Defence Research and Development Canada
Recherche et développement pour la défense Canada

DEFENCE R&D DÉFENSE

FASSET Refurbishment

B. Liao, R. Addison and J.D. Lambert

Defence R&D Canada – Ottawa
TECHNICAL MEMORANDUM
DRDC Ottawa TM 2004-134
July 2004

Canada

FASSET Refurbishment

B. Liao
DRDC Ottawa

R. Addison
Communications Research Centre

J. D. Lambert
DRDC Ottawa

Defence R&D Canada – Ottawa

Technical Memorandum

DRDC Ottawa TM 2004-134

July 2004

Abstract

The FASSET system is an extremely high frequency (EHF) satellite communications payload and ground terminal test bed, originally developed under contract for Defence R&D Canada – Ottawa (DRDC Ottawa). In July of 2002, DRDC Ottawa signed a contract with Harris Corporation that resulted in the refurbishment and delivery of FASSET to assist in the development of their EHF modem. This document details the repairs and improvements that were conducted for Harris in support of the contract. The primary goal of the refurbishment was to reconstitute FASSET to the state of functionality that it possessed prior to being stored for many years.

This report covers the significant refurbishment events including items repaired, modifications and improvements, and unresolved problems. It provides a technical reference of knowledge collected as well as a record and explanation of refurbishment events. Other topics covered include test equipment, maintenance procedures, lessons learned, and future work.

Résumé

Le système FASSET est un banc d'essai sur terminal terrestre et à charge utile de communications par satellite aux fréquences extrêmement hautes (EHF) qui a été conçu par contrat à l'intention de R & D pour la défense Canada – Ottawa (RDDC Ottawa). En juillet 2002, RDDC Ottawa a signé un contrat avec Harris Corporation pour la remise en état et la location du FASSET à Harris Corp. afin de faciliter la conception du modem EHF de cette entreprise. Le présent document décrit les réparations et les améliorations apportées pour Harris dans le cadre du contrat. La remise en état avait pour principal but de rétablir la fonctionnalité antérieure du FASSET, qui avait été entreposé de nombreuses années à RDDC Ottawa.

Le présent rapport porte sur les principales activités de remise en état, y compris les articles réparés, les modifications et améliorations, et les problèmes non résolus. Il constitue une référence technique renfermant les connaissances recueillies, ainsi qu'un registre et des explications sur les activités de remise en état. Les autres sujets abordés comprennent l'équipement d'essai, les procédures d'entretien, les leçons retenues et le travail à venir.

This page intentionally left blank.

Executive summary

The FASSET system is an extremely high frequency (EHF) satellite communications payload and ground terminal test bed that was developed in the 1990's for Defence R&D Canada – Ottawa (DRDC Ottawa). In July of 2002, DRDC Ottawa signed a contract with Harris Corporation that resulted in the refurbishment and delivery of FASSET to assist in the development of their EHF modem. The primary goal of the refurbishment was to reconstitute FASSET to the state of functionality that it possessed prior to being stored for many years at DRDC Ottawa.

This document details the repairs and improvements that were conducted for Harris in support of the contract. It is a technical reference of knowledge collected as well as a record and explanation of refurbishment events. A familiarity with principles of satellite communications is assumed including knowledge of the EHF standard MIL-STD-1582.

In this report three main areas of refurbishment are discussed in depth: items repaired, modifications and improvements, and unresolved problems. The section in the report dealing with items repaired is the largest, as it was one of the main areas of work. Descriptions of various maintenance procedures that were developed have been outlined. Key lessons learned during the refurbishment of FASSET are explained. These include issues with commercial-off-the-shelf products, long-term storage of hardware, and classified shipments to industry. Potential areas for future work have also been identified and described.

Most of the refurbishment tasks involved a preliminary investigation and analysis of the symptoms, followed by hardware repairs, software testing and debugging. Many complex problems were resolved to bring FASSET to a fully operational state. Expertise was required in the fields of high-speed digital processing, operating systems and development tools, spread spectrum signaling, complex synchronization techniques, radio frequency (RF) and analog circuits, and diagnostic methods and instrumentation.

Briefings and hands-on training were provided to Harris personnel once FASSET was delivered to the Harris site. This training covered the background, functionality and operation of FASSET. It should be noted that the current level of training at Harris is insufficient to provide for maintenance or improvements.

DRDC Ottawa is currently providing on-site and remote technical support to Harris Corporation during the lease of FASSET. If working knowledge and familiarity with FASSET operation are to be preserved, then an ongoing program of training and familiarization with FASSET operation and maintenance will be necessary. A difficulty with this program is that there is a need for continued physical access to the FASSET equipment in order to develop or maintain expertise.

FASSET is an EHF test bed that was delivered to Harris to facilitate the development of their modem. It meets or exceeds all of Harris' objectives. Refurbishment, delivery and training were completed by September 2003. Improvements in capability and reliability can be made by fixing spares and resolving outstanding issues identified in this document. In addition to meeting Harris' objectives, FASSET's performance was enhanced and new capabilities were introduced.

Liao, B., Addison, R., Lambert, J. D., 2004. FASSET Refurbishment. DRDC Ottawa TM 2004-134, Defence R&D Canada – Ottawa.

Sommaire

Le système FASSET est un banc d'essai sur terminal terrestre et à charge utile de communications par satellite aux fréquences extrêmement hautes (EHF) qui a été conçu dans les années 90 à l'intention de R & D pour la défense Canada – Ottawa (RDDC Ottawa). En juillet 2002, RDDC Ottawa a signé un contrat avec Harris Corporation pour la remise en état et la location du FASSET à Harris Corp. afin de faciliter la conception du modem EHF de cette entreprise. La remise en état visait à rétablir la fonctionnalité antérieure du FASSET, qui avait été entreposé de nombreuses années à RDDC Ottawa.

Le présent document décrit les réparations et les améliorations effectuées pour Harris dans le cadre du contrat. Il constitue une référence technique renfermant les connaissances recueillies, ainsi qu'un registre et des explications sur les activités de remise en état. La lecture du document nécessite une connaissance des principes de communications par satellite, y compris de la norme MIL-STD-1582 sur les EHF.

Le rapport examine en détails trois grands volets de la remise en état: les articles réparés, les modifications et améliorations, ainsi que les problèmes non résolus. Comme la majorité du travail a porté sur la réparation d'articles, la section sur ce volet est la plus longue. Les diverses procédures d'entretien sont décrites. Les principales leçons retenues pendant la remise en état du FASSET sont expliquées. Ces leçons portent sur les produits commerciaux, le stockage à long terme du matériel et les envois classifiés aux industries. En outre, les points pouvant éventuellement faire l'objet de travail sont cernés et décrits.

La majorité des tâches de remise en état ont nécessité une enquête préliminaire et une analyse des symptômes, puis une réparation du matériel, une mise à l'essai des logiciels et une mise au point. De nombreux problèmes complexes ont été résolus pour permettre la remise du FASSET en état de fonctionnement. Le travail a exigé des compétences spécialisées dans les domaines du traitement numérique haute vitesse, des systèmes d'exploitation et des outils logiciels, de la signalisation à étalement du spectre, des techniques de synchronisation complexe, des circuits analogues et des radiofréquences, ainsi que des méthodes diagnostiques et de l'instrumentation.

Une fois le FASSET remis à Harris, le personnel de l'entreprise a assisté à des séances d'information et suivi une formation pratique. Cette formation portait sur l'historique, la fonctionnalité et l'exploitation du FASSET. Veuillez noter que le niveau de formation qu'ont actuellement les employés de Harris est insuffisant pour leur permettre d'assurer la maintenance et les travaux d'amélioration.

