs.

15. The scheduler shall [SRS029] report rate group over-runs to the application via an API service for incorporation in the telemetry data stream.

16. The scheduler shall [SRS216] provide an API call to specify which task was running within the rate group which over-ran.

17. The scheduler shall [SRS030] provide a mechanism to inform a task when it did not complete during the previous frame and restart it at the beginning of the task.

18. The scheduler shall [SRS181] set the 50 Hz interval timer to a count down value so as to cause the next minor frame interrupt at 20 msec from the previous interrupt congruently in all operational FCPs.

19. The scheduler shall [SRS032] issue a 50 Hz interrupt to the ICPs by means of a VMEbus IRQ5 interrupt.

20. The scheduler shall [SRS191] issue the 50 Hz interrupt to all the ICPs with a skew no greater than 330 microseconds.

21. The scheduler shall [SRS033] send the minor frame number, vehicle mode, mission elapsed time (MET), and separation elapsed time (SEP) to the ICP prior to the 50 Hz interrupt. Note: The NE unique identifier (NE ID) is available to the ICPs via the ftss_my_icp() API call.

22. The scheduler shall [SRS034] take no longer than 1 millisecond to execute scheduler and Time Services FTSS overhead tasks in each rate group. This means that the time

from the 50 Hz timer interrupt to the start of the first task in the 50 Hz rate group will be less than or equal to 1 millisecond, assuming 27 packets of data need to be delivered.

23. The FTSS software shall [SRS278] provide an API call that provides the application program the minor frame number.

The behavior of synchronous tasks executed by the scheduler must be deterministic.

3.2.2.2 Task and Rate Group Execution

1. The scheduler shall [SRS035] provide rate groups that execute at 50 Hz, 10 Hz and 1 Hz., with a drift rate no greater than 50 microseconds per second, and with a jitter no greater than 330 microseconds.

2. The scheduler shall [SRS037] provide a method to schedule tasks at a set rate and in a set order within the rate group.

3. The scheduler shall [SRS198] execute all the tasks in each of the rate groups that have been installed in the current mode.

4. The scheduler shall [SRS039] rely on the order used in adding tasks to a rate group to determine the task priorities.

5. The scheduler shall [SRS042] provide a method for a task to be scheduled as a 50 Hz "helper" task for source congruency input exchanges and voted output exchanges that starts in a particular minor frame but runs only during every 5th or 50th minor frame, effectively running at a lower, sub-rate, 10 Hz or 1 Hz, respectively.

6. The scheduler shall [SRS270] provide a task deadline capability that allows the application to specify which minor frame a task should start in and finish in.

All tasks in rate groups and their corresponding schedules for all vehicle modes will be setup at system initialization.

Tasks in a rate group must suspend on a scheduler API call at the top of their execution loop.

3.2.2.3 Exception Handling

For purposes of handling exceptions, exceptions are defined as either software or hardware exceptions. Software exceptions are defined as those mapped into VxWorks signals. All other exceptions are classified as hardware exceptions.

Table 3.2-1 shows the mapping of software exceptions to VxWorks signals.

Table 3.2-1. Software Exception Mapping Table.

SIGNAL	CODE	EXCEPTION
SIGBUS	_EXC_OFF_MACH	Machine check

SIGBUS	_EXC_OFF_INST	Instruction access
SIGBUS	_EXC_OFF_ALIGN	Alignment
SIGILL	_EXC_OFF_PROG	Program
SIGBUS	_EXC_OFF_DATA	Data access
SIGFPE	_EXC_OFF_FPU	Floating point unavailable
SIGTRAP	_EXC_OFF_INST_BRK	Instruction breakpoint
SIGTRAP	_EXC_OFF_TRACE	Trace
SIGILL	_EXC_OFF_SYSCALL	System call

1. Upon the occurrence of an exception of either kind (hardware or software), the FCP shall [SRS172] make the error type available to the application, via an API service, for incorporation in the telemetry stream and include all context data relevant to the exception, namely the contents of the Machine State Register (MSR), and the machine status Save/Restore Registers (SRR0 & SRR1).

2. The scheduler shall [SRS031] provide a mechanism for a task optionally to define a user written software-exception-handling routine that runs in the context of the task.

3. For hardware exceptions and reserved exceptions, the FTSS shall [SRS276] make the error type and its context data available to the application, then return from the exception handler to the task that was running when the exception occurred.

4. For software exceptions occurring within the FTSS, the FTSS shall [SRS277] make the error type and its context data available to the application, then restart the offending task at its beginning.

5. For other software exceptions, regardless of whether or not a user written exception handling routine is invoked, if an exception occurs, the scheduler shall [SRS173], after making available the error type and context data to the application, resume processing (after the exception-handling routine runs, if provided) at the initialization point of the offending task.

6. For software exceptions occurring during Startup, FTSS shall [SRS301] issue a VME reset to the FCR in which the exception occurred.

3.2.3 Memory Management Services

3.2.3.1 Memory Protection

There are two types of memory violations that might occur: 1) as a result of a hardware fault or SEU and 2) as a result of a common mode (usually, software) error. Memory violations that result from random hardware faults will be detected in the same way as any other hardware fault is detected in the FTPP and don't require memory protection for them to be detected and dealt with. In the second case, NASA has determined that however the memory protection function is implemented, the policy will be to restart the task that is executing when a memory violation (exception) is detected.

The watchdog timer and ground based testing will uncover some but not all of the possible memory faults.

3.2.4 Communication Services

The FTSS communication services provide message-passing capabilities that are layered on top of the packet based network element communication hardware. Messages are contiguous blocks of variable length data that are transferred from one task to another. Messages are addressed with a global unique communication identifier that routes them to the appropriate virtual group (VG) and socket. Associated with the message are descriptor fields describing the sender, receiver, the type of message, and how the message is to be exchanged. The unique identifier for an end point consists of a virtual group identifier and a socket identifier. The sending and receiving end points may live on the same virtual group or on different virtual groups.

Communication Services are divided into two constituent capabilities: "Synchronous" message services and "Immediate" message services. "Synchronous" message services send and receive data on rate group frame boundaries; thus allowing safe inter-rate group communication. "Synchronous" message services are provided by message queue sockets. "Immediate" message services unlike "synchronous" message services initiate a message transfer immediately. When used for inter-VG communication, "immediate" message services interface directly with the Byzantine Resilient Virtual Circuit (BRVC) abstraction level communications interface and force an immediate network element access. "Immediate" message passing between virtual groups is restricted to the highest priority rate group on the FCP. This restriction does not apply to the ICPs. "Immediate" message passing within a virtual group is not restricted to the highest rate group, but must be used carefully by the application to prevent desynchronization. "Pipe" sockets provide "Immediate" message services.

Communication services provide a message passing capability that guarantees congruent use of the network element among the members of a virtual group under fault free conditions.

1. Communication services shall [SRS047] provide "synchronous" message passing services in the form of "message queues".

2. Communication services shall [SRS048] provide "immediate" message passing services in the form of "pipes". "Pipes" provide fast data throughput between virtual groups or within a virtual group when minimal data latency is necessary.

3. Communication services shall [SRS049] provide the capability to "broadcast" messages to all virtual groups.

4. Communication services shall [SRS050] restrict the use of "immediate" message passing services between virtual groups (from FCP to ICP) to tasks running in the highest rate group on the FCP. This restriction does NOT apply to the ICPs since they are running as simplex VGs.

5. Communication services shall [SRS051] detect message passing between application tasks living on the same virtual group and bypass the usage of the network element.

6. Communication services shall [SRS052] route messages to the proper virtual group(s) and socket.

7. Communication services shall [SRS053] deliver messages in the same order at each member of a virtual group.

8. Communication services shall [SRS054] perform synchronous message passing at rate group frame boundaries. This ensures that all redundant instantiations of a given rate group task have consistent messages throughout the rate group frame.

9. Communication services shall [SRS235] detect a babbling NE or ICP within 20 milliseconds of the receipt of the first erroneous packet.

10. FTSS shall [SRS255] mask out a babbling NE or ICP within 40 milliseconds after it is detected.

3.2.4.1 Sockets

Sockets are the end points of FTSS communication, which provide a transparent interface to the BRVC communications layer and a useful interface to the application layer. Sockets maintain the buffers between the underlying packet based communication primitives that directly access the network elements and the message based communication services used by the rate group tasks. Sockets used for "synchronous" message passing behave differently than those used for "immediate" message passing.

1. Synchronous message passing sockets shall [SRS055] queue outgoing messages until they are transmitted at frame boundaries. The "create" and "open" API calls for synchronous sockets allow the application to specify the maximum message size and how many incoming messages the socket may buffer.

2. If there is insufficient space to enqueue a message for transmission, Communication services shall [SRS059] return an error to the corresponding task. Sockets are non-blocking and place the burden of polling on the application task.

3.2.4.1.1 Message Queue Sockets

Message queue sockets allow a single task to queue a variable number of messages, each of variable length. One task is allowed to receive messages from this queue. Message queue sockets define a dedicated communication path between two tasks with guaranteed message delivery. Message queue sockets provide "synchronous" communication and perform sending/receiving of messages at frame boundaries.

1. Communication services shall [SRS062] provide a message queue communication mechanism that guarantees message delivery between a sending and receiving task.

2. Communication services shall [SRS063] provide an API for "message queue" communication.

3. Communication services shall [SRS064] provide the following error handling information as feedback to the "message queue" API calls:

a)   notification of invalid or out of range application specified parameters on all operations,

b)   notification of an attempt to create a broadcast message queue,

c)   message queue "open" of end point (SENDER/RECEIVER) by non-assigned virtual group,

d)   message queue is full when performing a send operation,

e)   connection/transmission error,

f)   FTSS unable to create/open message queue, and

g)   notification that a received message was truncated to the buffer size provided.

4. The message queue "create" API requires the application to specify the sending and receiving virtual group identifiers. Communication services shall [SRS066] only allow a single task living on each specified virtual group to "open" the respective end of the queue.

3.2.4.1.2 Pipe Sockets

"Pipe" sockets are used for "immediate" communication. They may be created with a broadcast capability. Pipes may only be opened by one sending task. Pipes may be opened by multiple receiving tasks if they are created with the "broadcast" capability; otherwise they may only be opened by one receiving task. Pipes are the only broadcast mechanism available to the application.

1. Communication services shall [SRS067] provide a "pipe" communication mechanism allowing immediate message passing through the network and allowing a 50hz FCP transfer task to poll until it can read an immediate message from the ICP.

2. Communication services shall [SRS068] provide an API for "pipe" communication.

3. Communication services shall [SRS069] provide the capability to create "pipe"s which "broadcast" their messages to all virtual groups.

4. Communication services shall [SRS070] provide the following error handling information as feedback to the "pipe" API calls:

a)   notification of invalid or out of range application specified parameters on all operations,

b)   notification of an attempt to create a broadcast pipe with an ICP as the sending virtual group,

c)   pipe "open" of end point (SENDER/RECEIVER) by non-assigned virtual group,

d)   notification upon receiving a message that the previous message was overwritten,

e)   connection/transmission error,

f)   FTSS unable to create/open pipe, and

g)   notification that a received message was truncated to the buffer size provided.

5. If the broadcast option is used, each virtual group should open the pipe and read from it to avoid flow control problems.

6. The "pipe" "create" API requires the application to specify the sending and receiving virtual group identifiers. Communication services shall [SRS073] only allow a single task living on each specified virtual group to "open" the respective end of the pipe. In the case of a broadcast "pipe", communication services allows one task in each virtual group of the system to open the receiving end of the "pipe".

3.2.5 Fault Detection and Isolation

Fault Detection and Isolation (FDI) provides the capability to detect and diagnose faults within FCC hardware. The functionality of FDI is decomposed into 2 capabilities-Initial Built-In Test (IBIT) and Continuous BIT (CBIT). FDI IBIT provides the facilities for the detection and diagnosis of faults during system initialization (at power on or CPU reset) on FCPs, ICPs, PMC 1553s, and MPCCs. FDI CBIT provides the facilities for the detection and diagnosis of faults on FCPs during all operational phases. In general, these tests execute system-wide tests using the fault tolerance characteristics of the FTPP architecture.

3.2.5.1 Initial BIT

Initial BIT constitutes a series of self-tests provided by the manufacturer of the equipment being tested. Initial BIT tests constitute tests of the processors, and I/O devices. Note that by configuring the network elements to automatically enter ISYNC on Power Up, there is no opportunity to perform IBIT on the NEs. The fact that an NE is in sync with the other NEs will have to substitute for a separate NE IBIT function.

FTSS IBIT executes on the Flight Control Processors (FCPs) at system initialization. These tests exercise the functionality of the various system components.

1. The FTSS software shall [SRS237] configure the FCP to act as the Radstone IBIT master, with the exception that the ICP on the NEFU is the master.

2. Requirement deleted.

3. The FTSS shall [SRS260] configure each FCP to perform IBIT Minimum Processing Environment (MPE) Tests, Power-up Tests, and Initial BIT on each FCP, as shown inTable 3.2-2. .

4. The FTSS shall [SRS261] configure each FCP to halt processing if any of the MPE tests fail.

5. The FTSS shall [SRS262] configure each FCP to continue processing if any of the Power-up or Initial BIT tests fail.

Table 3.2-2. FCP IBIT Table.

KIND OF TEST	TEST NAME
MPE Test	Program Programmable Read Only Memory (PROM) Test
MPE Test	Flash BootROM Checksum Test
MPE Test	On-board Random Access Memory (RAM) Test
MPE Test	Universe Device Test
Power-up Test	Timebase & Decrementer Test
Power-up Test	System Input Output Industry Standard Architecture (ISA) Bridge Test
Power-up Test	Main RAM Test
Power-up Test	Counter/Timer and Parallel I/O CIO Timers Test
Power-up Test	PowerPC Cache Test (on-chip only)
Power-up Test	PowerPC Memory Management Unit (MMU) Test
Power-up Test	PowerPC Floating Point Unit (FPU) Test
Power-up Test	Boot Flash Test
Power-up Test	Hardware Register Test
Initial BIT	Universe Test on Power-up

6. The FTSS shall [SRS287] configure each ICP to perform IBIT Minimum Processing Environment (MPE) Tests, Power-up Tests, and Initial BIT on each ICP, as shown inTable 3.2-3.