RDDC Ottawa fournit actuellement un soutien technique à distance et sur place à Harris Corporation pendant la location du FASSET. Pour maintenir les connaissances de travail et la bonne connaissance de l'exploitation du FASSET, il faudra créer un programme permanent de formation sur le fonctionnement et l'entretien du FASSET. Cependant, cette initiative sera difficile à mettre en œuvre, car un accès physique continu à l'équipement du FASSET est nécessaire pour développer ou maintenir l'expertise.

Le système FASSET est un banc d'essai EHF loué à Harris pour l'aider à concevoir son modem. Le système comble ou surpasse tous les objectifs de l'entreprise. Les activités de remise en état, de location et de formation ont été achevées en septembre 2003. La capacité et la fiabilité peuvent être améliorés grâce à la réparation de pièces de remplacement et à la résolution des problèmes non réglés cernés dans le présent document. Outre avoir satisfait aux objectifs de Harris, le FASSET a été amélioré et a obtenu de nouvelles capacités.

Liao, B., Addison, R., Lambert, J. D., 2004. FASSET Refurbishment. RDDC Ottawa TM 2004-134, R & D pour la défense Canada – Ottawa.

Table of contents

Abstract i

Executive summary iii

Sommaire iv

Table of contents v

List of figures xi

List of tables xii

1. Introduction 1

2. Background 2

2.1 FASSET Description 2

2.1.1 A Brief History of FASSET 2

2.1.2 FASSET Overview 3

2.1.3 EHF Waveform Description 5

2.1.4 Factors Affecting FASSET Capabilities 6

2.1.4.1 Availability of EHF Standards 6

2.1.4.2 Contract Focus on Interoperability 6

2.1.4.3 Interoperability Modifications at DRDC Ottawa 6

2.1.5 Lincoln Lab Interoperability Trial 7

2.2 Harris Corporation Involvement 8

2.2.1 Harris Requirements 8

2.2.2 Refurbishment for Harris Corporation 9

2.2.2.1 System 9

2.2.2.2 Payload 10

2.2.2.3 Ground Terminal 11

2.2.3 Post-delivery Service at Harris Corporation 11

2.3 Current Status 12

2.3.1 Capabilities and Limitations 12

2.3.2 Spares 14

2.3.3 Work Remaining 14

2.4 Test Equipment Used 15
2.4.1 Included with FASSET 15
2.4.2 Essential for FASSET 15
2.4.3 Required for Payload Loopback Testing 15
2.4.4 Useful During Normal Operation 16
2.4.5 Useful During Debugging 16
2.5 Sequence of Refurbishment 17
2.5.1 November 2001 – August 2002 17
2.5.2 August 2002 – September 2002 18
2.5.3 September 2002 – December 2002 18
2.5.4 December 2002 – February 2003 19
2.5.5 February 2003 – September 2003 20

3. FASSET Items Repaired 21
3.1 System 21
3.1.1 User Data Patch Panel Connectors 21
3.1.2 Bit Error Rate Test Set – Failed Self Test 21
3.1.3 Excessive Uplink Attenuation 22
3.2 Payload 22
3.2.1 Network Transceiver Faults 23
3.2.2 Transmitter Subsystem Not Responding 25
3.2.3 Receiver Attenuator Malfunction 26
3.2.4 Timing Control Subsystem Power Supply Failures 27
3.2.5 Timing Control Subsystem Payload Controller Interface Failure 28
3.2.6 LDR Baseband Subsystem – TT&C Not Working 32
3.2.7 LDR Baseband Subsystem – C2 Messages Not Working 33
3.2.8 Downlink Delay Module – Excessive Bit Error Rate 34
3.2.9 TT&C Loopback Modification and Connector Repairs 35
3.2.10 Downlink Expander Fault 35
3.2.11 Uplink Expander Fault 36
3.2.11.1 Background 36
3.2.11.2 Fault Analysis 38
3.2.11.3 Post-Repair Analysis 39
3.3 Ground Terminal 40
3.3.1 Replacement Battery Backed-up RAM for Single Board Computers 40

3.3.2 Hard Disk Corruption 40
3.3.3 X-terminal Failure 41
3.3.4 Hyperbus Interconnect – Short Circuit 43
3.3.5 TRANSEM Uplink Attenuator Failure 45
3.3.6 Frequency Reference Generator – Out of Lock 45

4. Modifications and Improvements 47
4.1 System 47
4.1.1 Standard Configuration Files Implemented 47
4.2 Payload 47
4.2.1 Configuration File Format Modified 48
4.2.2 External Loopback Capability Added 48
4.2.3 Timing Control Subsystem Power Supplies – Diagnostic Monitoring Implemented 49
4.2.4 Support for Multiple User Data Accesses Added 50
4.2.5 Uplink to Downlink Mapping Enhanced for TT&C Messaging 51
4.2.6 Uplink to Downlink Mapping Enhanced for C1 Messaging 52
4.2.7 Additional C3 Message Sources Implemented 52
4.3 Ground Terminal 53
4.3.1 VME Development System Built 53
4.3.2 Removable Hard Disks Installed 54

5. Problems Identified But Not Resolved 55
5.1 Line Printer Not Printing 55
5.2 Ground Terminal DDS Controller Causing Downlink Sync Failure 55
5.3 Remaining Issues of Excessive Uplink Attenuation 56
5.4 Excessive Bit Errors in Uncoded Multi-user Accesses 56
5.5 Spares 57
5.6 Incorrect Wiring in Payload Controller Interface 58

6. Maintenance Procedures 59
6.1 Changing Passwords 59
6.1.1 Payload Password 59
6.1.2 Ground Terminal Passwords 59
6.2 Modification of Payload Computer Application 60
6.2.1 Regeneration of Payload Computer Executable 60

6.2.2 Regeneration of Screen Library 60
6.2.3 Modification of Payload Screens 61
6.3 Modification of LDR Symbol Processor Software 62
6.3.1 Regeneration of LDR Symbol Processor Kernel Firmware 62
6.3.2 Regeneration of LDR Symbol Processor Download Software 63
6.4 Ground Terminal Tape Backup and Restore 64
6.4.1 Installation of Tape Drive 64
6.4.2 Backup 65
6.4.2.1 Backup Root 65
6.4.2.2 Backup Usr 65
6.4.3 Restore 65
6.4.3.1 Install Base Operating System 65
6.4.3.2 Restore Root 66
6.4.3.3 Restore Usr 66
6.4.3.4 Optional Post-Restore Steps 66

7. Lessons Learned 68
7.1 Reliability Issues Caused by Shipping 68
7.2 Transfer of Classified Documents to Industry 69
7.3 Issues Associated with Long-term Storage 69
7.4 Long-term Availability of COTS Not Guaranteed 69

8. Conclusions and Future Work 71
8.1 Summary 71
8.2 Conclusions 71
8.3 Future Work 72

9. References 73

Annex A VME Bus Trace Information 75
A.1 Payload 75
A.1.1 Debugging Information 75
A.1.1.1 LDR Data Formatter Routine 75
A.1.1.2 Configure Symbol Processor Routine 76
A.1.1.3 Decode C2 Message Routine 76
A.1.1.4 Process C2 Message Routine 77
A.1.2 Additional Information 77

A.2 Ground Terminal 78
A.2.1 Debugging Information 78
A.2.1.1 Calculate RMS Frequency Track Error Routine 78
A.2.1.2 Calculate RMS Time Track Error Routine 79
A.2.1.3 Downlink Frame Interrupt Service Routine 79
A.2.1.4 Downlink Tracking Routine 80
A.2.1.5 Perform LDR Reset Routine 80
A.2.1.6 Process C2 Data Routine 80
A.2.1.7 Receive C2 Data Routine 81
A.2.1.8 Send Synchronization Status Routine 81
A.2.1.9 Synchronization Routine 81
A.2.1.10 Time and Frequency Control Routine 81
A.2.1.11 Address and Value Cross Reference Table 83
A.2.2 Additional Information 84
A.2.2.1 Floating Point Format 84
A.2.2.2 Interrupts 84
A.2.2.3 Board Addresses 85
A.2.2.4 MCP Inter-processor Communications Buffers 86
A.2.2.5 DLP to MCP Buffer 86

Annex B Summary of Payload File Changes 89
B.1 Firmware 89
B.2 Payload Computer (Unclassified disk) 89
B.3 Red Disk (Classified disk) 91

Annex C Test Matrices 93
C.1 Standard Test Matrix 93
C.2 User Data (C0) Test Matrix 93
C.3 Secondary User Data (C1) Test Matrix 94
C.4 User Data to/from Payload (TT&C) Test Matrix 94
C.5 Multi-user Data Test Matrix 95
C.6 Uplink Access Control (C2) Test Matrix 95
C.7 Downlink Access Control (C3) Test Matrix 95

Annex D Custom Software Development Tools 96
D.1 ABStoBIN 96

D.2 CountFields 97

List of acronyms 98

List of figures

Figure 1 FASSET payload and ground terminal 3

Figure 2 Payload network connections..... 24

Figure 3 Payload network bulkhead connectors..... 25

Figure 4 Payload Transmitter Subsystem DIU reset circuit 26

Figure 5 Extract from original PCI EPROM wiring diagram..... 30

Figure 6 Modified EPROM diagram showing corrected labels and A14 grounded..... 31

Figure 7 EPROM diagram with recommended final modifications 32

Figure 8 Payload Frequency Synthesizer connection diagram..... 37

Figure 9 Ground terminal system layout 42

Figure 10 Physical layout of the Hyperbus (front view) 44

Figure 11 Payload TCS power supply monitor 50

List of tables

Table 1 Mandatory requirements from Harris Corporation 9

Table 2 Desired requirements from Harris Corporation 9

Table 3 FASSET capabilities and limitations 14

Table 4 Power supplies in the TCS 27

Table 5 Field counts in payload screens 61

Table 6 LDR data formatter addresses 76

Table 7 Configure symbol processor addresses 76

Table 8 Decode C2 message addresses 76

Table 9 Process C2 message addresses 77

Table 10 Decoder endian format swap addresses 77

Table 11 Frequency track error addresses 78

Table 12 Time track error addresses 79

Table 13 Downlink frame interrupt addresses 80

Table 14 Downlink tracking addresses 80

Table 15 LDR reset addresses 80

Table 16 Process C2 data addresses 80

Table 17 Receive C2 data addresses 81

Table 18 Send synchronization status addresses 81

Table 19 Synchronization addresses 81

Table 20 Time and frequency control addresses 82

Table 21 Address and value cross reference table 83

Table 22 Ground terminal interrupts 85

Table 23 Ground terminal VME addresses 85

Table 24 Modem Control Processor buffer addresses 86

Table 25 Downlink processor to MCP addresses 88

Table 26 LDR Symbol Processor firmware 89

Table 27 Modified payload unclassified files 90

Table 28 Modification to payload user interface screens 91

Table 29 Modified payload classified files 92

Table 30 Standard test matrix 93

Table 31 User data (C0) test matrix 94

Table 32 Secondary user data (C1) test matrix 94

Table 33 Test matrix for TT&C 94

Table 34 Multi-user data test matrix 95

Table 35 Uplink access control (C2) test matrix 95

Table 36 Downlink access control (C3) test matrix 95

This page intentionally left blank.

1. Introduction

The FASSET (Functional Advanced Development Model of an EHF Satellite Communications System for Evaluation and Test) system is an extremely high frequency (EHF) satellite communications (SATCOM) payload and ground terminal test bed that was developed in the early 1990's for Defence R&D Canada – Ottawa (DRDC Ottawa). In July of 2002, DRDC Ottawa signed a contract with Harris Corporation that resulted in the refurbishment and delivery of FASSET to their facility in Melbourne, Florida. The objective of the contract was to enable Harris to use the FASSET system to assist in the development of their EHF modem.

This document details the repairs and improvements that were conducted for Harris in support of the contract. It is a technical reference of knowledge collected as well as a record and explanation of refurbishment events. A familiarity with SATCOM principles is assumed including knowledge of the EHF standard MIL-STD-1582 [1, 2].

The primary goal of the refurbishment was to reconstitute FASSET to the state of functionality that it possessed prior to being stored for many years at DRDC Ottawa. With refurbishment completed by March 2003, the system was delivered to Harris, and after some follow-up troubleshooting FASSET became a functioning tool that was available for their exploitation. Briefings and hands-on training documented in [3] were provided to Harris personnel covering FASSET's background, functionality, and operation.

An analysis and review of significant refurbishment events are discussed in the following document. The subsections may be read independently, and as such necessary background information has sometimes been repeated to aid the reader.

The three main areas of refurbishment, items repaired, improvements, and unresolved problems, are covered in depth. A listing of test equipment used during refurbishment is provided, along with descriptions of various maintenance procedures that were developed. A list of lessons learned and potential areas for future work are also outlined.

The section in the report covering items repaired is the largest, as it was one of the main areas of refurbishment work. Significant effort was invested in fixing critical faults in the Timing Control Subsystem (TCS). The other major section in this report covers modifications and improvements that were implemented in FASSET during refurbishment. Although some minor work was done in the ground terminal, the main improvements focused on payload software.

2. Background

This section describes the history of FASSET, and the process leading up to the delivery of the system to Harris Corporation. Background information is provided to give some explanation of the scope and detail of the refurbishment topics that follow.

The first subsection below gives a brief description of FASSET's functionality, its capabilities, and its ability to produce an EHF waveform that is interoperable with the EHF SATCOM data link standard entitled MIL-STD-1582B. Harris' requirements for leasing the FASSET system, and the motivation for refurbishing FASSET are discussed. An update on FASSET's current status is also provided, including discussion of remaining spares and future work. The test equipment required for operating and troubleshooting FASSET is listed and categorized into groups depending on the testing requirements. Finally, a summary of significant refurbishment events is provided in chronological order to give some context to the refurbishment topics that follow in Sections 3, 4, and 5.

2.1 FASSET Description

The history of the FASSET test bed will be presented followed by a description of the main components. Some details of the EHF waveform supported by FASSET will also be provided. Finally, the factors affecting FASSET capabilities will be examined.

2.1.1 A Brief History of FASSET

In 1986, the Department of National Defence (DND) started a major project called EHF Satellite Communications (SATCOM) for approximately $48 million. It involved research and development to allow the military to evaluate EHF satellite communications and to further advance Canadian capability in that technology. The project included antenna research and development of an EHF test bed designated FASSET (Functional Advanced Development Model of an EHF Military Satellite Communications System for Evaluation and Test).

The aim of the FASSET contract was to provide a test bed that could be used to develop, evaluate and test the EHF satellite technology and system concepts for an operational EHF SATCOM system. It also aimed to provide DND with a facility for the ongoing evaluation, test and demonstration of EHF SATCOM.

A consortium led by MPR Teltech in Burnaby, British Columbia won the FASSET contract. It included Com Dev of Cambridge, Ontario and Raytheon Canada of Waterloo, Ontario. FASSET was delivered in 1996 to Defence Research Establishment Ottawa (DREO), now called Defence R&D Canada - Ottawa (DRDC Ottawa). The former name is found on most of the FASSET documentation.