7. The FTSS shall [SRS288] configure each ICP to halt processing if any of the MPE tests fail.

8. The FTSS shall [SRS289] configure each ICP to continue processing if any of the Power-up or Initial BIT tests fail.

Table 3.2-3 ICP IBIT Table

KIND OF TEST	TEST NAME
MPE Test	Program Programmable Read Only Memory (PROM) Test
MPE Test	Flash BootROM Checksum Test
MPE Test	On-board Random Access Memory (RAM) Test
MPE Test	Universe Device Test
Power-up Test	Timebase & Decrementer Test
Power-up Test	System Input Output Industry Standard Architecture (ISA) Bridge Test
Power-up Test	Main RAM Test
Power-up Test	Counter/Timer and Parallel I/O CIO Timers Test
Power-up Test	PowerPC Cache Test (on-chip only)
Power-up Test	PowerPC Mass Memory Management Unit (MMU) Test
Power-up Test	PowerPC Floating Point Unit (FPU) Test
Power-up Test	Boot Flash Test
Power-up Test	Hardware Register Test
Power-up Test (ICPs only)	Enhanced Serial Communications Controller Test. (ICPs only)
Initial BIT	Universe Test on Power-up

9. The FTSS shall [SRS264] configure each ICP/PMC1553 to perform IBIT MPE Tests and Initial BIT as shown in Table 3.2-4.

10. The FTSS shall [SRS265] configure each ICP/PMC1553 to halt processing if any of the MPE tests fail.

11. The FTSS shall [SRS266] configure each ICP/PMC1553 to continue processing if any of the Initial BIT tests fail.

Table 3.2-4. ICP/PMC1553 IBIT Test Configuration.

KIND OF TEST	TEST NAME
MPE	Electrically Erasable Programmable Read Only Memory (EEPROM) header contents check to meet the expected values
MPE	EEPROM checksum for correct contents (test code and data are valid)
Initial BIT	Advance Communications Engine (ACE) 0 existence test
Initial BIT	ACE 0 RAM Test
Initial BIT	ACE 0 Register Test
Initial BIT	ACE 0 Interrupt Test

12. The FTSS shall [SRS267] configure each MPCC to perform MPE Tests as shown in Table 3.2-5.

13. Each MPCC is configured to halt processing if any of the MPE tests, listed in Table 3.2-5, fails.

Table 3.2-5. MPCC IBIT Test Configuration.

KIND OF TEST	TEST NAME
MPE	Control and Status Register Test
MPE	Erasable PROM Checksum Test
MPE	Local RAM Test
MPE	68020 Processor Test
MPE	Dual-port RAM Test

14. When the IBIT is complete, the FTSS in the channels that are part of the fault masking group shall [SRS239] report the results of IBIT for all Radstone boards to the application software for telemetry.

15. In IBIT failure cases that cause processing to halt, the failure shall [SRS269] be handled as described in Section 3.2.6.2, Recovery.

16. FTSS shall [SRS290], in ICP and FCP IBIT failure cases that allow processing to continue, after saving the results of IBIT for reporting to the application, in the first minor frame after Startup or recovery, consider the FCR to be failed, and start performing recovery actions for the FCR.

3.2.5.2 Continuous BIT

Continuous BIT executes on the FCP at all times after initialization is complete. In general, these tests execute system-wide tests using the fault tolerance characteristics of the FTPP architecture.

1. Continuous BIT, in conjunction with Redundancy Management and Scheduler operations running in the 50 Hz rategroup after the application tasks, shall [SRS091] take less than 2 milliseconds under nominal no-fault conditions.

2. Continuous BIT, in conjunction with Redundancy Management and Scheduler operations running in the 50 Hz rategroup after the application tasks, shall [SRS183] take less than 3 milliseconds while processing faults.

3. Continuous BIT shall [SRS093] execute on the FCP virtual group.

4. Continuous BIT shall [SRS094] reset the processor's built-in watchdog timer at 50 Hz. A failure to reset the watchdog timer within the allotted time (nominally 1.6 seconds) will generate a processor reset.

5. Continuous BIT shall [SRS095] exercise the presence test at 50 Hz to ensure that all processors in the FCP virtual group are synchronized.

6. The presence test shall [SRS184] also ascertain that all processors are executing the same 50 Hz, 10 Hz and 1 Hz frames.

7. Continuous BIT shall [SRS096] diagnose the faulty FCR within 1 second after detecting a failure.

8. Continuous BIT shall [SRS097] detect a failed ICP processor by detecting the absence of a periodic message for 2 consecutive minor cycles.

9. Continuous BIT shall [SRS098] report all diagnosed failures and recovery actions to the application for incorporation in the telemetry stream.

3.2.5.3 RAM Scrub

1. RAM scrub shall [SRS043] actively trigger the EDAC function by cyclically reading (and writing back if an error is found) all used RAM.

2. RAM scrub shall [SRS044] report detected errors to the application, congruently on all channels, via an API service for inclusion in the telemetry stream.

3. RAM scrub shall [SRS187] be capable of scrubbing at least 10 megabytes every 8 minutes, given at least 1% of the CPU is available for this processing.

4. RAM scrub shall [SRS275] not scrub the area used for telemetry data.

3.2.6 Redundancy Management

Redundancy Management maintains the mapping of physical hardware to virtual groups. This capability reconfigures the mapping in response to a diagnosis of a failed component, which can be a failed processor, failed network element or a link failure. Redundancy Management also performs transient fault analysis within constraints dictated by the mission management application software.

1. Redundancy Management shall [SRS099] provide an API call to enable the application to retrieve the health status of the processors, network elements, network element links, MPCCs, and ICP controlled interfaces.

2. Redundancy Management shall [SRS100] provide an API call to enable the application to request that the FTSS RM software initiate a voted reset of a channel.

3. Redundancy Management shall [SRS201] be able to accommodate power up of all 5 channels and maintain all 5 NEs active, assuming no failures.

3.2.6.1 Virtual Group Configuration

The virtual group configuration defines the mapping of physical hardware to virtual group(s). It defines the redundancy of each virtual group and the location of the processors

in the VME backplane. This mapping of virtual group member(s) is maintained as an ordered pair of network element and port on the associated network element.

1. Redundancy Management shall [SRS101] define an initial mapping of physical hardware to virtual group identifiers consisting of 1 quadruplex FCP virtual group and 5 ICP simplexes.

2. If an FCR is diagnosed as faulty during Startup, Redundancy Management shall [SRS102] exclude the FCP in the faulty channel from the initial FCP virtual group configuration.

3.2.6.2 Recovery

Redundancy Management configures the FCP virtual group, the network element, and the interconnection links for recovery of hardware resources in the operational system. Redundancy Management determines the recovery strategy to be executed based upon current configuration and whether alignment is permitted.

When a fault occurs, the configuration will change. The new configuration depends on the previous configuration. All the possible configuration changes are shown in Figure 3-2.
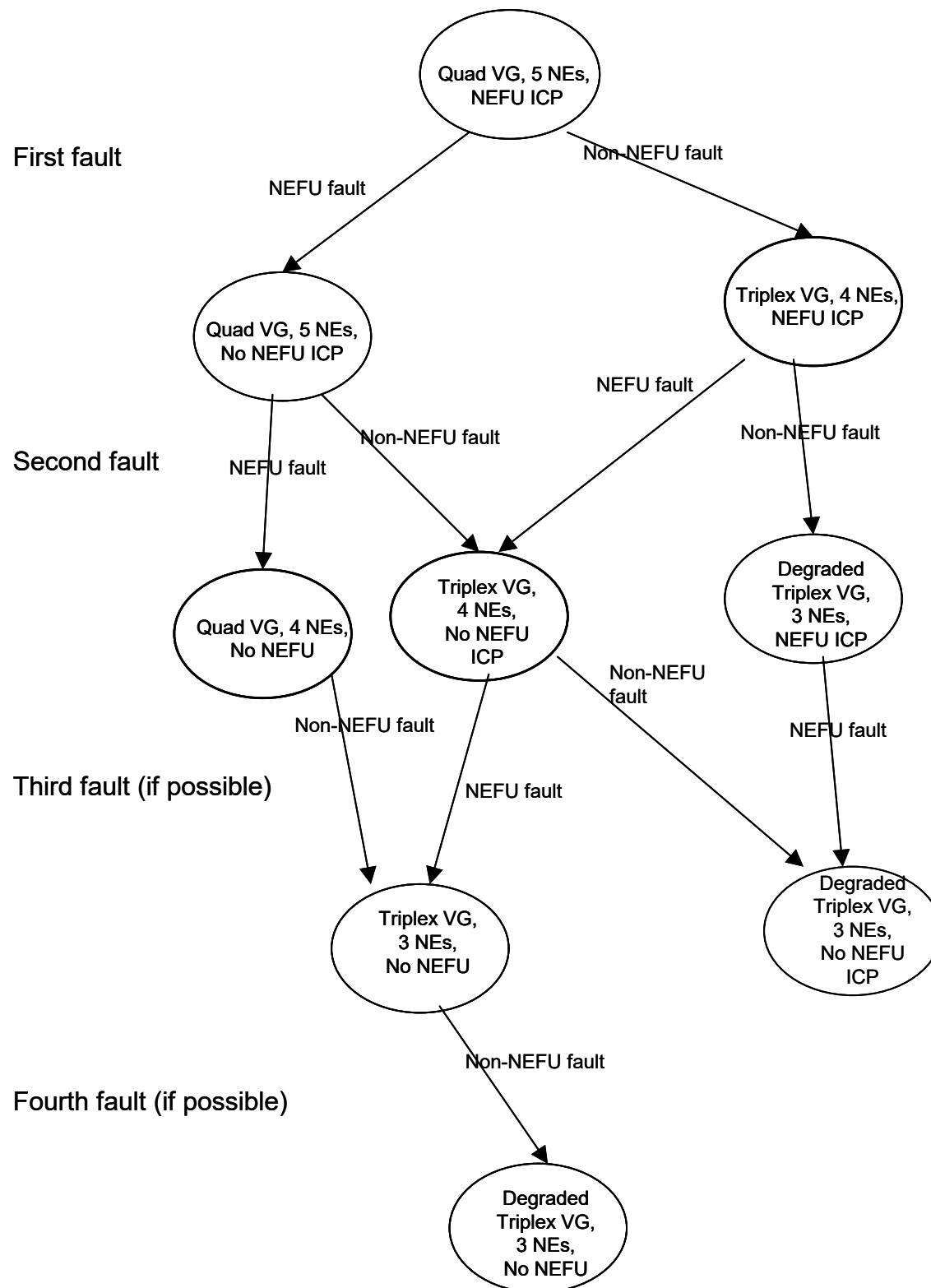
21

删除[Alex Edsall]: 10 August 2001

Figure 3-2 Fault-down Map

1. Redundancy Management shall [SRS104] implement the following strategies to reconfigure hardware resources:

    a)   degrade the FCP virtual group,

    b)   re-integrate an FCP processor into the FCP virtual group,

    c)   re-integrate a Network Element, or

    d)   mask a Network Element.

2. When the FCP virtual group is configured as a quadruplex and a failed FCR other than the NEFU has been diagnosed, Redundancy Management shall [SRS106] degrade the FCP virtual group to triplex, removing the FCR. The NE and the processors on the failed FCR will be removed from the NEs' Configuration Table (CT) and recovery of that channel will then take place, if alignment is permitted. Note that a failed FCR could be diagnosed using any method, including (but not limited to) Continuous BIT, ICP presence test, or NE syndrome analysis.

3. When the FCP virtual group is configured as a triplex, and if the NEFU is still active (4 NEs active total), and a failed FCR other than the NEFU has been diagnosed, Redundancy Management shall [SRS282] degrade the FCP virtual group to degraded triplex, removing the FCR. The NE and the processors on the failed FCR will be removed from the NEs' Configuration Table (CT) and recovery of that channel will then take place, if alignment is permitted. Note that a failed FCR could be diagnosed using any method, including (but not limited to) Continuous BIT, ICP presence test, or NE syndrome analysis.

4. If the FCP is configured as a triplex, and if the NEFU is not still active (3 NEs active total), and another failure in the FCP FCR is diagnosed, Redundancy Management shall [SRS284] mask out the processors on the failed FCR. The NE will remain in the CT and no recovery will take place. Note that a failed FCR could be diagnosed using any method, including (but not limited to) Continuous BIT, ICP presence test, or NE syndrome analysis.

5. If a failure in an FCR other than the NEFU is diagnosed when the FCP is configured as a degraded triplex, no action shall [SRS254] be taken. Note that a failed FCR could be diagnosed using any method, including (but not limited to) Continuous BIT, ICP presence test, or NE syndrome analysis.

6. For the NEFU, if the first failure is diagnosed, Redundancy Management shall [SRS245] issue a configuration update to mask out the failed processor. Note that the NE is allowed to remain in the configuration and no recovery will take place. Note also that the failed NEFU could be diagnosed using any method, including (but not limited to) ICP presence test, or NE syndrome analysis.

7. For the NEFU, if errors are identified after the processor has been masked out, and if at least 4 NEs are still active, the NE shall [SRS283] be removed from the configuration

and recovery will be attempted. Note that the NEFU recovery does not depend on whether alignment is permitted.

8. If the configuration needs to be changed due to a fault, as specified above, Redundancy Management shall [SRS128] issue a configuration update to mask out the failed network element.

9. Redundancy Management shall [SRS109] degrade the FCP virtual group within 3 minor frames of fault detection and isolation.

Recovery consists of the following steps:

10. Redundancy Management shall [SRS204] issue a voted reset to the failed channel, if alignment is permitted. (Note that NEFU recovery does not depend on whether alignment is permitted.)

11. Redundancy Management shall [SRS129] initiate transient NE recovery to restore Byzantine-resilient communications, if alignment is permitted. (Note that NEFU recovery does not depend on whether alignment is permitted.)

12. Redundancy Management shall [SRS110] reintegrate a failed FCP processor with the FCP virtual group when alignment is permitted and when the processor failure is not permanent.

13. From the time that the FCR failure has been identified, if the components of the FCR are recoverable and alignment is permitted, to the time the FCR is recovered, shall [SRS205] be no more than 1.5 minutes.

14. Redundancy Management shall [SRS208], within 60 milliseconds after 1.5 minutes has elapsed since the voted reset was sent to the failed channel, if the voted reset fails to recover the failed channel and alignment is still allowed, request from the application a power cycle of the channel. (Note that NEFU recovery does not depend on whether alignment is permitted.)

15. Redundancy Management shall [SRS209], within 60 milliseconds after 1.5 minutes has elapsed since the first power cycle request, if the FCR has not been recovered and alignment is still allowed, issue another request to the application for a power cycle of the channel. (Note that NEFU recovery does not depend on whether alignment is permitted.)

16. Redundancy Management shall [SRS211], if power cycle requests fail to result in a recovered channel, request the application to power down the channel and declare the channel to be permanently failed. Note that the same result will occur if the application software ignores or fails to respond to power cycle requests.

17. The application software shall [SRS285] have the capability to reset a permanently failed channel to its initial recovery state.

18. Redundancy Management shall [SRS117] reintegrate a processor that is temporarily disabled during a time when alignment was not permitted, when alignment is

subsequently permitted. Redundancy Management picks up where it left off in these attempts. For example, if Redundancy Management is at 1 minute in its 1.5 minute wait for a channel after the first power cycle request, and alignment is not allowed, when alignment is subsequently allowed Redundancy Management will wait another half minute and then try the next power cycle request.

19. An API call shall [SRS274] be provided that allows the application to notify FTSS that an FCR is intentionally being powered down.

20. Redundancy Management shall [SRS302] provide an API call to allow the application to specify whether recovery and alignment of failed FCRs is permitted. Note that recovery of the NEFU is always considered to be permitted.

3.2.6.2.1 Recovery from Processor Failure

This section deleted

3.2.6.2.1.1 Degrade Virtual Group

This section deleted.

3.2.6.2.1.2 Reintegrate Processor

When memory alignment is permitted, Redundancy Management attempts to reintegrate a processor with the other members of its original FCP virtual group by commanding the affected FCP virtual group to perform re-synchronization operations. Redundancy Management aligns the memory, clocks, cache, and other internal registers of the failed processor after synchronization has been achieved.

1. While synchronization is being attempted, the FCP virtual group shall [SRS123] maintain synchronous operations.

2. Only when memory alignment is permitted, Redundancy Management shall [SRS124] initiate periodic re-synchronization attempts on the FCP virtual group at a 1 second rate.

3. Redundancy Management shall [SRS125] perform memory alignment on a major frame boundary upon successful synchronization of all members of the FCP virtual group.

4. Redundancy Management shall [SRS281], during memory alignment, configure the NE to mask out the processor being re-synchronized.

5. Redundancy Management shall [SRS271] notify the application that alignment and reintegration of a processor will take place in 1 second.

6. Redundancy Management shall [SRS272] wait for the ICP to signal that it has completed initialization before suspending the application for memory re-alignment.

7. During alignment, Redundancy Management shall [SRS126] update MET (and, by extension, SEP).

8. Redundancy Management shall [SRS214], if alignment is permitted, incorporate a new channel within 1.5 minutes after power is applied to the channel.

9. Redundancy Management shall [SRS236], if alignment is permitted, serially incorporate two new channels if they are powered on simultaneously.

3.2.6.2.1.3 Data Management

Three types of memory are defined:

    a)   Congruent aligned   - always aligned

    b)   Congruent initialized   - initialized from nonvolatile memory

    c)   Non-congruent - never aligned

1. The FTSS API shall [SRS046] define a methodology for segregating and managing congruent aligned, congruent initialized, and non-congruent memory such that congruent aligned memory is aligned and congruent initialized memory is initialized during channel recovery. Non-congruent memory is not modified during realignment.

2. The FTSS API shall [SRS217] specify a memory map that provides the boundaries for congruent aligned memory, congruent initialized memory, and non-congruent memory.

3.2.6.2.1.4 Memory Alignment

Memory alignment occurs when a channel has been out of synchrony for some amount of time and then re-synchronizes with the other channels. The amount of time the channel is out of synchrony depends on the recovery mechanism. It could take as much as 4.5 minutes for the channel to be recovered and re-aligned (1.5 minutes per attempt for 3 attempts). The channel that is being brought back into synchrony is the "target" channel.

1. Memory alignment shall [SRS045] align processor state and congruent aligned memory locations. Processor state includes all registers. It also includes those timers used by FTSS.

2. The re-align function shall [SRS186] write the voted value from the currently synchronized channels into the target channel.

3. FTSS shall [SRS200] initialize congruent initialized memory locations from non-volatile memory.

4. Memory alignment shall [SRS203] take no more than 1 second per Megabyte of data to be realigned.

5. The FCP watchdog timer shall [SRS293] remain active during memory re-alignment.

6. Memory alignment shall [SRS294] reset the watchdog timer such that, in the absence of a fault, the timer never expires and resets the processor.

3.2.6.2.2 Recovery from Link Failure

This section deleted.

3.2.6.2.3 Recovery from Network Element Failure

This section deleted.

3.2.7 Time Services

1. Time Services shall [SRS142] provide Mission Elapsed Time and Separation Elapsed Time, with resolution partitioned as follows according to rate group, in order to guarantee identical copies of time representation across all FCPs:

    a)   50 Hz Mission Elapsed Time   20 milliseconds,

    b)   10 Hz Mission Elapsed Time   100 milliseconds,

    c)   1 Hz Mission Elapsed Time   1 second,

    d)   50 Hz Separation Elapsed Time   20 milliseconds,

    e)   10 Hz Separation Elapsed Time   100 milliseconds, and

    f)   1 Hz Separation Elapsed Time   1 second.

2. The Mission Elapsed Time shall [SRS218] have a drift rate of at worst 50 PPM.

3. The Mission Elapsed Time shall [SRS144] not rollover for 30 days.

4. The Separation Elapsed Time shall [SRS145] not rollover for 1 day.

5. The Mission Elapsed Time shall [SRS165] be initialized to zero at the first 50 Hz frame.

6. The Separation Elapsed Time shall [SRS161] be initialized to zero at startup, and start counting up in the next frame after being notified via an API call that the X-38 vehicle has been released from the Space Shuttle Remote Manipulator System.

7. The Separation Elapsed Time shall [SRS219] have a drift rate of at worst 50 PPM.

8. Time Services shall [SRS246] provide a utility timer to the application. Note that this timer is not voted, and must be assigned to a variable defined using non-congruent memory.

9. The utility timer shall [SRS247] have an accuracy equal to or better than 50 PPM.

10. The utility timer shall [SRS256] have a resolution equal to or better than 60.6 nanoseconds.

11. The utility timer shall [SRS248] shall be set to zero prior to the first application task running in the first minor frame of each major frame.

3.2.8 System Support Services

3.2.8.1 CTC Requirements

If transmission status indicates an error in telemetry and/or remote commanding operations 10 consecutive times, the following actions shall [SRS298] be taken:

1. Support Services shall [SRS299] switch to the redundant MPCC device to continue telemetry and/or remote commanding operations. Note that there are only two CTCs. CTC1 is connected to FCC1 and FCC3. CTC2 is connected to FCC2 and FCC4.

2. Support Services shall [SRS242] continue to close and reopen a faulty MPCC device until status shows that the device has recovered.

3. In all error cases, Support Services shall [SRS222] attempt to choose an error-free FCC-MPCC path, switching back and forth between channels if necessary.

4. Support Services shall [SRS286] provide an API call which allows the application to specify which MPCC channels in a C&T FCR should be used for telemetry and/or command reception.

3.2.8.1.1 Telemetry Requirements

The Telemetry Logging capability provides tasks with the capability for transmission to a telemetry-capturing device.

5. The telemetry capability shall [SRS148] be capable of transferring 12,800 bytes within the 10 Hz frame from the FCP.

6. The telemetry capability shall [SRS149] transfer the telemetry block from the FCP to the FCC-MPCC connected to the CTC.

7. The telemetry capability shall [SRS150] signal the FCC-MPCC to transfer the telemetry block to the CTC.

8. The telemetry capability shall [SRS300] provide status data to FTSS FDI about each FCC-MPCC RS-422 link to the CTC.

9. Support Services shall [SRS151] provide an API call to specify the address and length of a telemetry buffer.

10. Support Services shall [SRS257] use no more than 5.2 milliseconds of FCP processing time to move the telemetry data to the FCC-MPCC board and complete communication and error handling for the FCC-MPCC board.

3.2.8.1.2 Command Read Requirements

1. The Command Read capability shall [SRS152] check for the presence of a command and status message from each CTC on each FCC-MPCC at 10hz.

2. The Command Read capability on each FCP shall [SRS153] read the command data received from each CTC via the FCC-MPCC.

3. FTSS shall [SRS304] provide status data to the application about each FCC-MPCC RS-422 link to the CTC used for command data.

4. Support Services shall [SRS156] provide an API call to provide the current command data.

3.2.9 Power Down Services

FTSS shall [SRS249] provide an API call which closes and deletes all rate groups, deletes all communication mechanisms (including any internal to FTSS), and then deletes all tasks.

3.3 CSCI External Interface Requirements

3.3.1 Interface Identification and Diagram

The external interfaces to the FTSS CSCI are as follows:

1. Application Programming Interface

2. Network Element

3. Radstone

4. VxWorks

5. Flight Critical Processor-Instrument Control Processor

6. Multi-Processor Communications Controller

These interfaces are shown in Figure 3- 4 and elaborated further in subsequent paragraphs.

Figure 3- 4. Fault Tolerant System Services CSCI External Interfaces.

3.3.2 IRIG-B/FTSS Interfaces

This section deleted.

3.3.3 API/FTSS Interfaces

The Application Programmer's Interface (API) to Fault Tolerant System Services (FTSS) shall [SRS164] be as defined in the Application Programmer's Interface, Draper Document #297752.

3.3.4 Network Element/FTSS Interfaces

The Network Element (NE) provides fault tolerant communications among multiple virtual groups. The virtual groups are computational sites composed of processors. These processors may be configured as redundant virtual groups referred to as fault masking groups (FMGs) or as simplex virtual groups. Fault masking groups may consist of 2, 3, or 4 processors that execute identical control streams. A fault-masking group is composed of processors that reside in different fault-containment regions (FCR). Each FCR contains a Network Element (NE) and either 1 or 2 Processors (an FCP member on all but the fifth NE chassis, and an ICP). A simplex virtual group consists of a single processor. All virtual groups communicate with each other via the network element.

The Network Elements provide communication between virtual groups, keep the FCRs synchronized, and maintain data consensus among FCRs. The NEs are designed to implement the requirements for Byzantine resilience.

The Processing Elements are the computational sites. Each processor consists of a microprocessor, private RAM and ROM, and miscellaneous support devices, such as timers.

Interfaces between the Network Elements and the FTSS are shown in Figure 3- 6.

删除[Alex Edsall]: 10 August 2001

删除[Ruth Christopher]: Figure 3- 6

删除[kjm2514]: Figure 3- 4
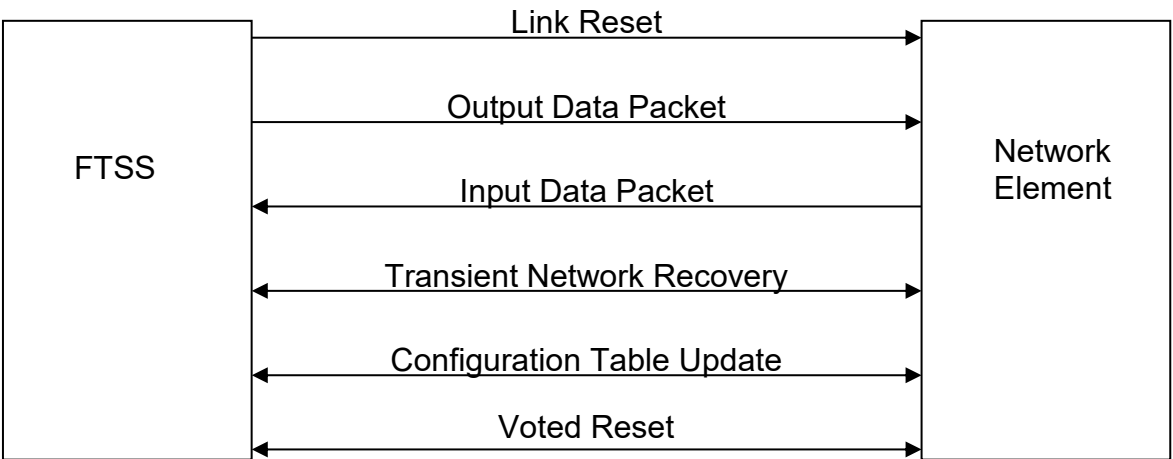
Figure 3- 6. Network Element Interfaces to FTSS CSCI.

删除[Ruth Christopher]: 6

删除[kjm2514]: 4

All transactions with the Network Element consist of a Data Descriptor Block and a Data Block. Each output transmission consists of an output descriptor block and an output data block. Each input reception consists of an input descriptor block and an input data block. The output descriptor and input descriptor blocks are defined in the Network Element Descriptor Block interfaces identified in Table 3.3-1. The output and input data blocks are defined in the Network Element Data Block Interfaces identified in Table 3.3-2; the format of the data blocks differs with the type of message transmitted. The table first identifies the type of message, and then provides the format of the data block for the given type.

Table 3.3-1. Network Element Descriptor Block Interface.

Identifier	Description	Source	Destination
Output Descriptor Block:			
Packet class	8-bit field. Selects the data exchange primitive to be executed by the NE.	FTSS CSCI	Network Element
toVID	8-bit field. Specifies the virtual group to which the packet is to be sent.	FTSS CSCI	Network Element
FromVID	8-bit field. Specifies the virtual group that sent the packet.	FTSS CSCI	Network Element
User Field	8-bit field. Used by FTSS.	FTSS CSCI	Network Element

Identifier	Description	Source	Destination
Input Descriptor Block:			
Packet class	8-bit field. Selects the data exchange primitive to be executed by the NE.	Network Element	FTSS CSCI
toVID	8-bit field. Specifies the virtual group to which the packet is to be sent.	Network Element	FTSS CSCI
FromVID	8-bit field. Specifies the virtual group that sent the packet.	Network Element	FTSS CSCI
User Field	8-bit field. Used by FTSS.	Network Element	FTSS CSCI
Vote Errors	Indicate if the data emanating from a participant during packet exchange disagreed with the majority in any way.	Network Element	FTSS CSCI
Clock Errors	Indicate that sometime since the last packet was exchanged by the NE, the FTC signal from the indicated NE fell outside the allowable skew window.	Network Element	FTSS CSCI
Link Errors	Indicate that sometime since the last packet was exchanged by the NE, an error was detected on the indicated fiber-optic link.	Network Element	FTSS CSCI
OBNE time-out	Indicates that the members of the source virtual group corresponding to the set bits did not request to send the packet within the allowable time skew.	Network Element	FTSS CSCI
IBNF time-out	Indicates that the members of the destination virtual group corresponding to the set bits did not free enough space in their input buffers to hold the incoming packet within the allowable time skew.	Network Element	FTSS CSCI
Scoreboard Vote Error	Indicates that the corresponding virtual group member did not agree with the majority regarding the type of packet to be exchanged.	Network Element	FTSS CSCI
Time stamp	32-bit field representing the time that the packet was exchanged.	Network Element	FTSS CSCI

Table 3.3-2. Network Element Data Block Interface.