FASSET was delivered with unresolved problems. Because money was short at the end of the FASSET contract, some problems had not been fixed and some features had never been implemented or tested. During the next year at DRDC Ottawa, FASSET was fixed and modified for interoperability tests. In 1997, FASSET was transported to Boston where interoperability was successfully demonstrated with a terminal at the Massachusetts Institute of Technology (MIT) Lincoln Lab. DRDC Ottawa had no further research projects using FASSET so it was subsequently packed in boxes and stored.

In 2001 Harris Corporation, as part of the Space Based Infrared Systems (SBIRS) project, became responsible for developing an EHF low data rate (LDR) compatible modem. Harris had no test equipment for a frequency-hopped waveform and was searching for a test bed to reduce

the risk associated with developing their modem. They first approached MIT Lincoln Lab who had built EHF test beds in the past, but were not able to secure a test bed in the required timeframe. Lincoln Lab suggested that the Canadian test bed had proved interoperable and might be a viable option. Discussion and negotiations between Harris Corporation and DRDC Ottawa ensued. In the summer of 2002, Harris contracted with DRDC Ottawa for the refurbishment, delivery, lease, post-delivery training and service of FASSET.

After five years in storage, FASSET had aged and degraded. The refurbishment process took longer than anticipated due to extensive problems encountered. Eventually, after refurbishment and enhancements, FASSET was delivered to Melbourne, Florida in early 2003. Several visits were made by the FASSET team to fix problems and train Harris personnel.

At the end of the summer of 2003, the SBIRS requirement for an EHF low data rate (LDR) compatible modem was cancelled, and the SBIRS project had no longer any need for an EHF test bed. The lease was nevertheless continued because another EHF project started at Harris and would probably require a test bed such as FASSET.

2.1.2 FASSET Overview

As delivered to DRDC Ottawa, the FASSET test bed (see Figure 1) consisted of four main components: two ground terminals, a payload, and an overall system controller called the Monitor and Alarm Controller (MAC). The payload and ground terminals were inter-connected at radio frequency (RF) via waveguide. The MAC controlled the payload and ground terminals using a Transmission Control Protocol/Internet Protocol (TCP/IP) network over thinwire coaxial cable.

Figure 1 FASSET payload and ground terminal

Due to the shortage of money near the end of the contract, not all the planned functionality was delivered. The second ground terminal was never completed or tested. The MAC control of the payload and remaining ground terminal caused excessive load on the ground terminal processor and emulations occasionally failed. To alleviate this problem, the MAC was discarded and local control was used on the payload and ground terminal.

While it was originally planned that the FASSET system be able to support medium data rate (MDR) and low data rate (LDR) communications, limited resources forced the cancellation of the MDR portion.

When running, the FASSET system was intended to allow the selection of three different operation modes. The most complex is the Global mode where the payload emulates a satellite within a constellation. It includes the ability to alter the path loss, propagation delay and doppler according to the emulated orbit. Physical mode is a mid-complexity mode, where the initial path loss, initial propagation delay and range rate are constant and specified at the beginning of the emulation. The simplest mode is Electrical mode where the attenuator settings, delay and frequency offsets are specified directly. Limited resources caused the functionality to be implemented only for Electrical mode and the other two modes were never completed.

In addition to the main components, FASSET also includes a bit-error-rate test set, RS-232 switch, patch panel and serial interconnects to data sources/sinks on the ground terminal and payload. These are used to generate pseudorandom data for the ground terminal to be transmitted to the payload, processed and then sent back to the ground terminal for error-rate measurements. The RS-232 switch allows different measurements to be made without re-cabling. It should be noted that the payload has a special user data port called the Tracking Telemetry and Command (TT&C) port that can also be selected to input downlink and output uplink data at the payload.

FASSET includes several test ports. There are logic signals that can be monitored on both the payload and ground terminal to determine uplink and downlink hop and framing clocks. On the payload, there is also a downlink RF monitor port and an uplink injection port. These can be used during debugging but are also used when the payload is being tested in loopback mode. The ground terminal also provides ports for injecting a signal on the downlink and monitoring of the uplink.

To start an emulation, configuration files are first loaded in both the payload and ground terminal by means of their user interfaces. Then a common time reference is injected into both. Finally the payload and ground terminals are commanded to start processing.

At the start of an emulation, the ground terminal normally has a small offset in its clocks relative to the payload. The ground terminal starts downlink synchronization, and then tracks the downlink while performing uplink synchronization. When completely synchronized, the ground terminal can send uplink data to the payload and bit-error-rate measurements can be made on the resulting downlink data.

FASSET was delivered with extensive hard copy documentation. The main difficulty is finding the required information within a vast collection of manuals. While the occasional error has been found, most of the documentation is accurate. Most remaining documentation errors exist because information is not up to date (for example, a schematic is given for a previous version of a custom printed circuit board).

The source code is available for all software, but some details are classified making it difficult to examine. The unclassified portion of the payload software is also included on the payload control computer. For some programmable logic devices, there is no documentation, source code or device image. This can cause problems if a device fails and a new one cannot be reprogrammed.

Logbooks are available for both the interoperability modifications and the refurbishment process [4, 5]. They are divided into multiple volumes (2 each) that consist of unclassified plots and notes, classified plots and classified notes. There is also a binder of resources which includes

useful information that was collected during the refurbishment process. Once FASSET was setup at the Harris facilities, a new logbook [6] was started.

2.1.3 EHF Waveform Description

The United States (US) EHF LDR Specification [1, 2] covers the EHF band for uplink (44.5 GHz with a 2 GHz bandwidth) and the super high frequency (SHF) band on the downlink (20.7 GHz with a 1 GHz bandwidth). It describes a complex waveform with many features to make it robust in normal operations and when an enemy is trying to disrupt communications. Most importantly, it is a spread-spectrum frequency-hopped waveform. Frequency-hopping is a technique where the RF centre frequency changes rapidly according to a pseudo-random pattern known by each end of the communications link (and not by the enemy). This pattern is generated by a transmission security (TRANSEC) device such as the KGV-11.

The greatest difficulty with frequency hopping is ensuring that both ends of the link are synchronized in time. The EHF satellite keeps the master time and all terminals must synchronize to it. Synchronization is further complicated by the time varying propagation delay between satellite and payload, which requires the ground terminal to maintain separate timing for the uplink and downlink. There are synchronization aids defined in the waveform to allow the terminal to synchronize.

The uplink structure uses a frame with a combination of frequency division multiple access (FDMA) and time division multiple access (TDMA) whereas the downlink structure only uses time division multiplexing (TDM). All data communications, synchronization aids, and access control communications use these structures.

While frequency hopping is the primary method of reducing enemy threats, there are other manipulations in the waveform to aid in transmission security. These TRANSEC manipulations include: time permutation (changing the order of bits within a frame), frequency permutation (changing the channel used within a frame), chip rotation (changing the bit order when multiple bits are sent on a single hop) and cover (a technique used to establish privilege for the access control protocol). For the real system, the KGV-11 is used as a source of TRANSEC patterns. For testing and debugging, other simpler or less sensitive patterns have been developed.

Downlink synchronization is accomplished using synchronization hops that are generated by the payload. These special hops have a known modulation that is expected and measured by the ground terminal. To achieve uplink synchronization the ground terminal sends special synchronization probes on the uplink. These probes are measured by the payload, which then responds with a synchronization response through an Acquisition Response Order Wire (AROW) message on the downlink. Both synchronization hops and probes are used for the ground terminal's initial acquisition of payload timing, and for tracking that timing.