Identifier	Description	Source	Destination
Output Data Packet:			
User data	64-byte packet of user data.	FTSS CSCI	Network Element
Input Data Packet:			
User data	64-byte packet of user data.	Network Element	FTSS CSCI

Identifier	Description	Source	Destination
Transient Network Element Recovery Packet:			
M(A)	TNR message as sourced by Network Element A.	Network Element	FTSS CSCI
M(B)	TNR message as sourced by Network Element B.	Network Element	FTSS CSCI
M(C)	TNR message as sourced by Network Element C.	Network Element	FTSS CSCI
M(D)	TNR message as sourced by Network Element D.	Network Element	FTSS CSCI
M(E)	TNR message as sourced by Network Element E.	Network Element	FTSS CSCI
RESULT	A byte indicating which NEs sourced the expected TNR message.	Network Element	FTSS CSCI
Configuration Table Update Packet:			
VID	The virtual group to be updated (VID 255 is reserved for NE and Clock masks).	Network Element/ FTSS CSCI	FTSS CSCI/ Network Element
Redundancy Level	The redundancy level to be used for the virtual group.	Network Element/ FTSS CSCI	FTSS CSCI/ Network Element
Processing Element Mask	Used to mask in selected members of the virtual group during data voting (VID 255 – NE mask).	Network Element/ FTSS CSCI	FTSS CSCI/ Network Element
Time-out	Selects the time-out to be used for the virtual group when calculating the OBNE and IBNF conditions (VID 255 – clock mask).	Network Element/ FTSS CSCI	FTSS CSCI/ Network Element
Member 0	Processor in the virtual group.	Network Element/ FTSS CSCI	FTSS CSCI/ Network Element
Member 1	Processor in the virtual group.	Network Element/	FTSS CSCI/ Network

Identifier	Description	Source	Destination
		FTSS CSCI	Element
Member 2	Processor in the virtual group.	Network Element/ FTSS CSCI	FTSS CSCI/ Network Element
Member 3	Processor in the virtual group.	Network Element/ FTSS CSCI	FTSS CSCI/ Network Element
Voted Reset Packet:			
VRESET Command	Byte 1- Command to perform the voted reset operation.	Network Element/ FTSS CSCI	FTSS CSCI/ Network Element
User (FTSS) defined	Remaining 63 bytes (unused).	Network Element/ FTSS CSCI	FTSS CSCI/ Network Element

3.3.5 Radstone/FTSS Interfaces

The Radstone firmware provides BIT capability on all Radstone boards on power up or reset. At the end of BIT the Radstone firmware saves the fault log.

Data element definitions for the Radstone/FTSS interface are shown in Table 3.3-3.

Table 3.3-3. Data Element Definition Table for Radstone/FTSS Interfaces.

Identifier	Description	Data Type	Limit/ Range	Source	Destination
Time Services:					
start_minor_cycle	count down timer interrupt	interrupt	NA	ISABridge HW timer	FTSS
utility_timer _value	Utility timer output	uword32	2^32	PPC time base register	FTSS
ticks	count down value for next minor cycle interrupt	uword16	65536	FTSS	ISABridge HW timer
Fault Detection and Isolation Services:					
rad_fault_log	Radstone self-test log	char []		Radstone firmware	FTSS

3.3.6 VxWorks/FTSS Interfaces

For VxWorks/FTSS interfaces see VxWorks Reference Manual. Appendix A of the API manual defines the allowable subset of VxWorks calls that can be safely used by the FCP application software.

3.3.7 Multi-Protocol Communications Controller (MPCC)/FTSS Interfaces

For MPCC/FTSS interfaces see the Radstone MPCC01 Firmware Manual, Pub #YD681MPCC1, and Radstone MPCC01 Hardware Manual, Pub #HH681MPCC1.

The telemetry serial line on the MPCC cards will be configured as follows:

- Mode: 0x1103

    o SLDC Mode

    o Buffered Transfer

    o Single Frame Transfer

    o Report a Break character, but do not close the RX channel

    o Normal Operation

        ▪ Note: CRC is always generated in SDLC mode

- Baud Speed: 2,097,152 bps

- Buffer size: 13,000 bytes

- No parity, 1 stop bit, 8 bit chars: 0x80

The command serial line on the MPCC cards will be configured as follows:

- Mode: 0x1103

    o SLDC Mode

    o Buffered Transfer

    o Single Frame Transfer

    o Report a Break character, but do not close the RX channel

    o Normal Operation

        ▪ Note: CRC is always generated in SDLC mode

- Buffer size: 332 bytes

35

- Baud Speed: 1,048,576 bps

- No parity, 1 stop bit, 8 bit chars: 0x80

3.3.8 FCP-ICP/FTSS Interfaces

The scheduler will populate shared memory with the data defined in Section 3.2.2.1, Item 21. The scheduler will issue a VME interrupt to the ICP every FCP 50 Hz minor frame. This interrupt will alert the ICP to enter a new ICP 50 Hz minor frame cycle. The minor frame number the ICP should be executing on is denoted by the value of the minor_frame identifier in shared memory.

Data element definitions for the ICP/FCP FTSS interface are shown in Table 3.3-4.

Table 3.3-4. Data Element Definition Table for FTSS Scheduler Interface.

Identifier	Description	Data Type	Limit/ Range	Source	Destination
minor_frame number	Minor Frame FCP is currently executing	int	0 to 49	FTSS CSCI	ICP
vehicle_mode	Current vehicle mode used to change tasking.	ulong	> 0	FTSS CSCI	ICP
met	FCP mission elapsed time	unsigned int	>0	FTSS CSCI	ICP
sep	FCP separation elapsed time	unsigned int	>0	FTSS CSCI	ICP
VME Interrupt	Interrupt raised from FCP side to vector an ISR on ICP side. This is for 50 Hz minor frame sync across processors	n/a	n/a	FCP	ICP

3.4 CSCI Internal Interface Requirements

There are no requirements for internal interfaces.

3.5 CSCI Internal Data Requirements

There are no requirements for internal data.

3.6 Adaptation Requirements

No requirements related to installation-dependent data or operational parameters have been identified.

3.7 Safety Requirements

No safety requirements have been identified.

3.8 Security and Privacy Requirements

No security requirements have been identified.

3.9 CSCI Environment Requirements

See Section 3.10.3.

3.10 Computer Resource Requirements

3.10.1 Computer Hardware Requirements

The FTSS shall [SRS158] execute on the Radstone Power PC 604R.

3.10.2 Computer Hardware Resource Utilization Requirements

The FTSS software and the VxWorks operating system, together shall [SRS193] utilize no more than 3 megabytes of ROM.

The largest single block of data transmitted on the VME Bus by the FTSS shall [SRS223] transmit in no longer than 100 microseconds.

All FTSS data provided for telemetry (as specified in the requirements) shall [SRS250] fit within the allocated budget of 5000 bits per second.

In addition, the FTSS software shall [SRS280] provide up to 600 bits of start-up data that indicates the state of the FTPP system during start-up.

Note that CPU usage limits, where needed, have been included in each of the sections with the requirements for the services provided. It is not possible to limit the total CPU usage of all services provided by the FTSS since the application calls the services an unknown number of times per major cycle.

3.10.3 Computer Software Requirements

The FTSS software shall [SRS159] be written in the C programming language.

FTSS shall [SRS160] use the VxWorks Operating System version 5.4.

The FTSS software and the VxWorks operating system shall [SRS258] utilize no more than 9 Megabytes of DRAM code and data space.

Of the 9 Megabytes of DRAM allocation, only 4 Megabytes of FTSS/VxWork's DRAM shall [SRS259] be re-aligned during any re-alignment attempts.

FTSS shall [SRS253] be compiled, linked and downloaded using Tornado 2 for the NT environment prior to delivery, for all engineering and formal releases.

FTSS object modules linked to the application on the four FCPs shall [SRS166] be identical.

After initial synchronization, the FCPs shall [SRS168] remain synchronized until a hardware fault occurs. For example, asymmetric I/O calls will not be allowed to induce a large enough skew to force the FCPs to desynchronize.

3.10.4 Computer Communications Requirements

NA

3.11 Software Quality Factors

NA

3.12 Design and Implementation Constraints

See Section 3.10.3.

3.13 Personnel-related Requirements

No personnel-related requirements have been identified.

3.14 Training-related Requirements

No training-related requirements have been identified.

3.15 Logistics-related Requirements

NA

3.16 Other Requirements

This section contains the requirements for the ICP.

3.16.1 ICP Services

FTSS shall [SRS303] provide an API call to allow the ICP application to determine on which channel it resides.

FTSS shall [SRS225] provide an API call to allow applications to send a status message to FDIR running on the FCP.

FTSS shall [SRS226] provide "immediate" message passing services in the form of "pipes". "Pipes" provide fast data throughput between virtual groups or within a virtual group when minimal data latency is necessary.

FTSS shall [SRS227] route messages to the proper virtual group(s) and socket.

If there is insufficient space to enqueue a message for transmission, FTSS shall [SRS228] return an error to the corresponding task. Sockets are non-blocking and place the burden of polling on the application task.

FTSS shall [SRS229] provide the following error handling information as feedback to the "pipe" API calls:

a)   notification of invalid or out of range application specified parameters on all operations,

b)   pipe "open" of end point (SENDER/RECEIVER) by non-assigned virtual group,

c)   notification upon receiving a message that the previous message was overwritten,

d)   connection/transmission error,

e)   FTSS unable to create/open pipe, and

f)   notification that a received message was truncated to the buffer size provided.

FTSS shall [SRS230] only allow a single task residing on each specified virtual group to "open" the respective end of the pipe.

The presence or absence of an NEFU ICP shall [SRS220] not impact the FTSS software (i.e. the FTSS ICP load will not be different).

The FTSS shall [SRS231] provide an API call to retrieve the current minor frame number sent from the FCP over the VME interface. Note that the NEFU ICP will not have this information since it does not have an FCP processor.

The FTSS shall [SRS232] provide an API call to retrieve the current MET value sent from the FCP over the VME interface. Note that the NEFU ICP will not have this information since it does not have an FCP processor.

The FTSS shall [SRS233] provide an API call to retrieve the current SEP value sent from the FCP over the VME interface. Note that the NEFU ICP will not have this information since it does not have an FCP processor.

The FTSS shall [SRS295] notify the application on the ICP, via an API call, 2 minor frames prior to an alignment.

3.17 Packaging Requirements

FTSS deliveries shall [SRS252] be made using CD ROM media.

3.18 Precedence and Criticality of Requirements

No precedence or criticality of requirements has been identified.

4. QUALIFICATION PROVISIONS

The following qualification methods will be used for the FTSS software.

Demonstration (D) - The operation of the CSCI (or some part of the CSCI) to observe its functional operation. The functional operation is directly observable, and it requires no elaborate instrumentation or special test equipment.

Test (T) - The operation of the CSCI (or a part of the CSCI) using instrumentation or other special test equipment to collect data for later analysis.

Analysis (A) - The processing of data accumulated from other qualification methods to determine correct results (e.g., interpretation of data collected by special test equipment).

Inspection (I) - The visual examination of CSCI code, documentation, etc.

The qualification methods that will be used for each software requirement are specified in the Certification Test Procedures document.

5. REQUIREMENTS TRACEABILITY

Table 5-1 provides the traceability between the X-38 Fault Tolerant Parallel Processor (FTPP) Requirements document number JSC 28671, and this document. The FTPP Requirements document specifies the FTSS system requirements. Table 5-1 is sorted by the system requirement number.

Table 5-1. FTPP to SRS Trace Table.

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
3.1	FTPP System Requirements	Each FTPP shall (3.1.1) consist of five (5) NEs (one for each FCC and one for the NEFU) and FTSS software.	NA
3.1	FTPP System Requirements	For the FTPP system (5 NEs per flight system), the contractor shall (3.1.2) deliver the following end products:	NA
3.1	FTPP System Requirements	The FTPP spare hardware shall (3.1.28) include one (1) radiation hardened FTPP set (5 NEs) and three (3) individual NEs including all optical connects, cables, and required accessories which are flight certified to meet the requirements specified herein for the X-38 space flight vehicles.	NA
3.1	FTPP System Requirements	The contractor shall (3.1.3) develop a preliminary design for the FTPP Architecture.	NA
3.1	FTPP System Requirements	This system shall (3.1.4) provide real time redundancy and fault tolerance among the four FCCs and the	SRS043, SRS091, SRS093, SRS094, SRS095, SRS096, SRS102, SRS104, SRS106, SRS109,

42

删除[Alex Edsall]: 10 August 2001

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		NEFU.	SRS128, SRS183, SRS184, SRS187, SRS235, SRS282, SRS283
3.1	FTPP System Requirements	The FTPP system shall not (3.1.5) solely exceed these timing requirement budgets.	SRS034, SRS035
3.1	FTPP System Requirements	In the presence of a maximum 2.5 second power-on skew, the FTPP system shall (3.1.6) be capable of completing FCC system power-up and initialization without synchronization errors.	SRS008
3.1	FTPP System Requirements	Following power being applied to all five chassis, the FTPP system shall (3.1.7) become operational in at most 1.5 minutes.	SRS015
3.1	FTPP System Requirements	The FTPP shall (3.1.26) detect a babbling NE or ICP within 20 milliseconds of the receipt of the first erroneous packet.	SRS235
3.1	FTPP System Requirements	The FTPP shall (3.1.27) recover from a babbling NE or ICP within 40 milliseconds after it is detected.	SRS255
3.1	FTPP System Requirements	An exchange of a single packet of data from an ICP to the FCPs via the NE shall (3.1.8) take no longer than 200 microseconds.	hw
3.1	FTPP System Requirements	An exchange and broadcast of a single packet of data from the FCPs to the ICPs and the FCPs via the NE shall (3.1.9) take	hw

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		no longer than 150 microseconds. A packet size is assumed to be 60 bytes	
3.1	FTPP System Requirements	Under no fault conditions, after initial power on, the FTPP system shall (3.1.10) create six Virtual Groups (VGs), a fault masking FCP, and the five ICPs, and enter the data in the NE Configuration Table (CT).	SRS101, SRS296
3.1	FTPP System Requirements	From the time that the NE failure has been identified, and the NE is recoverable, to the time the NE is recovered shall (3.1.11) be no more than 1.5 minutes.	SRS205
3.1	FTPP System Requirements	The FTPP shall (3.1.12) be capable of restoring a corrected faulty computer into the flight critical computer set.	SRS124, SRS214
3.1	FTPP System Requirements	The FTPP shall (3.1.14) take no more than 1 second per Megabyte of data to be realigned.	SRS203
3.1	FTPP System Requirements	The failed channel, provided it is recoverable, shall (3.1.15) be recovered in less than 1.5 minutes.	SRS205
3.1	FTPP System Requirements	The FTPP voting implementation shall (3.1.16) isolate and remove a faulty computer from the flight critical computer set within 60 milliseconds, once the	SRS109

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		fault has manifested itself.	
3.1	FTPP System Requirements	The FTPP system shall (3.1.17) be two fault tolerant for any two non-simultaneous hardware faults through out all phases of the X-38 mission (i.e., from power on to power off, without degradation).	SRS043, SRS091, SRS093, SRS094, SRS095, SRS096, SRS102, SRS104, SRS106, SRS109, SRS128, SRS183, SRS184, SRS187 SRS235, SRS282, SRS283, SRS281
3.1	FTPP System Requirements	The FTPP system shall (3.1.18) be capable of powering up and operating with any combination of 3 of the 5 FCR's active.	SRS010, SRS102
3.1	FTPP System Requirements	The FTPP system shall (3.1.19) be able to accommodate power up of all 5 channels and maintain all 5 NEs active, assuming no failures.	SRS201
3.1	FTPP System Requirements	The FTPP system shall (3.1.20) incorporate additional channels in the active set as they are powered-on by the application software.	SRS124, SRS125, SRS126, SRS214
3.1	FTPP System Requirements	After the initial power-up of only two FCRs and the NEFU, the FTPP system shall (3.1.25) be able to incorporate two more FCRs, upon their simultaneous or separate power-up.	SRS236
3.1	FTPP System Requirements	The FTPP system shall (3.1.21) monitor for the presence of new channels at a 1 Hz periodic rate.	SRS124
3.1	FTPP System Requirements	All healthy FCRs shall (3.1.22) be	SRS125, SRS126, SRS214

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		incorporated as they become available.	
3.1	FTPP System Requirements	All FCP processing channels (up to 4) shall (3.1.23) be incorporated into the FCP virtual group as they become available, provided that recovery and memory alignment is allowed by the application software.	SRS123, SRS124, SRS125, SRS126, SRS214
3.1	FTPP System Requirements	When memory realignment is not permitted, the FTPP system shall (3.1.24) maintain, at a minimum, 3 channels of I/O or 2 channels of I/O and the NEFU.	SRS254
3.2.1	Network Element Addressing Convention	The FCC hardware shall (3.2.1.1) use 3-digit binary numbers as outlined above for both addresses.	hw
3.2.2.1	Data Exchange Primitives	The X-38 NE shall (3.2.2.1.1) provide four types of data exchange primitives as described below, in accordance with the Byzantine-resilient replicated determinism requirements described in [1]. (Class 0, 1, 2, Broadcast)	hw
3.2.2.2	Configuration Table Updates	The NE shall (3.2.2.2.1) keep track of the grouping of physical processors into virtual groups. This mapping is contained in the Configuration Table (CT).	hw
3.2.2.2	Configuration Table Updates	The CT shall (3.2.2.2.2) also	hw

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		contain time-outs and vote masks.	
3.2.2.2	Configuration Table Updates	It shall (3.2.2.2.3) be possible to modify the CT whenever any of this information is changed by using a CT update primitive in a synchronous and atomic manner.	hw
3.2.2.3	Initial Synchronization (ISYNC)	The NE shall (3.2.2.3.1) be configured to automatically enter ISYNC microcode following power on.	hw
3.2.2.3	Initial Synchronization (ISYNC)	The NE shall (3.2.2.3.2) start transmitting a sync message using class 2 exchanges once 3 fault tolerant clocks have synchronized, thus enabling inter-NE data exchanges.	hw
3.2.2.3	Initial Synchronization (ISYNC)	A time-out period shall (3.2.2.3.3) be started when 3 NEs have joined in.	hw
3.2.2.3	Initial Synchronization (ISYNC)	The ISYNC procedure shall (3.2.2.3.4) terminate when all 5 NEs are synchronized, four NEs are synchronized (when only four NEs are powered), or after the time-out period.	hw
3.2.2.3	Initial Synchronization (ISYNC)	The NE microcode shall (3.2.2.3.6) initialize the time-outs and vote masks in the NE CT using values stored in the NE.	hw
3.2.2.3	Initial Synchronization (ISYNC)	The maximum time-out value shall (3.2.2.3.7) be 327.68 microseconds (256	hw

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		counts of the least significant bit of the fault tolerant clock which is 1.28 microseconds).	
3.2.2.4	Transient NE Recovery (TNR)	An NE that fails to synchronize with other NEs during ISYNC after a pre-defined time-out period shall (3.2.2.4.1) then enter TNR microcode.	hw
3.2.2.4	Transient NE Recovery (TNR)	An NE shall (3.2.2.4.12) directly enter TNR microcode after a voted reset or an NE watchdog timer reset.	hw
3.2.2.4	Transient NE Recovery (TNR)	It shall (3.2.2.4.2) stay in TNR mode indefinitely until a successful TNR exchange is observed.	hw
3.2.2.4	Transient NE Recovery (TNR)	The operational NEs shall (3.2.2.4.3) enter a "working group" TNR routine when the FCPs request to send a TNR packet.	hw
3.2.2.4	Transient NE Recovery (TNR)	If no new NE is observed, the functioning NEs shall (3.2.2.4.6) return to the operational mode within 500 microseconds.	hw
3.2.2.4	Transient NE Recovery (TNR)	If there is a new NE, the state of the reintegrated NE shall (3.2.2.4.13) be made congruent with the state of the operational NEs as follows.	hw
3.2.2.4	Transient NE Recovery (TNR)	The Configuration Table shall (3.2.2.4.7) be exchanged and voted into the newly	hw

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		recovered NE.	
3.2.2.4	Transient NE Recovery (TNR)	Time-outs in the scoreboard shall (3.2.2.4.8) be aligned by resetting all time-outs.	hw
3.2.2.4	Transient NE Recovery (TNR)	The global synchronous timer shall (3.2.2.4.9) be realigned by exchanging and voting the timer value. The timer value will stop incrementing until the realignment of the timer is complete.	hw
3.2.2.4	Transient NE Recovery (TNR)	Provided the failed NE is in a recoverable state, recovery of a failed NE shall (3.2.2.4.11) take no longer than 500 microseconds. This recovery requirement includes the time from which the FTSS software initiates the recovery to the time the recovery is complete.	hw
3.2.2.5	Voted Resets	There shall (3.2.2.5.1) be built-in support on the NE for voted resets, including a special packet type for executing the primitive.	hw
3.2.2.5	Voted Resets	The NE shall (3.2.2.5.3) provide the capability to perform a voted VME bus reset.	hw
3.2.2.6	Error Syndrome Reports	The NE shall (3.2.2.6.1) place error syndromes in the input information block whenever a packet is successfully delivered to the FCP.	hw

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
3.2.2.6	Error Syndrome Reports	The NE syndromes shall (3.2.2.6.2) be located in the second longword of the input information block buffer cell.	hw
3.2.2.6	Error Syndrome Reports	The NE syndromes shall (3.2.2.6.3) include indications of vote errors, fault-tolerant clock synchronization errors, and fiber-optic link errors, as detailed below.	hw
3.2.2.6	Error Syndrome Reports	Each syndrome shall (3.2.2.6.4) represent an occurrence of the indicated error at some time between delivery of the previous packet and delivery of the current packet.	hw
3.2.2.6	Error Syndrome Reports	The scoreboard syndromes shall (3.2.2.6.5) be located in the third longword of the input information block buffer cell.	hw
3.2.2.6	Error Syndrome Reports	They shall (3.2.2.6.6) include indications of scoreboard vote errors, Output Buffer Not Empty (OBNE) time-outs, and Input Buffer Not Full (IBNF) time-outs, as follows.	hw
3.2.2.6	Error Syndrome Reports	When a majority, but not a unanimity, of FCP members are observed with packets in their output buffers, a time-out shall (3.2.2.6.7) be initiated.	hw
3.2.2.6	Error Syndrome Reports	If the time-out expires before the other members transmit the	hw

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		packet, the remaining member shall (3.2.2.6.8) be ignored, the packet exchanged, and an OBNE time-out recorded.	
3.2.2.6	Error Syndrome Reports	When a majority, but not a unanimity, of FCP members are observed with room in their input buffers, a time-out shall (3.2.2.6.15) be initiated.	hw
3.2.2.6	Error Syndrome Reports	If the time-out expires before the other members have room in their input buffers, any member without room in their input buffer shall (3.2.2.6.16) be ignored, the packet exchanged, and an IBNF time-out recorded.	hw
3.2.2.7	Timestamps	The NE shall (3.2.2.7.1) place a timestamp in the input information block of each packet successfully delivered to an FCP or ICP.	hw
3.2.2.7	Timestamps	The timestamps shall (3.2.2.7.2) be congruent across all members of the destination FCP.	hw
3.2.2.7	Timestamps	The timestamps shall (3.2.2.7.3) also be congruent across all active processors in the case of a broadcast.	hw
3.2.2.7	Timestamps	The timestamp shall (3.2.2.7.4) be a 32-bit quantity that indicates relative time within the FCC.	hw

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
3.2.2.7	Timestamps	The resolution of the timestamp shall (3.2.2.7.5) be 1.28 microseconds.	hw
3.2.2.7	Timestamps	When the timestamp counter reaches the maximum value (Hex FFFFFFFF or approximately 5500 seconds), it shall (3.2.2.7.6) wrap around to zero.	hw
3.2.2.7	Timestamps	The timestamp counter shall (3.2.2.7.7) be initialized to zero during ISYNC.	hw
3.2.2.7	Timestamps	The counter shall (3.2.2.7.8) increase monotonically after that, except during TNR.	hw
3.2.2.7	Timestamps	The timestamps shall (3.2.2.7.9) be frozen during TNR until the realignment is complete.	hw
3.2.2.8	Debug Commands	The NE shall (3.2.2.8.1) implement in microcode support commands to aid in debugging new NEs and for performing stand-alone diagnostics and self-tests.	hw
3.2.2.8	Debug Commands	As a minimum, the following diagnostic functionality shall (3.2.2.8.2) be supported. 1. VMEbus Interface test 2. Message wraparound test 3. Class 1 Voter test 4. Class 2 Voter test 5. Input Buffer Test 6. Output Buffer test	hw

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		7. CT entry test 8. CT update test 9. Timestamp test	
3.2.3.1.1	Prototype NE Physical Characteristics	The prototype NE shall (3.2.3.1.1.1) reside on a single commercial grade 6U VME board.	hw
3.2.3.1.1	Prototype NE Physical Characteristics	The prototype NE shall (3.2.3.1.1.2) dissipate no more than 35 Watts.	hw
3.2.3.1.1	Prototype NE Physical Characteristics	The operating temperature range for the prototype NEs shall (3.2.3.1.1.3) be from 0 to 32.2º C at the inlet to the cooling fans.	hw
3.2.3.1.1	Prototype NE Physical Characteristics	The storage temperature range shall (3.2.3.1.1.4) be from - 30º to + 60º C.	hw
3.2.3.1.1	Prototype NE Physical Characteristics	The prototype NEs shall (3.2.3.1.1.5) use convection cooling.	hw
3.2.3.1.1	Prototype NE Physical Characteristics	The prototype NE shall (3.2.3.1.1.6) be fabricated using commercial grade components.	hw
3.2.3.1.2	Flight NE Physical Characteristics	Each flight NE shall (3.2.3.1.2.1) reside on a single ruggedized, wedge-locked, conduction-cooled 6U VME board.	hw
3.2.3.1.2	Flight NE Physical Characteristics	It shall (3.2.3.1.2.2) be able to be installed in an Air Transportable Rack (ATR) Chassis with .8 pitch spacing card slots.	hw
3.2.3.1.2	Flight NE Physical Characteristics	The conduction-cooled boards' mechanical core shall (3.2.3.1.2.3) be designed in	hw

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		accordance with IEEE 1101.2.	
3.2.3.1.2	Flight NE Physical Characteristics	Power de-coupling mechanisms shall (3.2.3.1.2.4) be designed into the NE.	hw
3.2.3.1.2	Flight NE Physical Characteristics	Backplane connector form factors shall (3.2.3.1.2.5) be in accordance with the VME64 with extensions (5 row P1 and P2) draft standard.	hw
3.2.3.1.2	Flight NE Physical Characteristics	Connector P2 shall (3.2.3.1.2.6) have all pins in row z connected to ground;Row d pins 1 through 31 connected to ground and row d pin 32 connected to +5 VDC.	hw
3.2.3.1.2	Flight NE Physical Characteristics	Connector P1 shall (3.2.3.1.2.7) have Row z pins 1 through 32 connected to ground, row d pins 1 and 32 connected to +5 VDC, row d pins 2 through 31 connected to ground with the exception of pins 3 through 8, pins 12, 14, 16, 18, 20, 22, 24, 26, 28, and 30 which will not be connected.	hw
3.2.3.1.2	Flight NE Physical Characteristics	Connector P1 pins 1 and 32 of row d shall (3.2.3.1.2.8) be connected to +5V.	hw
3.2.3.1.2	Flight NE Physical Characteristics	The inter-NE communications shall (3.2.3.1.2.9) be through fiber.	hw
3.2.3.1.2	Flight NE Physical Characteristics	Test points shall (3.2.3.1.2.10) be made available at the	hw

删除[Alex Edsall]: 10 August 2001

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		P2 connector for examining the operational status of the NE.	
3.2.3.1.2	Flight NE Physical Characteristics	Each flight NE shall (3.2.3.1.2.11) dissipate no more than 35 Watts.	hw
3.2.3.1.2	Flight NE Physical Characteristics	Each flight NE shall (3.2.3.1.2.12) be conduction-cooled.	hw
3.2.3.1.2	Flight NE Physical Characteristics	Flight hardware shall (3.2.3.1.2.13) be fabricated using radiation-hardened and/or radiation tolerant components.	hw
3.2.3.1.2	Flight NE Physical Characteristics	Fabrication and assembly of the boards shall (3.2.3.1.2.14) meet NAS 5300.4(3L), NAS 5300.4(3J-1), NHB5300.4(3A-2), IPC 275, IPC 6011, IPC 6012 and GSFC-S-312-P-003.	hw
3.2.3.1.3	Flight NE Environmental Qualification Conditions	The flight NE shall (3.2.3.1.3.1) be capable of meeting all performance requirements specified herein during and after exposure to the environmental service conditions specified herein.	hw
3.2.3.1.3	Flight NE Environmental Qualification Conditions	The flight NE shall (3.2.3.1.3.2) be designed and constructed so that no part of any component shifts in setting, position, or adjustment.	hw
3.2.3.1.3	Flight NE Environmental Qualification	No degradation shall (3.2.3.1.3.3) be caused in the	hw

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
	Conditions	performance that is specified in Subsections 3.2.3.1.3.1 through 3.2.3.1.3.11	
3.2.3.1.3.1	Temperature	The flight NE shall (3.2.3.1.3.1.1) meet the following temperature requirement while operating, 32 F to 149 F (0 C to 65 C) at the card edge.	hw
3.2.3.1.3.1	Temperature	The NE shall (3.2.3.1.3.1.2) operate with a worst case card edge thermal interface temperature of +65 °C for greater than 10 hours.	hw
3.2.3.1.3.1	Temperature	The storage (i.e., non-operating) temperature range for the flight NE shall (3.2.3.1.3.1.3) be from -30 °C to 60 °C.	hw
3.2.3.1.3.1	Temperature	For qualification testing, the flight NE shall (3.2.3.1.3.1.4) meet the following temperature requirement while operating, 12 F to 152 F (-11 C to 66.7 C) at the card edge.	hw
3.2.3.1.3.2	Vibration	The flight NE shall (3.2.3.1.3.2.1) be capable of complying with all of the performance specified herein while not operating during all specified levels.	hw
3.2.3.1.3.2.1	Random Vibration	The flight NE shall (3.2.3.1.3.2.1.1) be capable of withstanding the following environment in non-operational	hw