Once synchronization is achieved and the ground terminal has an adequate estimate of payload timing, data communications can begin. The LDR EHF communications standard supports both primary user data channels (designated C0) at rates of 75 to 2400 b/s and secondary user data channels (designated C1) at rates of 75 to 300 b/s. Message traffic sent from the ground terminal to the payload for access control is called C2 data. C3 messages contain access control data that is transmitted from the payload to the ground terminal. C2 and C3 messaging occurs at rates of 75 to 300 b/s.

Data communications can be accomplished with a variety of modulations including differential phase shift keying (DPSK) and frequency shift keying (FSK). For the downlink, there are many modes of DPSK allowing for trade-off between robustness and data rate. For the uplink and for the most robust communications on the downlink, FSK modulation using various rates of symbol diversity (repetition) is used. Rate 1/2 convolutional forward error correction (FEC) coding, along with various interleavers, is provided to improve performance.

2.1.4 Factors Affecting FASSET Capabilities

In this section, the factors that affect FASSET capabilities will be detailed. Originally, FASSET was to include an overall controller, a lab ground terminal, a ruggedized ground terminal and a payload. It was to have supported both LDR and MDR communications. Because of availability of information and resource shortfalls these requirements were reduced prior to delivery to DRDC Ottawa in 1996. After the delivery, resources were focused on resolving problems in FASSET in an effort to enable and improve interoperability testing at Lincoln Lab.

2.1.4.1 Availability of EHF Standards

During the specification phase of FASSET, the US LDR data link standard was not available to Canada. Instead, Canadian EHF LDR and MDR data link standards were developed for use in FASSET. Later, while the contract was underway, through international agreements, the US LDR data link standard [1] was made available. Unfortunately, only the main body was provided, and the appendices containing the access control protocol were not included.

The FASSET contract was modified to be compatible with [1] though not without cost. The modification set the project back at least one year and took all of the project contingency funds. In addition, to focus more resources on the LDR features, the MDR portion and the ruggedized ground terminal were dropped.

Later, the appendices and the next version of the standard [2] were made available to Canada, but it was too late to change the FASSET contract specification.

2.1.4.2 Contract Focus on Interoperability

The US LDR data link standard was made available to Canada in part because of cooperation in military research in EHF satellite communications. The military was also concerned that any satellite communications system be able to communicate with the US – a major military ally. Both of these reasons provided strong incentives to prove interoperability of the FASSET test bed with US systems.

As the FASSET contract was finishing and funding was nearly depleted, the work had to be prioritized because not all features and problems could be addressed with the available resources. The primary concern was interoperability testing (and all features necessary to accomplish this). Consequently, areas not essential for interoperability were not completed, debugged or tested. Key components that were dropped were the overall test bed controller called the Monitor and Alarm Controller (MAC), the second ground terminal, and the MDR mode of operation.

2.1.4.3 Interoperability Modifications at DRDC Ottawa

Interoperability between Canadian satellite communications systems and US systems is concerned primarily with user data communications. This is true for the FASSET test bed as well. Following delivery of FASSET from the contractors, the fixes and modifications done at DRDC Ottawa concentrated on interoperable data communications. Specifically, the modifications consisted of the features necessary to support synchronization (a prerequisite for data communications), data communications, flexible uplink and downlink assignments and compatible cryptographic equipment.

When FASSET was formally delivered to DRDC Ottawa, there were some important shortcomings. The MDR mode was never completed, and only one of the ground terminals was working. The MAC was discarded, and only local control of the payload and the ground terminal

was used to set up and run the emulations. Almost all tests were run using a 2400 b/s user data rate, and most other options had not been explored or even debugged.

Before enhancements were made, it was necessary to fix some problems that existed when FASSET was delivered to DRDC Ottawa. The payload frequency reference failed so it was replaced and the frequency reference chain was simplified. Some cables and interconnects were replaced to improve reliability. A programmable logic device also failed on one of the payload's custom boards and had to be replaced.

When FASSET was delivered, the payload used a fixed mapping of access parameters linking uplink to downlink communications. The modulation type, access mode and map type on the uplink determined the downlink hop assignments and modulation. This was considered too inflexible for interoperability because it did not allow specification of downlink hops nor downlink modulation type. Enhancements were added to FASSET to allow independent specification of uplink and downlink access parameters.

For testing with the Lincoln Lab test bed, a special TRANSEC pattern was developed called Magic 5. The effects of this pattern on all the TRANSEC manipulations of the waveform were documented in a Lincoln Lab report [7]. The pattern was used in place of the KGV-11 TRANSEC devices to simplify debugging. It was determined that this would be very useful for interoperability testing as well. Lincoln Lab had this pattern built into their test bed, but DRDC Ottawa was unable to do the same for FASSET. Instead, an intercept box was developed that was placed between FASSET and the TRANSEC simulator and allowed this pattern to be injected instead of the simulator data.

Once this box was completed, it was realized that the ground terminal could not synchronize using the Magic 5 pattern because of a known problem. A similar pattern called Rotate 6 was developed that could be used with the FASSET ground terminal. The effect of the Rotate 6 pattern on all of the TRANSEC manipulations of the waveform was documented in [8] with explanation notes on page 80 of [3] to allow easy debugging with Rotate 6.

To support the flexible mapping, some of the payload user interface screens were modified. At the same time, others were reformatted and hints added to improve usability. Some default values used by the payload and ground terminal were modified to be more appropriate for DRDC Ottawa usage.

A test matrix covering the desired interoperability tests was created in conjunction with Lincoln Lab (this matrix was very similar to the User Data C0 Test Matrix in Annex C). The configuration files for the payload and ground terminal were developed and tested.

2.1.5 Lincoln Lab Interoperability Trial

Both the United Kingdom (UK) and Canada were involved in cooperating with the US in EHF military satellite communications. The EHF Interoperability Working Group and TTCP C3I TP-6 contacts provided mechanisms and motivation for interoperability trials.

The UK test bed was first tested in 1995 at Defence Evaluation and Research Agency (DERA) Defford (now QinetiQ) with the Wright Patterson airborne terminal. A year later, in 1996, it was successfully tested at the Advanced MilSatCom Technologies laboratory at MIT Lincoln Lab in Boston with their test bed.

In 1997, FASSET was moved to Lincoln Lab for interoperability trials. The testing was to include various combinations of the FASSET payload, FASSET ground terminal, US payload and US ground terminal. The sequence of testing was downlink synchronization, uplink synchronization, 2400 b/s user data, other data rates, and user data bit-error-rate performance. The 2400 b/s user data was to be at various combinations of coding, diversity, interleaving and TRANSEC manipulations. There were no access control tests because FASSET did not support access control protocols.

For the tests, the KGV-11 TRANSEC devices were used. This was because Lincoln Lab did not have the Rotate 6 pattern and the FASSET ground terminal could not use the Magic 5. In addition, the US and FASSET TRANSEC simulators were found to be incompatible.

A common time reference was obtained using a trigger pulse, generated by the FASSET Calibrated Time-of-Day (CTOD) Injector, which was provided to both FASSET and the US test bed. Access control was avoided by turning off that feature in FASSET and the US test bed prior to emulation start.

While debugging uplink synchronization, it was determined that the FASSET equipment expected AROW (the synchronization probe response) messages one frame earlier than the US. DRDC Ottawa did not have the ability to change the FASSET ground terminal so the FASSET test bed could not be changed. Instead, the US modified their ground terminal to match the FASSET interpretation. All interoperability tests were done only with the FASSET payload, FASSET ground terminal and US ground terminal.