删除[Alex Edsall]: 10 August 2001

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		mode: Frequency, Hz Qualification Level g2/Hz 20 0.026 20-50 +3 dB/Octave 50-800 0.16 800-2000 -3 dB/Octave 2000 0.026	
3.2.3.1.3.2.1	Random Vibration	For qualification testing, the sweep time per axis shall (3.2.3.1.3.2.1.2) be 3 minutes, 14.1 g-rms overall.	hw
3.2.3.1.3.2.1	Random Vibration	The flight NE shall (3.2.3.1.3.2.1.3) be non operating during the application of this vibration in all axes.	hw
3.2.3.1.3.3	Ionization Radiation	The flight NE shall (3.2.3.1.3.3.1) be designed to be capable of complying with all the performance requirements specified herein while being subjected to the total dose, single event upset and latchup immune requirements in SSP 30512 Rev. C.	hw
3.2.3.1.3.3	Ionization Radiation	All radiation test results shall (3.2.3.1.3.3.7) be characterized and documented.	hw
3.2.3.1.3.4	Shock (Non-operating)	The flight NE shall (3.2.3.1.3.4.1) be designed to be capable of complying with all the performance requirements specified herein after being subjected to a total of	hw

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		six impact shocks, consisting of six shocks (one in each opposite direction) along each of three orthogonal axes.	
3.2.3.1.3.4	Shock (Non-operating)	The waveform and amplitude of the shock pulses shall (3.2.3.1.3.4.2) be sawtooth shock pulse 20g peak, 11 milliseconds nominal duration.	hw
3.2.3.1.3.5	Humidity	The flight NE shall (3.2.3.1.3.5.1) be designed to comply with all of the performance requirements specified herein while withstanding the effects of up to 90% relative humidity, non condensing while operating.	hw
3.2.3.1.3.6	Pressure	The flight NE shall (3.2.3.1.3.6.1) be designed to be capable of complying with all the performance requirements specified herein while non-operating in an atmosphere of 8 to 18 Psia.	hw
3.2.3.1.3.8	Electromagnetic Radiation	The flight NE shall (3.2.3.1.3.8.1) be designed to be electromagnetically compatible with the Radstone Power PC 604R	hw
3.2.3.1.3.10.1	MTBF	The design, manufacturing, and radiation environment composite failure rate/Mean Time	hw

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		Between Failures (MTBFs) of the flight NE shall (3.2.3.1.3.10.1.1) be predicted using techniques in MIL-HDBK 217 or other commonly accepted procedures.	
3.2.3.1.3.10.2	Operational Service Life	The useful operating service life shall (3.2.3.1.3.10.2.1) be a minimum of 30,000 hours.	hw
3.2.3.1.3.10.2	Operational Service Life	The useful operating life shall (3.2.3.1.3.10.2.2) be determined by analysis and presented at the critical design review.	hw
3.2.3.1.3.10.3	Storage Life	The flight NE storage life shall (3.2.3.1.3.10.3.1) be 5 years or better.	hw
3.2.3.1.3.12.1	Temperature Range	For acceptance testing, the flight NE shall (3.2.3.1.3.12.1.3) meet the following temperature requirement while operating, 32 F to 132 F (0 C to 55.6 C) at the card edge.	hw
3.2.3.1.3.12.2	Random Vibration Requirements	The ESS random vibration Power Spectral Density for the flight NE shall (3.2.3.1.3.12.2.1) be as follows: Total 10.8 grms …[Frequency, Hz Level g2/Hz 20 0.015 20-50 +3 dB/Octave 50-800 0.09	hw

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		800-2000 -3 dB/Octave 2000 0.015	
3.2.3.2	Operational Characteristics	The X-38 NE shall (3.2.3.2.1) be able to communicate with 4 other NEs.	hw
3.2.3.2	Operational Characteristics	The NE shall (3.2.3.2.2) support at least two physical processors per FCR.	hw
3.2.3.2	Operational Characteristics	The NE shall (3.2.3.2.3) support at least 6 virtual groups.	hw
3.2.3.2	Operational Characteristics	The NE shall (3.2.3.2.5) support class 1 and class 2 bandwidth of at least 1 Mbyte/sec.	hw
3.2.3.2	Operational Characteristics	The skew between NEs, measured at delivery of messages to dual-port RAM accessible by the processors via the VME bus, shall (3.2.3.2.6) be no more than 100 nsecs.	hw
3.2.3.2	Operational Characteristics	The NEs shall (3.2.3.2.7) be able to achieve phase-locked synchronization of the fault-tolerant clocks in less than 5 milliseconds from the time the last FCC powered up and exited the reset state	hw
3.2.4	Miscellaneous NE Requirements	The contractor shall (3.2.4.1) maintain a product identification and tracking system.	NA
3.2.4	Miscellaneous NE Requirements	Each NE and flight cable assembly shall (3.2.4.2) be identified by a part or type number and a unique serial number,	NA

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		consistent with the configuration management system and the specification for the contract.	
3.2.5	Network Element Fifth Unit (NEFU) Requirements	The presence, or absence of, an NEFU ICP shall (3.2.5.2) not impact the NE firmware (i.e., the NE firmware will not be different).	hw
3.3.1	Programming Language and Operating System [2]	Fault Tolerant System Services (FTSS) software shall (3.3.1.1) use VxWorks Version 5.4 as the Operating System.	SRS160
3.3.1	Programming Language and Operating System [2]	The software shall (3.3.1.2) be written in the C programming language, with the exception of the system loader software which may be written in scripts, and operate on a PowerPC 604R.	SRS158, SRS159
3.3.1	Programming Language and Operating System [2]	The FTSS software and the VxWorks operating system, together, shall (3.3.1.3) take up no more than 3 Megabytes of ROM, when loaded into FCP or ICP memory.	SRS193
3.3.1	Programming Language and Operating System [2]	The FTSS software and the VxWorks operating system shall (3.3.1.4) utilize no more than 9 Megabytes of DRAM code and data space.	SRS258
3.3.1	Programming Language and Operating System [2]	Of the 9 Megabytes of DRAM allocation, only 4 Megabytes of FTSS/VxWork's DRAM shall (3.3.1.5)	SRS259

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		be re-aligned during any re-alignment attempts.	
3.3.2	Start Up	Upon CPU reset caused by power on, watchdog timer or by other means, Start Up shall (3.3.2.1) execute the initial BIT (IBIT).	SRS237
3.3.2	Start Up	After successfully completing IBIT, the software shall (3.3.2.3) continue with the initialization of VxWorks and FTSS software.	SRS234
3.3.2	Start Up	If MPCC IBIT failed, the FTSS SW shall (3.3.2.24) switch to using the redundant MPCC card in that C&T FCR.	SRS299
3.3.2	Start Up	If the 5th ICP fails, the FTSS SW shall (3.3.2.25) ignore the error and allow the NE to continue the synch process and become part of the voting group, if possible.	SRS243
3.3.2	Start Up	On each FCP, the FTPP system shall (3.3.2.26) configure the Radstone firmware to perform the IBIT tests shown in Table 3.3.2.1, FCP IBIT Table.	SRS260
3.3.2	Start Up	The FTPP system shall (3.3.2.27) configure the Radstone processor to halt processing if any of the MPE tests, mentioned in Table 3.3.2.1, FCP IBIT Table, fail.	SRS261
3.3.2	Start Up	The FTPP system	SRS262

删除[Alex Edsall]: 10 August 2001

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		shall (3.3.2.28) configure the Radstone processor to continue processing if any of the Power-up tests or Initial BIT tests mentioned in Table 3.3.2.1, FCP IBIT Table, fail.	
3.3.2	Start Up	In all FCP IBIT cases, provided the hardware state permits, the FTSS shall (3.3.2.29) log the error and report it in the X-38 telemetry stream.	SRS239
3.3.2	Start Up	Upon completion of logging a Power-up test or Initial BIT test failure, the FTSS system shall (3.3.2.43) consider the FCP failed and attempt recovery actions as stated in requirement 3.3.5.29.	SRS290
3.3.2	Start Up	On each ICP, the FTPP system shall (3.3.2.30) configure the Radstone firmware to perform the IBIT tests shown in Table 3.3.2.2, ICP IBIT Table.	SRS287
3.3.2	Start Up	The FTPP system shall (3.3.2.31) configure the Radstone processor to halt processing if any of the MPE tests, mentioned in Table 3.3.2.2, ICP IBIT Table, fail.	SRS288
3.3.2	Start Up	The FTPP system shall (3.3.2.32) configure the Radstone processor to continue processing if any of the Power-up	SRS289

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		tests or Initial BIT tests mentioned in Table 3.3.2.2, ICP IBIT Table, fail.	
3.3.2	Start Up	In all ICP IBIT cases, provided the hardware state permits, the FTSS shall (3.3.2.33) log the error and report it in the X-38 telemetry stream.	SRS239
3.3.2	Start Up	Upon completion of logging a Power-up test or Initial BIT test failure, the FTSS system shall (3.3.2.44) consider the ICP failed and attempt recovery actions as stated in requirement 3.3.5.29.	SRS290
3.3.2	Start Up	On each ICP/PMC1553, the FTPP system shall (3.3.2.34) configure the Radstone firmware to perform the IBIT tests shown in Table 3.3.2.3, ICP/PMC1553 IBIT Table.	SRS264
3.3.2	Start Up	The FTPP system shall (3.3.2.35) configure the Radstone ICP/PMC1553 card to halt processing if any of the MPE tests, mentioned in Table 3.3.2.3, ICP/PMC1553 IBIT Table, fail.	SRS265
3.3.2	Start Up	The FTPP system shall (3.3.2.36) configure the Radstone ICP/PMC1553 card to continue processing if any of the Initial BIT tests mentioned in Table 3.3.2.3, ICP/PMC1553 IBIT	SRS266

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		Table, fail.	
3.3.2	Start Up	In all ICP/PMC1553 IBIT cases, provided the hardware state permits, the FTSS shall (3.3.2.37) log the error and report it in the X-38 telemetry stream.	SRS239
3.3.2	Start Up	On each MPCC, the FTPP system shall (3.3.2.38) configure the Radstone firmware to perform the IBIT tests shown in Table 3.3.2.4, MPCC IBIT Table.	SRS267
3.3.2	Start Up	In all MPCC IBIT cases, provided the hardware state permits, the FTSS shall (3.3.2.40) log the error and report it in the X-38 telemetry stream.	SRS239
3.3.2	Start Up	If IBIT fails, the FTSS SW shall (3.3.2.41) handle the failure as stated in the preceding IBIT requirements and in requirement 3.3.5.29's table, FTPP Failure Response/Recovery Mechanisms.	SRS269
3.3.2	Start Up	However, the FTSS SW shall (3.3.2.42) notify the application software if the 5th ICP's heartbeat ceases to exist.	SRS098
3.3.2	Start Up	The surviving triplex shall (3.3.2.5) attempt to sync with the failed FCP.	SRS177
3.3.2	Start Up	If the failed FCP has not synced in 2.5 seconds, after the	SRS178

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		surviving triplex has detected the loss of the FCP, then the surviving triplex shall (3.3.2.6) send a voted VMEbus reset through the NE to the failed FCP.	
3.3.2	Start Up	Start Up shall (3.3.2.12) synchronize its FCP with other operational FCPs.	SRS008
3.3.2	Start Up	Start Up shall (3.3.2.13) make their state congruent.	SRS011
3.3.2	Start Up	The FCP state shall (3.3.2.14) include all volatile memory, read/write memory, registers, timers, and counters, except that part of the memory exclusively set aside for channel-unique information.	SRS011
3.3.2	Start Up	It shall (3.3.2.17) support normal synchronization following power on or reset.	SRS194
3.3.2	Start Up	Start Up shall (3.3.2.18) be able to synchronize all operational FCPs in the presence of this skew in the power on sequence.	SRS008
3.3.2	Start Up	Start Up shall (3.3.2.19) test to ensure that all four FCPs are synchronized.	SRS008, SRS010
3.3.2	Start Up	Unsynchronized processors shall (3.3.2.20) be excluded from the FCP configuration.	SRS010, SRS296
3.3.2	Start Up	During start up the	SRS014, SRS292

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		FCP watchdog timer shall (3.3.2.45) be active.	
3.3.2	Start Up	System Initialization shall (3.3.2.21) initiate execution of FTSS and X-38 application code.	SRS199
3.3.3	Vehicle/Mission Manager	The Mission Manager Template shall (3.3.3.1) provide a mechanism for (but not limited to) creating and controlling task execution, creating message queues and other interprocessor communication mechanisms, and provide on/off capability for processor resynchronization.	SRS215, SRS302
3.3.4	Scheduler	The Scheduler shall (3.3.4.1) support three rate groups: 50 Hz (minor frame), 10 Hz (medium frame), and 1 Hz (major frame).	SRS197, SRS195, SRS035
3.3.4	Scheduler	The FTSS software shall (3.3.4.18) take at most 1 msec of a 50 Hz minor frame.	SRS024, SRS034
3.3.4	Scheduler	The Scheduler shall (3.3.4.3) set the timer to a count down value so as to cause the next minor frame interrupt at 20 msec (+/- 330 usecs) from the previous interrupt congruently in all operational FCPs.	SRS035, SRS181
3.3.4	Scheduler	The FTSS software shall (3.3.4.19) provide an API call which provides the application program	SRS278

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		the minor frame number.	
3.3.4	Scheduler	Process scheduling shall (3.3.4.4) only be performed at certain controlled locations (synchronization points).	SRS022
3.3.4	Scheduler	The Scheduler shall (3.3.4.5) place processing time bounds on all rate groups to ensure that no rate group monopolizes the FCC's processor.	SRS028
3.3.4	Scheduler	It shall (3.3.4.6) be possible to reassign a task to a different rate group as a function of the mission mode.	SRS017, SRS018, SRS195, SRS196, SRS197, SRS198
3.3.4	Scheduler	Tasks within a rate group shall (3.3.4.7) be executed in the order in which the mission manager registers the tasks.	SRS022, SRS039, SRS037, SRS198
3.3.4	Scheduler	It shall (3.3.4.8) be possible to alter the execution sequence of tasks within a rate group as a function of mission mode.	SRS002, SRS017, SRS196, SRS018, SRS019, SRS020, SRS021, SRS197, SRS195, SRS198
3.3.4	Scheduler	Higher iteration tasks shall (3.3.4.9) have higher priority over lower iteration tasks.	SRS027
3.3.4	Scheduler	The Scheduler shall (3.3.4.12) detect 50 Hz, 10 Hz, and 1 Hz frame overruns at the next frame following the end of their respective rate boundaries.	SRS028
3.3.4	Scheduler	The Scheduler shall (3.3.4.13) attempt recovery from a frame	SRS030

删除[Alex Edsall]: 10 August 2001

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		overrun according to the following policy:	
3.3.4	Scheduler	if the scheduler determines that a task did not finish within its specified rate boundary, the scheduler shall (3.3.4.14) signal that a task overrun occurred.	SRS030
3.3.4	Scheduler	When the task restart begins, the FTSS shall (3.3.4.15) provide a mechanism to signal the task to execute its startup recovery actions, including updating the I-Load data and pre-stored last good data state.	SRS030
3.3.4	Scheduler	Following a task overrun, the scheduler shall (3.3.4.17) provide an application programmer's interface call which specifies which task was running within the rate group which has overrun.	SRS216
3.3.4	Scheduler	The FTSS software shall (3.3.4.20) provide a task deadline capability which allows an application to specify which minor frame that an application should start in and finish in.	SRS270
3.3.4	Scheduler	The Scheduler in the FTPP shall (3.3.4.16) keep all redundant copies of a process, which are executing in different computers, in synchronization.	SRS181
3.3.5	Fault Detection, Identification, and	The scope of FDIR shall (3.3.5.1) be	SRS095, SRS096, SRS184, SRS235,

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
	Recovery	limited to the hardware on the four FCP boards, the four MPCC/CTC boards, the five ICPs, and the five NEs.	SRS298, SRS299, SRS300, SRS304
3.3.5	Fault Detection, Identification, and Recovery	FDIR shall (3.3.5.2) receive this information and, after two consecutive missed "heartbeats," conclude that the ICP is failed.	SRS097
3.3.5	Fault Detection, Identification, and Recovery	FDIR shall (3.3.5.3) report the total FCC status to the Vehicle/Mission Manager when requested to do so by the Vehicle/Mission Manager.	SRS098, SRS099
3.3.5	Fault Detection, Identification, and Recovery	The FDIR shall (3.3.5.4) execute CBIT during all operational phases.	SRS095
3.3.5	Fault Detection, Identification, and Recovery	The CBIT shall (3.3.5.5) be executed at a 50 Hz rate, after all 50 Hz flight critical operations are complete.	SRS093, SRS095, SRS034
3.3.5	Fault Detection, Identification, and Recovery	The CBIT, at a minimum, shall (3.3.5.6) include a "presence test" to ascertain that all FCP processors are synchronized and are at the same relative point in time in the current minor frame.	SRS093, SRS095, SRS184
3.3.5	Fault Detection, Identification, and Recovery	The presence test shall (3.3.5.7) also ascertain that all processors are executing the same 50 Hz, 10 Hz, and 1 Hz frames.	SRS184

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
3.3.5	Fault Detection, Identification, and Recovery	The CBIT shall (3.3.5.8) also arm and reset the hardware watchdog timer.	SRS014, SRS094
3.3.5	Fault Detection, Identification, and Recovery	The CBIT shall (3.3.5.10) be executed without interfering with the normal execution of the application tasks.	SRS034
3.3.5	Fault Detection, Identification, and Recovery	The FDIR shall (3.3.5.11) not take more than 2 msec per minor frame under nominal no-fault conditions.	SRS091
3.3.5	Fault Detection, Identification, and Recovery	The FDIR shall (3.3.5.12) not take more than 3 msec per minor frame while processing faults.	SRS183
3.3.5	Fault Detection, Identification, and Recovery	The FDIR shall (3.3.5.13) be able to discriminate between permanent and non-permanent faults.	SRS106, SRS110, SRS117, SRS204, SRS208, SRS209, SRS211, SRS282, SRS298
3.3.5	Fault Detection, Identification, and Recovery	The FDIR shall (3.3.5.14) reset and retry the failed entity, such as an FCP or an NE, to perform this discrimination.	SRS106, SRS110, SRS117, SRS129 SRS204, SRS208, SRS209, SRS211, SRS282
3.3.5	Fault Detection, Identification, and Recovery	To clear the failure, FDIR shall (3.3.5.15) request the Vehicle/Mission Manager to cycle power to that FCR.	SRS208, SRS209
3.3.5	Fault Detection, Identification, and Recovery	The FDIR shall (3.3.5.16) be able to identify a fault source, at least to an FCR.	SRS095, SRS184, SRS096, SRS097
3.3.5	Fault Detection, Identification, and Recovery	The FDIR shall (3.3.5.17) place all fault and recovery information in shared memory for inclusion in the frames that will	SRS098, SRS044

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		be telemetred and recorded by the CTC.	
3.3.5	Fault Detection, Identification, and Recovery	For the first permanent FCP failure, FDIR shall (3.3.5.30) degrade the redundancy level of the FCP from 4 to 3.	SRS106
3.3.5	Fault Detection, Identification, and Recovery	If a second permanent FCP failure occurs, then FDIR shall (3.3.5.31) degrade the redundancy level of the FCP from 3 to 2 and operate in a degraded triplex mode.	SRS282
3.3.5	Fault Detection, Identification, and Recovery	Additionally, FDIR shall (3.3.5.20) reinitialize and integrate an FCP if permitted by the Vehicle/Mission Manager.	SRS104, SRS110, SRS123, SRS124, SRS125, SRS126, SRS281, SRS302
3.3.5	Fault Detection, Identification, and Recovery	FTSS shall (3.3.5.35) notify the applications that memory re-alignment and re-integration of an FCP is going to occur in 1 second.	SRS271
3.3.5	Fault Detection, Identification, and Recovery	FTSS shall (3.3.5.36) wait for the ICP to signal that it has completed initialization before suspending the application for memory re-alignment.	SRS272
3.3.5	Fault Detection, Identification, and Recovery	The FCP watchdog timer shall (3.3.5.38) remain active during memory re-alignment.	SRS293, SRS294
3.3.5	Fault Detection, Identification, and Recovery	Reintegration of an FCP shall (3.3.5.21) be completed in at most 1.5 minutes.	SRS214
3.3.5	Fault Detection, Identification, and	It shall (3.3.5.22) be possible to perform	SRS204

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
	Recovery	voted VMEbus resets via the NEs.	
3.3.5	Fault Detection, Identification, and Recovery	For a permanent NE failure, FDIR shall (3.3.5.23) mask the failed NE.	SRS104, SRS245
3.3.5	Fault Detection, Identification, and Recovery	For a transient NE failure, FDIR shall (3.3.5.24) mask the failed NE.	SRS104, SRS106, SRS245, SRS282
3.3.5	Fault Detection, Identification, and Recovery	Additionally, FDIR shall (3.3.5.25) reinitialize and integrate the NE.	SRS104
3.3.5	Fault Detection, Identification, and Recovery	Intentional powering down of an FCP, ICP, or NE shall (3.3.5.32) not be classified as a fault.	SRS285, SRS274, SRS128
3.3.5	Fault Detection, Identification, and Recovery	FTSS shall (3.3.5.34) provide an API call which allows the application to notify FTSS that an FCP, ICP, or NE is intentionally being powered down.	SRS274
3.3.5	Fault Detection, Identification, and Recovery	FTSS shall (3.3.5.37) provide an API call which allows the application to take an FCR out of the permanently failed state and place it back in the initial recovery state.	SRS285
3.3.5	Fault Detection, Identification, and Recovery	A failed FCP or NE shall (3.3.5.27) be masked within three minor frames of fault detection and isolation.	SRS109
3.3.5	Fault Detection, Identification, and Recovery	The FTSS FDIR shall (3.3.5.28) exchange the status information of detected faults in the FCP, ICP, NE, and MPCC/CTC hardware	SRS044, SRS098

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		with the NASA provided software.	
3.3.5	Fault Detection, Identification, and Recovery	The FTPP system shall (3.3.5.29) perform the "FTPP Failure Response/Recovery Mechanisms"as listed in the following matrix and notes of interest.	SRS100, SRS104, SRS242, SRS222, SRS208, SRS209, SRS211, SRS245, SRS283, SRS284, SRS298, SRS299, SRS300, SRS304
3.3.6	Communications	Synchronous communication shall (3.3.6.1) be in the form of messages enqueued for transmission at the start of the next rate group frame and dequeued for reading by the recipient task within the next rate group frame after it is received.	SRS047, SRS052, SRS063, SRS064
3.3.6	Communications	A transmit packet queue and a receive packet queue shall (3.3.6.7) be maintained for each task or Communication ID (CID).	SRS053, SRS054, SRS055, SRS059, SRS062, SRS066
3.3.6	Communications	Access to the transmit queues shall (3.3.6.8) be controlled within the communication service primitives .	SRS062, SRS066
3.3.6	Communications	Message passing communications primitives shall (3.3.6.9) be provided for task-to-task communication as well as for broadcast to all processors.	SRS047, SRS048, SRS049, SRS051 SRS052, SRS062, SRS069
3.3.6	Communications	Broadcast primitives shall (3.3.6.10) not be available on ICPs.	SRS064, SRS070
3.3.6	Communications	For the highest rate	SRS048, SRS050,

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		group tasks (i.e., tasks that can not be preempted), Immediate Message Services shall (3.3.6.11) also be provided.	SRS067, SRS068, SRS069, SRS070, SRS073
3.3.6	Communications	A version of the Immediate Message Services shall (3.3.6.12) be provided to the ICPs that allows Class 2 writes to NEs and Class 1 reads from NEs.	SRS226, SRS227, SRS228, SRS229, SRS230, SRS303
3.3.6	Communications	Communications services shall (3.3.6.13) provide a version of Immediate Message Services between rate groups within the FCP that bypasses the NE and that can be used to control and monitor inter-rate group communications.	SRS051
3.3.6	Communications	Communication services shall (3.3.6.14) provide the capability for a "helper" task to be created to run in the 50 Hz rate group, but running in specific minor cycles (every 5th or every 50th) to provide data from the ICP to the lower rate tasks.	SRS042
3.3.7	System Loader	The procedures used to build the executable FTSS software using a cross-compiler and linker, down-load the image to the target processors, and burn the load image into the Radstone	NA

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		PowerPC flash RAM shall (3.3.7.1) be documented in the release notes that accompany Engineering Releases of the FTSS software and in the Software Users Manual.	
3.3.7	System Loader	Any makefiles or other automated scripts that support the build, down-load, and flash programming processes shall (3.3.7.2) be delivered with the software to NASA.	NA
3.3.8	Memory Management	For each mission mode, the congruent and non-congruent memory boundaries shall (3.3.8.1) be known and fixed.	SRS217
3.3.8	Memory Management	The Memory Management software shall (3.3.8.2) periodically "scrub" volatile and read/write memory in the FCP.	SRS043
3.3.8	Memory Management	It shall (3.3.8.3) not be necessary to scrub memory that is not used by the flight software.	SRS043
3.3.8	Memory Management	It shall (3.3.8.16) not be necessary to scrub that area used to store telemetry data.	SRS275
3.3.8	Memory Management	Memory scrubbing shall (3.3.8.5) be executed without interfering with normal execution of applications tasks.	SRS187
3.3.8	Memory Management	The memory scrubbing software shall (3.3.8.6) be	SRS187

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		capable of scrubbing 10 Megabytes in 8 minutes.	
3.3.8	Memory Management	The RAM scrub software shall (3.3.8.15) at most use 1% of an FCP CPU duty cycle.	SRS187
3.3.8	Memory Management	Even though memory scrubbing is performed locally and the errors would not be congruent, the recording of errors shall (3.3.8.10) be congruent.	SRS044
3.3.8	Memory Management	To support reintegrating a desynchronized channel, as specified in the FDIR requirements, the Memory Management software shall (3.3.8.11) "re-align" all of the volatile and read/write congruent memory, registers, timers and other locations that fit the description of "volatile and read/write congruent locations".	SRS045, SRS126, SRS186, SRS200
3.3.8	Memory Management	The re-align function shall (3.3.8.12) write the voted value from the good channels into the target channel.	SRS186, SRS281
3.3.8	Memory Management	The re-align function shall (3.3.8.13) be allowed only when permitted by the Vehicle/Mission Manager.	SRS110, SRS117, SRS125, SRS302
3.3.8	Memory Management	Memory Management software shall (3.3.8.14) include (but not be limited to) memory scrubbing	SRS043, SRS044, SRS046, SRS217, SRS045, SRS186, SRS200, SRS203, SRS187

删除[Alex Edsall]: 10 August 2001

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		and memory realignment.	
3.3.9	Memory Protection	Memory shall (3.3.9.1) be categorized into congruent (identical data) and non-congruent (data that is not identical) memory.	SRS046
3.3.10	Time Management	Time Management shall (3.3.10.4) provide MET.	SRS142
3.3.10	Time Management	The MET shall (3.3.10.5) be initialized to zero at the first 50 Hz frame.	SRS165
3.3.10	Time Management	The MET shall (3.3.10.6) measure real-time from this event with an accuracy of at most 50 Parts Per Million (PPM).	SRS218
3.3.10	Time Management	The MET shall (3.3.10.7) have a resolution of 20 msec. for 50 Hz tasks, 100 msec for 10 Hz tasks, and 1 second for 1 Hz tasks.	SRS142
3.3.10	Time Management	The MET shall (3.3.10.8) be able to increment to at least 30 days without rolling over.	SRS144
3.3.10	Time Management	The MET shall (3.3.10.9) be congruent across all FCP members.	SRS142
3.3.10	Time Management	Following a processor recovery, during which time is frozen, the FTSS software shall (3.3.10.10) account for the frozen time and update MET to its proper value.	SRS218
3.3.10	Time Management	Time Management shall (3.3.10.11)	SRS142

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		provide SEP.	
3.3.10	Time Management	Time Management shall (3.3.10.12) initialize SEP to zero within one minor cycle of the time when the vehicle/mission manager software has notified the FTSS software that the X-38 vehicle is released from the Space Shuttle Remote Manipulator System.	SRS161
3.3.10	Time Management	The SEP shall (3.3.10.13) measure real-time from this event with an accuracy of at most 50 Parts Per Million (PPM).	SRS219
3.3.10	Time Management	The SEP shall (3.3.10.14) have a resolution of 20 msec. for 50 Hz tasks, 100 msec for 10 Hz tasks, and 1 second for 1 Hz tasks.	SRS142
3.3.10	Time Management	The SEP shall (3.3.10.15) be able to increment to at least 1 day without rolling over.	SRS145
3.3.10	Time Management	The SEP shall (3.3.10.16) be congruent across all FCP members.	SRS142
3.3.10	Time Management	Following a processor recovery, during which time is frozen, the FTSS software shall (3.3.10.17) account for the frozen time and update SEP to its proper value.	SRS219
3.3.10	Time Management	If the SEP API call is made prior to actual separation, the call	SRS161