Downlink synchronization and tracking worked. Additional synchronization hops were enabled by the FASSET payload to match expectations of the US ground terminal. This is because the US ground terminal supported one set of assignments, which are a specific implementation of the more general standard [2]. Uplink synchronization worked after modifications were done to the US ground terminal to accommodate the AROW frame expectation.

User data communications worked well at 300 b/s including various combinations of modulation, coding, and interleaving. A similar frame numbering problem with the AROWs seemed to affect the data rates from 75 to 300 b/s. With these data rates, some combinations of interleaving and coding failed.

Randomly generated C3 messages with proper cyclic redundancy check (CRC) fields were sent by the FASSET payload to the US ground terminal. These messages passed the ground terminal's CRC validation indicating that the forward error correction (FEC) and CRC portion of the access control messages were interoperable.

The final test, just prior to departure from Boston, was the bit-error-rate performance. The performance was poor, but was within 4 dB of theoretical. Several factors were identified as likely contributors to the poor results. The noise source was not properly calibrated, and the effects of a sub-optimal surface acoustic wave (SAW) block demodulator in the payload were never quantified. Furthermore, the test was conducted at the last minute, and it was not repeated.

Overall, the tests were considered successful. They proved the interoperability of FASSET with respect to synchronization, data communications, TRANSEC manipulations and part of the access control protocol.

2.2 Harris Corporation Involvement

In 2001 Harris Corporation, as part of the Space Based Infrared Systems (SBIRS) project, became responsible for developing an EHF LDR compatible modem. Harris contracted with DRDC Ottawa for the use of FASSET as an EHF test bed to reduce the risk associated with the development of their modem. The contract included refurbishment of FASSET, delivery to their site, lease, training and servicing of FASSET.

2.2.1 Harris Requirements

During refurbishment, many problems were encountered and the complexity of FASSET meant that not all could be fixed. The Harris team developed two groups of requirements. The first group contained the features required to use FASSET as a test bed. The second group of requirements contained the features that were desirable once the "must haves" were addressed.

Table 1 below lists the "must have" requirements, and Table 2 lists the desired requirements. The columns show the requirements and the performance goals for both the payload and ground terminal in each table. After post-delivery repairs, all of the performance goals were met or exceeded.

#	Requirement	Payload	Ground Terminal
1	Downlink synchronization	Generate coarse and fine synchronization hops	Acquire and track synchronization hops
2	Uplink synchronization	Process probes and generate AROW	Generate coarse and fine time probes at 2400 b/s
3	Data communications	Run test emulations with filenames: C0DATA01 to C0DATA13 with TRANSEC on at 75 and 2400 b/s	Run test emulations with filenames: C0DATA01 to C0DATA13 with TRANSEC on at 75 and 2400 b/s
4	Uplink Access Control	Process C2 messages	Generate C2 messages (least robust, long and short)
5	Downlink Access Control	Generate C3 messages, one per channel enabled	Process C3 messages
6	TRANSEC	Demonstrate selective bypass for one test case	Demonstrate selective bypass for one test case

Table 1 Mandatory requirements from Harris Corporation

#	Desired	Demonstration
1	TT&C Communications	Two test cases with TT&C in both directions
2	Payload Loopback	C0 data and access control
3	Bit Error Rate Performance	One test case of 10^{-5} without coding with TRANSEC on

Table 2 Desired requirements from Harris Corporation

2.2.2 Refurbishment for Harris Corporation

Once the contract was in place, the refurbishment for Harris Corporation started. The intent was to get the equipment working quickly by fixing minor problems and failures caused by FASSET being in storage for five years. However, many problems were discovered resulting in a more substantial effort. The ground terminal would not boot up at all. Several problems had been lurking for years and only surfaced during the refurbishment. Power supplies had degraded affecting their reliability and reset circuits. Commercial-off-the-shelf (COTS) spares were unavailable for the most part due to their age and obsolescence. Corporate memory had deteriorated over the five years. Still, these problems were surmounted and FASSET was made to work, and even enhanced.

In the following subsections, work done to the overall system is described first. A summary of the payload work follows, and finally work that was done to the ground terminal is detailed.

2.2.2.1 System

The RS-232 switch and patch panel, which routs the data source and sink to the various user data ports, was the origin of several problems. One channel of the switch failed and several

bad connections were found in the rear of the switch. The channel module was replaced, and the wiring was fixed and secured.

In general, during refurbishment failed parts and interconnects were replaced with laboratory supplies and borrowed parts. Later in refurbishment, these replacements were procured and the laboratory parts restocked.

Prior to shipping FASSET to Harris' facilities, the ground terminal's Transceiver Emulator (TRANSEM) was moved and the waveguide interconnects were redone to make space available for Harris' modem in the rack. The TRANSEM is a hardware component of the RF front end of the ground terminal, which includes the final upconversion stage and the first downconversion stage.

Sophisticated timing triggers were developed during debugging. These triggers allowed examination of the hopped spectrum and time-based signals on the spectrum analyzer using the logic analyzer as the trigger source. A detailed explanation of these timing triggers is given on pages 82-85 of [3].

Problems with reliability of the system forced a close examination of the power levels. It was determined that some of the programmable attenuators were not responding properly and they were bypassed (in the payload receiver, see Section 3.2.3). The minimum values for proper synchronization and communications were obtained by adjusting manual attenuators. Reasonable levels were selected above the minimums to allow for greater reliability. Instructions for measuring these power levels are given on pages 71-72 of [3].

The original test configuration files were reconstructed for the ground terminal. Based on Harris requirements, new configuration files were developed in the test matrix (see Annex C). All tests were run multiple times using different TRANSEC sources (Rotate 6, TRANSEC simulator and KGV-11).

2.2.2.2 Payload

One of the most common problems during refurbishment was the failure of aging local area network (LAN) transceivers used on an internal network in the payload to communicate between subsystems. Consultation with a Harris LAN expert suggested that the coaxial grounding technique designed into the payload is responsible for the poor reliability of these adaptors. Nevertheless, many were changed and even more spares were procured.

A source of intermittent problems in the payload was poor interconnects. The connectors and wiring harnesses had been plugged, unplugged and torqued beyond their lifetime and occasionally failed. All of the digital interconnects and back panel connectors in the payload racks were examined and touched-up. Some cables still had problems so replacements were procured and installed.

Originally, a monitor port was provided on the Timing Control Subsystem (TCS) to monitor the hop and frame clocks. The signals on this port were found to be noisy, and were not adequate as timing triggers in a logic analyzer. It was determined that they were connected (with no buffer) directly to an internal bus. Because of the danger of probing and their lack of utility, these ports were blocked with covers and no longer used. Instead, inter-subsystem cables were intercepted to extract an RS-422 version of the same signals.

At one point, emulations were conducted and significant drift was found between the ground terminal and payload timing. A close examination of the payload frequency reference showed that it was not stable. Its power supply was replaced and the frequency calibration was verified.

Degraded power supplies caused the Transmitter Subsystem to fail because the reset circuit timing was too short. Reliability of power-up was improved by extending the reset time.

The Timing Control Subsystem, the most densely packed subsystem, required most of the refurbishment time and effort. The power supplies failed several times and were replaced. A

failed portion of the Frequency Synthesizer (the Downlink Expander) was replaced with its spare. Because the Downlink Expander is so critical, an attempt was made to fix the failed expander, without success. Bit-error-rate performance of the payload was improved significantly when the Downlink Delay Module was replaced because of an internal fault.

The payload LDR Baseband Subsystem also had much work done to it. Software had to be modified to fix and enhance support for the multiple C0 data mode, to fix support for C1 data, to fix and provide flexible mapping for TT&C data, to fix and enhance C2 messaging, to enable fixed C3 messages, and to fix payload loopback testing. In addition, the TT&C start-up plug was replaced with a simpler and more meaningful loopback switch. Usage of this switch is described on page 75 of [3].

On the payload computer, the user interface was modified to fix several screens, to add hints, and to support the enhancements mentioned in the previous paragraph. Also, both the classified and unclassified payload hard disks were backed-up and documented.

2.2.2.3 Ground Terminal

Initially, the ground terminal failed to start. A long debugging process determined that this was caused by a failed alternating current (AC) switch, a failed processor board, a corrupted hard disk, and expired battery backed-up memory on the processor boards. The battery backed-up memory components had expired due to age, and ambiguity existed in the documentation as to the values that should be used in their replacement. These values were determined by inspection of the software and rationalization of those results with the documentation. Once the hard disk had been restored and the processor board replaced, the ground terminal worked properly.

In addition to restoring the ground terminal's hard disk, spare hard disks had to be acquired through purchases from used equipment auctions. The processor board firmware required a special version of the disk's erasable programmable read only memory (EPROM) that in some cases had to be reprogrammed. All hard disks were put in removable caddies to aid in handling (security) and for swapping convenience. Multiple backups were performed on the restored ground terminal hard disk.

Many processor boards failed and had to be replaced. Spare processor boards were acquired in a similar manner to the hard disks. These were used parts and in some cases were not working when received. Parts were swapped and boards were cannibalized to obtain as many working boards as possible.

Early on, the X-terminal, which is used for ground terminal operator interaction and control, failed. These terminals were unavailable for sale, new or used. As an alternative, Harris procured a laptop and Hummingbird Exceed, an X-terminal emulator, was installed.

2.2.3 Post-delivery Service at Harris Corporation

After FASSET was setup at Harris Corporation's facility in Melbourne, additional problems were found. The Hyperbus connector, used to connect three ground terminal boards together, started to produce smoke due to an electrical fault. It is likely that it was simply shaken loose during transport, but it was replaced regardless. The Fireberd bit-error-rate test set did not start properly and was fixed by reseating the internal boards and connectors. A 12 V supply failed in the payload Timing Control Subsystem (TCS) and had to be replaced. At the same time, the 5 V supply was tweaked to ensure a good supply was available to the internal boards. Some payload interconnect cables were intermittent and had to be replaced. During setup, one of the flex-waveguide connections broke and had to be replaced. At the same time, Harris installed a frequency translator for their modem in the FASSET rack.

Later, the Timing Control Subsystem again failed to respond. After extensive debugging, it was determined that there was a design flaw in the EPROM sockets of the Payload Controller

Interface within this subsystem. These sockets were originally configured for a different EPROM and with the current EPROM installed, some address lines were left floating. After this was fixed, the subsystem responded properly.

Reliability issues with emulations were a continuous problem throughout refurbishment. Sometimes the ground terminal would be unable to acquire downlink synchronization. At another point, the ground terminal failed completely and it was determined that there was an intermittent problem with the ground terminal's Frequency Reference Generator (a part of the hopping synthesizer). Harris repaired the phase-lock-loop circuitry, and the ground terminal worked properly again.

After training, the replacement 12 V power supply in the Timing Control Subsystem began to behave strangely and had to be replaced. At the same time, an external monitor port was added to provide the capability of measuring internal power supply levels without disassembling the system.

Again, the ground terminal failed to synchronize and it was determined that the problem was a failed single board computer in the ground terminal and a failure of the payload's Uplink Expander (part of the hopping synthesizer). The single board computer was replaced and the Uplink Expander was repaired. From this point, the FASSET test bed continued working well, and was available for testing with the Harris modem.

2.3 Current Status

The FASSET test bed has worked relatively reliably from Autumn 2003 to Spring 2004. In this section, its capabilities will be presented along with the associated limitations. FASSET hardware has been shown to fail and need replacement periodically, so an examination of the spares is provided. Finally, work that was not done because of time and funding limitations will be detailed.

2.3.1 Capabilities and Limitations

While FASSET is mostly compatible with [1], there are some important differences that are shown in the following table. These differences are well documented in the various technical manuals [10, 11 and 12]. More details can be found in the FASSET Engineering Change Notices [13, 14].

Feature	Capability	Limitations
Overall	Compatible with 1582B less access control appendices. LDR only.	No MDR.
Downlink Synchronization	High hop rate (HHR) fine and coarse, low hop rate (LHR) synchronization hops generated and used for acquisition and tracking. All synchronization hops can be turned on or off.	The ground terminal must see synchronization hop in first ¼ frame. Fine synchronization hop has one bit in error.

Feature	Capability	Limitations
Uplink Synchronization	Uplink Acquisition: 75 and 2400 b/s least robust protocols supported Payload supports uplink tracking but it has never been tested.	AROWs are returned one frame early. Medium and most robust 75 b/s protocols not supported. The ground terminal does not uplink track. Contention is not supported.
Data communications (C0/C1)	C0 data rates 75-2400 b/s coded, 150-4800 b/s uncoded. C1 data rates 75-300 b/s coded, 150-600 b/s uncoded Ground terminal supports 2 C0/C1 ports. Payload supports 1 TT&C C0 port (2400 b/s coded, 4800 b/s uncoded). Extensions up to 19.2 kb/s were never successfully tested.	75 b/s works uncoded and without interleaving (effectively 150 b/s) 150 b/s does not work with both coding and interleaving enabled. 300 b/s does not work with coding off and interleaving enabled. It is believed that these are ground terminal limitations. LHR data communications do not work.
Interleaving	All supported at 2400 b/s and 75 b/s. Short convolutional interleaver supported at all data rates.	150-1200 b/s only have short convolutional interleaver. See data communications above for limitations
Coding	All supported. Disabling coding doubles the data rate at the port	See data communications above for limitations
Modulation and Diversity	All supported. Extension of FSK/1 on the uplink supported.	The ground terminal does not support downlink diversities higher than FSK/2
TRANSEC manipulations	All are supported and can be selectively disabled.	The ground terminal must see a synchronization hop in first ¼ frame, which limits TRANSEC mode selections to those that have the synchronization hop in the right place. C2 cover does not work.
TRANSEC Sources	KGV-11A or KGV-11C TRANSEC simulator Intercept Box: Magic 5, Rotate 6, other simple patterns	The ground terminal must see a synchronization hop in the first ¼ frame, which means it cannot use Magic 5.

Feature	Capability	Limitations
Access Control (C2/C3)	Generate fixed or random C2 messages from ground terminal. Verify C2 CRC check at the payload. Received C2 message can be examined in the payload using the VMETRO Bus Analyzer. Generate fixed or random C3 message from the payload. Verify C3 CRC check at the ground terminal. Configuration and accesses must be preset prior to emulation.	The access control appendix was not available. No protocol support. Only one normal C2 and one robust C2 repeat cycle supported (3^{rd} entry of Table IX and 1^{st} entry of Table X in [9]). Only supported C2 diversity is 8. Only least robust C3 is supported.

Table 3 FASSET capabilities and limitations

2.3.2 Spares

As delivered to DRDC Ottawa, FASSET came with spares for the critical items. Non-complex (COTS) items were not spared as it was expected that they could be easily purchased when required. Several years have passed since the parts were originally bought and it is no longer easy to replace them. In some cases, such as the hard disks, single board computers and X-terminal, new units are no longer for sale.