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		shall (3.3.10.27) return zero (0).	
3.3.10	Time Management	The Time Services, if dealing with the year designation, shall (3.3.10.21) be Year 2000-compliant.	No year designation used in any requirement
3.3.10	Time Management	Time Management shall (3.3.10.23) provide a utility timer.	SRS246, SRS248
3.3.10	Time Management	The utility timer shall (3.3.10.24) be available via an FTSS API call(s).	SRS246
3.3.10	Time Management	The utility timer shall (3.3.10.25) have a resolution of 60.6 nanoseconds.	SRS256
3.3.10	Time Management	The utility timer shall (3.3.10.26) have an accuracy of at most 50 PPM.	SRS247
3.3.11	Input/Output Services	Load modules of the four FCPs shall (3.3.11.1) be identical.	SRS166
3.3.11	Input/Output Services	Control flow of the four FCPs shall (3.3.11.2) be similar, if not identical.	SRS008, SRS011, SRS181, SRS191, SRS053, SRS054, SRS095, SRS184, SRS123, SRS125, SRS126, SRS045, SRS186, SRS200, SRS166, SRS168
3.3.11	Input/Output Services	Asymmetric I/O calls shall (3.3.11.3) not be allowed to induce a large enough skew to force the FCPs to desynchronize.	SRS168
3.3.11	Input/Output Services	A subset of FTSS shall (3.3.11.7) reside on the ICP.	SRS225, SRS226, SRS227, SRS228, SRS229, SRS230
3.3.11	Input/Output Services	FTSS shall (3.3.11.8) provide an API call which allows the application to specify which MPCC channels in a C&T FCR should	SRS286

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		be used for telemetry and/or command reception.	
3.3.12	Exception Handling	Upon occurrence of an exception, the FTPP system shall (3.3.12.1) log the error and include all context data relevant to the exception e.g. the contents of the Machine State Register (MSR) and the machine status Save/Restore Registers (SRR0 & SRR1).	SRS172
3.3.12	Exception Handling	The error type and its context data shall (3.3.12.4) be made available to the application via an API call.	SRS172
3.3.12	Exception Handling	For software exceptions, the FTPP system shall (3.3.12.2) then transfer control to a user specified exception handling routine, if one is provided.	SRS031
3.3.12	Exception Handling	For hardware exceptions, the FTPP system shall (3.3.12.5) "handle" the exception by making the error and its context data available to the application and then returning from the exception handler.	SRS276
3.3.12	Exception Handling	For reserved exceptions, the FTPP system shall (3.3.12.6) "handle" the exception by making the error and its context data available to the application and then	SRS276

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		returning from the exception handler.	
3.3.12	Exception Handling	Finally, for software exceptions only, the FTPP system shall (3.3.12.3) then "jump back" to the initialization point for the offending task.	SRS173
3.3.12	Exception Handling	If the exception occurs within the FTSS software, the FTPP system shall (3.3.12.7) "jump" back to the beginning of the task, skip all initialization code, and begin processing the task's code again.	SRS277, SRS301
3.3.13	Application Interface	An application programming interface shall (3.3.13.3) be documented in a FTSS API document.	SRS164
3.3.14	NEFU	The presence, or absence of, an NEFU ICP shall (3.3.14.1) not impact the FTSS software (i.e., the FTSS ICP load will not be different).	SRS220
3.3.15	Power Down	FTSS services shall (3.3.15) provide an API call which provides the capability to close and delete all communication mechanisms delete all rate groups, and suspend and delete all tasks.	SRS249
3.4.1.1	FCP-ICP Communication Architecture	The FCP-ICP communications shall (3.4.1.1.1) provide the following capabilities (these are written from the viewpoint of the FCP): 1. Signal the	SRS025, SRS032, SRS033, SRS048, SRS097, SRS231, SRS232, SRS233, SRS295

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		start of the 50 Hz rate group in the ICP. 2. Synchronize frames across the four ICPs. 3. Receive congruent sensor data from ICPs. 4. Send voted actuator and other output device commands to ICPs. 5. Receive health and status of the ICPs and all the FCC hardware for which the ICPs are responsible. 6. Provide the ICPs with current minor frame number, X-38 flight phase/segment number, vehicle mode number, MET, and SEP. 7. Provide the ICPs with notification of FCP memory alignment two minor frames prior to the start of the alignment.	
3.4.1.2	FCP-ICP Communication Requirements	As part of start-up or after recovering from a transient fault: after completing IBIT, the FCP shall (3.4.1.2.1) wait 15 seconds for the ICP to initialize all of its non-Radstone VME slave boards and its NE interface	SRS297
3.4.1.2	FCP-ICP Communication Requirements	then the FCP VG shall (3.4.1.2.2) send the FCP VG Ready Signal to the ICPs to indicate that the FCP VG is ready to begin FCP-ICP communications.	SRS221
3.4.1.2	FCP-ICP Communication Requirements	To permit these task and pipe initializations in the ICPs, the FCP VG shall (3.4.1.2.8) wait at least 2.5	SRS189

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		seconds for the ICP Ready Signals after the FCP VG has been notified that the ICPs received the FCP VG Ready Signals.	
3.4.1.2	FCP-ICP Communication Requirements	The FCP shall (3.4.1.2.5) signal the start of each minor frame in all ICPs by means of a VMEbus IRQ5 interrupt.	SRS032
3.4.1.2	FCP-ICP Communication Requirements	The interrupts across all channels shall (3.4.1.2.6) have a skew no greater than 330 microseconds.	SRS191
3.4.1.2	FCP-ICP Communication Requirements	Each interrupt shall (3.4.1.2.7) be preceded by the FCP writing the information listed in 3.4.1.1.1-6 to a shared memory block over the VME backplane bus to its counterpart ICP.	SRS033
3.4.2	FCP-CTC Communications	The FTSS software shall (3.4.2.2) provide to the telemetry program the FTPP telemetry data which consists of the following elements: a. FCP status, b. NE status, c. # Transient Errors, d. Transient recovery attempts, e. Frame overruns, f. ICP status.	SRS029, SRS044, SRS098
3.4.2	FCP-CTC Communications	The data mentioned in requirement 3.4.2.2 and any other Draper-provided telemetry data shall (3.4.2.9) fit within Draper's allocated telemetry budget of 5000 bits/sec.	SRS250

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
3.4.2	FCP-CTC Communications	In addition, the FTSS software shall (3.4.2.11) provide up to 600 bits of start-up data that indicates the state of the FTPP system during start-up.	SRS280
3.4.2	FCP-CTC Communications	Once every medium frame, the FTSS software shall (3.4.2.3) accept a pointer from the telemetry program to the buffer space containing the telemetry data.	SRS148, SRS151
3.4.2	FCP-CTC Communications	The FTSS software shall (3.4.2.4) move this buffer to the MPCC/CTC over the VMEbus.	SRS149, SRS150
3.4.2	FCP-CTC Communications	The FTSS software shall (3.4.2.10) use no more than 5.2 milliseconds of FCP processing time to move the telemetry data to the MPCC/CTC board and complete communication and error handling for the MPCC/CTC board.	SRS257
3.4.2	FCP-CTC Communications	The FTSS software shall (3.4.2.5) receive telemetry commands from both CTCs via the MPCC/CTC once every medium frame.	SRS152
3.4.2	FCP-CTC Communications	The FTSS software shall (3.4.2.6) congruently decide which FCP channel should be the source of CTC data.	SRS222
3.4.2	FCP-CTC Communications	This decision shall (3.4.2.7) be made based on the health	SRS222

FTPP Section #	FTPP Section Name	FTPP Requirement	SRS Req #s
		and status of all the physical links from the FCP to the CTC.	
3.4.2	FCP-CTC Communications	The FTSS software shall (3.4.2.8) deliver to the requisite NASA application program commands received from both CTCs.	SRS153, SRS156
3.5	Miscellaneous Other Requirements	Draper shall (3.5.5) not violate the 100 microseconds requirement.	SRS223
3.5	Miscellaneous Other Requirements	Draper shall (3.5.6) deliver FTSS engineering release version 5/6, and any subsequent versions of the FTSS software, only after the release has been proven to work under Tornado 2 for the NT environment.	SRS253
3.5	Miscellaneous Other Requirements	Draper shall (3.5.7) deliver FTSS engineering release version 5/6, and any subsequent versions of the FTSS software, via CD-ROM.	SRS252

6. NOTES

6.1 List of Acronyms

Acronym	Definition
API	Application Programmer's Interface
BIT	Built-in-Test
BRVC	Byzantine Resilient Virtual Circuit
BSP	Board Support Package
CBIT	Continuous Built In Test
COTS	Commercial Off-The-Shelf
CSCI	Computer Software Configuration Item
CT	Configuration Table
CTC	Command and Telemetry Computer
DID	Data Item Description
DIO	Digital Input/Output
ECR	Engineering Change Request
FCC	Flight Critical Computer
FCP	Flight Critical Processor
FCR	Fault Containment Region
FDI	Fault Detection and Isolation
FMG	Fault Masking Group
FTC	Fault Tolerant Clock
FTPP	Fault Tolerant Parallel Processor
FTSS	Fault Tolerant System Services
IBIT	Initial Built In Test
ICP	Instrument Control Processor
ID	IDentifier
IRS	Interface Requirements Specification
ISR	Interrupt Service Routine
ISYNC	Initial Synchronization
MET	Mission Elapsed Time
MPCC	Multi-Protocol Communications Controller
MPE	Minimum Processing Environment
NASA	National Aeronautics and Space Administration
NE	Network Element
PPC	Power PC
PPM	Parts Per Million
RAM	Random Access Memory

ROM	Read Only Memory
RM	Redundancy Management
RTC	Real Time Clock
SEP	SEParation elapsed time
SRS	Software Requirements Specification
TAEM	Terminal Area Energy Management
TNR	Transient Network Element Recovery
VG	Virtual Group
VME	Versa Module Eurocard

6.2 Glossary

Babbler – A processor or NE that continually unexpectedly sends messages over the fiber network, thus overloading the fiber network with unnecessary traffic.

Byzantine Faults – Faults consisting of arbitrary behavior on the part of failed components, and may include stopping and then restarting execution at a future time, sending conflicting information to different destinations, and in short, anything within a failed component's power to attempt to corrupt the system. [1]

Byzantine Resilient Virtual Circuit (BRVC) – An abstracted view of the Network Elements and the fiber optic interconnection network. The NE hardware and fiber optic inter-connection network appear to the software a virtual message passing interface with certain guarantees about the order and consistency of message delivery in the face of arbitrarily malicious faults.

Debug mode – A mode of operating the network elements that allows software to control the operation of the network elements as well as to test their operation. This mode is used by the NE stand alone test software and is one of two modes that the NE can be configured to power up into. For X38, the flight NEs will not be configured to power up into this mode.

Degraded triplex – A fault containment region that has 2 processors, but at least 3 NEs. The FCR is configured to have a third processor that is masked out. The third processor could be "on" the NEFU (even thought there is no FCP software loaded on the processor card on that channel), in order to maintain the VG.

Exceptions.- External signals, errors, or unusual conditions arising in the execution of instructions. When exceptions occur, information about the state of the processor is saved to certain registers and the processor begins execution at an address (exception vector) predetermined for each exception.

Fault Containment Region (FCR) – A set of hardware that meets the following criteria: Electrical Isolation, Independent Power, Independent Clocking and, if necessary, physical separation. Errors internal to an FCR are contained within the FCR and errors outside the FCR do not adversely affect the operation of an FCR, i.e., external errors are prevented from inducing errors in the FCR. Among other characteristics, Byzantine resilience

depends upon the concept of Fault Containment Regions. Throughout this document, the terms FCR and channel are used interchangeably.

Hardware Exceptions.- Any exception not mapped to a VxWorks signal.

Initial Synchronization (ISYNC) – The process (mode) by which the network elements initially achieve synchronization at power up.

Link – A one way, point to point connection between the transmitter in one Network Element (NE) and a receiver in another NE.

Lost soul sync – The process by which a lost (i.e., not in sync with the other members of its virtual group) processor is brought into synchronization with the other members of a redundant virtual group.

Masking –Preventing erroneous data from propagating beyond the voters (the act of voting masks any single error from propagating beyond the voters, i.e., the voters deliver a majority result in the presence of one fault.) Also, the setting of the voters to ignore input from a channel known (or suspected) of being faulty. Setting the voters to ignore (mask) the input from a channel know to be faulty is also referred to as reconfiguring the voters.

NE watchdog timer reset – A reset of the network element that arises as a result of the NE's on-board watchdog timer not being pulsed in a timely, periodic manner.

Permanent Failure – A failure is declared permanent when all attempts to reconfigure the channel and remove the failure by means of either a FTSS Response/Recovery Mechanism or a Application Failure Response/Recovery Mechanism have failed.

Power on reset – A reset of an entire channel that is the result of initially powering on the channel or cycling power to the channel.

Power-up skew – Power-on skew is defined here to be the time between switching on power to the first processor and power being applied to the fifth processor.

Re-integration – The process of bringing a "Lost Soul" processor back into synchronization with the other members of a redundant virtual group and then re-aligning its internal state including congruent memory, processor registers, and timers.

Scoreboard – That part of the network element that acts as the message scheduler. Among other things, the scoreboards in the redundant NEs, acting as a group, collectively decide when messages are ready to be sent.

Simplex – A non-redundant virtual group. When a single processor forms a virtual group, this is called a simplex virtual group. In the X-38 architecture, each ICP is a simplex virtual group. The FCPs form a single redundant virtual group (often called a quad).

Simultaneous Failure – A second fault is considered simultaneous if the fault occurs between the time the first fault is observed and when the system is reconfigured in response to the fault. The nature of this reconfiguration will vary depending upon the

system's configuration at the time of the fault and the effect(s) of the fault. Refer to FTSS SRS/IRS section 3.2.6.2 for reconfiguration details.

Software Exceptions.- Any exception mapped to a VxWorks signal.

Start-up – The period of time from when power is initially applied to the system or a reset is asserted to an individual channel until the 50 Hz interrupt is enabled. Activities occurring during this time include but are not limited to IBIT, ISYNC, TNR, processors synchronization, task create and initialization, definition and initialization of Communications Services sockets, etc.

Synchronization – The process of coordinating in time, multiple hardware and/or software entities. For redundant entities, this means bringing them to the same point within some allowable skew. For FCP-ICP communications and operation, this refers to the process of coordinating their activities in time both at start up and during steady-state operation using a variety of methodologies.

TNR mode – The mode that a network element enters when it finds itself alone and unable to synchronize with other NEs.

Transient Failure –Faults, such as those caused by SEUs, that can be recovered either by the FTSS Response/Recovery Mechanism or by the Application Failure Response/Recovery Mechanism.

Transient Network Element Recovery (TNR) – The process of recovering and bringing a lost network element into synchronization with the operating (working group) NEs and aligning or initializing its internal state so that it can resume synchronous operation with the other NEs.

Virtual group –Virtual processor capable of accepting work in a parallel processing environment. They are comprised of processor entities, each of which must be resident in a different fault containment region. [1] In the X-38 architecture, each ICP is a simplex virtual group. The FCPs form a single redundant virtual group (often called a quad).

VMEbus reset – A reset applied to the VMEbus.

Voted reset – A reset that is applied to an NE as the direct result of the FCP processors in the other channels agreeing to use and then actually using the voted reset function of the NEs to reset a faulty channel. An NE will assert a VMEbus reset and enter TNR directly (by-passing ISYNC) as a result of receiving a voted reset from the other NEs.

Working group TNR routine – The software commanded function that causes the NEs still in synch with one another to first look for and then synchronize with a lost NE.