Some of the "spares" have turned out to be boards that were not completely compatible and are thus useless as spares (except possibly for cannibalization). Most spares have never been tested by DRDC Ottawa and are of unknown reliability.

Neither source code nor device images are available for many of the payload programmable logic devices. In the event of a failure of these parts, it would be impossible to reconstitute them.

Some modules and boards have already been replaced with spares and thus have reduced or no spares currently. A listing of modules that were replaced with spares during refurbishment can be found in Section 5.5.

2.3.3 Work Remaining

Because of a shortage of time and money, and because of focusing on Harris Corporation's immediate requirements, several problems have not been corrected. Should the resources become available, and new requirements for the broken features arise, they should be fixed.

Within the payload's hopping synthesizer, the Downlink Expander was replaced with the only spare (see Section 3.2.10 for details). In the failed unit, one of the diodes in an RF switch matrix is shorted out. This could be fixed by replacement of the RF switch matrix (it may be impossible to find) or by replacement of the diodes. Another less appealing option is to obtain a switch matrix with a different form factor and place it outside the module.

The payload's Receiver Subsystem has two programmable attenuators. At this time, the first one has been bypassed and the second one is not properly commanded (see Section 3.1.3 for details). This improper commanding may also be affecting other areas, but this investigation has not been pursued. Because the second attenuator has a limited range (0-15 dB) and only rarely changes (and then only on power-up), this component has been left in this condition. Should the second attenuator change, then the manual potentiometers can be adjusted to compensate. This

can be fixed by resolving the problems with commanding the second attenuator, and by replacing the first attenuator. Additional details on this excessive uplink attenuation problem can be found in Section 5.3.

A description of other problems that have were identified during refurbishment, but were not resolved, is given in Section 5.

2.4 Test Equipment Used

During refurbishment and post-delivery setup and service, many pieces of test equipment were used. As listed below, some are included with FASSET and others, though not included, are essential for running and monitoring FASSET during emulations. There is special equipment required for payload loopback testing, and there is other equipment that is useful to have nearby both for normal operation and to aid in debugging when a problem occurs.

2.4.1 Included with FASSET

The following equipment is included as part of FASSET and is essential for operating FASSET:

a. Fireberd 4000 bit error rate (BER) analyzer – this is used to measure bit error rate on data accesses.

b. Miteq Mixer (TB0400LW1) – this is used for payload loopback.

c. TRANSEC Simulator (6 units) – these are used to provide a pseudo-random TRANSEC pattern that doesn't require the cryptographic control procedures needed for the KGV-11s.

d. TRANSEC Intercept Box (3 units) – these are used to provide special TRANSEC patterns (when connected to the TRANSEC Simulators) such as Magic 5 and Rotate 6.

2.4.2 Essential for FASSET

The following equipment is essential for monitoring FASSET while emulations are running:

a. HP 8565E 50 GHz Spectrum Analyzer (with option 007 for fast time domain sweeps) – this is used to observe the hopped spectrum, for monitoring LOs and dehopped signals, and as general-purpose test equipment.

b. HP 16500B Logic Analyzer (with 16550A 100 MHz State/500 MHz Timing and 16534A 2 GS/s Oscilloscope) – this is used to monitor the various clocks, to provide triggering for the Spectrum Analyzer, and as general-purpose test equipment.

c. KGV-11A (3 units, with key fill device and maintenance key) – these are used to provide a real TRANSEC source.

2.4.3 Required for Payload Loopback Testing

Along with the mixer identified in Section 2.4.1, the following equipment is necessary to do payload loopback testing:

a. 20 GHz Synthesizer – this is used to provide the translation frequency local oscillator (LO) between the uplink and downlink bands.

b. 20 GHz amplifier – this is used to amplify the synthesizer output to a high enough level to activate the mixer (it is not required if the synthesizer has a high enough output level).

c. 12 V DC (direct current) Power Supply – this is required to provide DC power to the amplifier.

2.4.4 Useful During Normal Operation

During normal operation it is handy to have the following equipment available with FASSET, or easily on-call:

a. A second HP 8565E 50 GHz Spectrum Analyzer – this is used because occasionally it is necessary to observe two spectra simultaneously especially when there are two LOs in the same chain or generated by the same module.

b. A second Fireberd 4000 or 6000A BER Analyzer – this is useful when performing the multiple C0 tests that require a second data source and sink.

c. Dumb terminal or a laptop computer with HyperTerminal for ground terminal console – this is useful during all FASSET testing to act as the ground terminal console allowing Unix errors to be observed, log files to be cleared, and shutdown to be invoked and monitored.

d. LaserJet II (or better) Printer – this is useful for printing instrument screens and files from either the ground terminal or payload.

e. Cabling, general purpose interface bus (GPIB) conversion and switching for printer – when the above printer is used, switches allow one to bypass the requirement for changing cables each time a different source is used to print.

f. RF cables, adaptors (SMA to WR-42, V to WR-22, V to SMA or K), transitions and waveguide – these are useful for implementing loopback or making test connections during debugging.

2.4.5 Useful During Debugging

Many problems were encountered during the refurbishment, enhancement and operation of FASSET. The problems were diverse and required many different types of test equipment as diagnostic aids. The ones most commonly used during debugging are:

a. VMETRO VBT-325C – this is used for monitoring the VME (VERSA module Eurocard) bus execution of the ground terminal or some subsystems of the payload.

b. HP Internet Advisor with Ethernet cradle – this is useful for monitoring the inter-subsystem communications of the payload LAN and to a lesser extent the transactions between the ground terminal and the X-terminal (or laptop with X-terminal emulator).

c. Frequency and Time Interval Analyzer – this is useful for observing the hopping patterns at RF and has some utility as a training device for people not used to frequency hopping.

d. Frequency reference with both 5 MHz and 10 MHz – this is useful to eliminate any drift between the ground terminal and payload for long emulations (because the ground terminal does not uplink time track).

e. RS-232 breakout box – useful for analyzing serial data links throughout FASSET, including data transmission from the BER Analyzer and sometimes to the printer.

f. RS-422/449 breakout box – many of the subsystem interconnects on the payload adhere to the RS-422/449 standard for connector and twisted pair wiring.

g. Cable tester – many cables have failed and it is useful to have a method of quick and thorough verification.

h. Various RS-232 cables, gender-changers – these are useful for making test connections as well as for connecting to the printer.

2.5 Sequence of Refurbishment

With the goal of leasing FASSET to Harris Corporation, DRDC Ottawa embarked on a refurbishment process that was hindered by a series of obstacles. The following subsection is a summary of significant refurbishment events in chronological order. A brief history describing FASSET's development and the motivation for refurbishment can be found in 2.1 and 2.2. Early in refurbishment, the scope of the project was limited to checking out FASSET and estimating the amount of work required to reconstitute it to its original operating condition. Later in refurbishment, the work focused more on system improvements and meeting the requirements of Harris. Throughout the work, the priority was placed mainly on the payload since it was possible for Harris to use their modem to test against a clone of itself through the FASSET payload.

2.5.1 November 2001 – August 2002

Initial checkout involved moving FASSET from a storage building to active laboratory space where it was setup for testing. A relatively large piece of lab space was made available with adequate power and cooling capabilities. The racks were setup and the subsystems were installed with all of the necessary cabling. Both of the ground terminals were also brought from storage, although only LGT2 was properly connected into the payload. The subsystem interconnections were carefully inspected and verified with diagrams and pictures, and then powered up.

Upon powering up the ground terminal, problems were encountered immediately when the three MVME147SB-1 single board computers of the ground terminal were not able to read their boot values from battery backed-up random access memory (BBRAM). Effort was invested in replacing the BBRAM chips with current, but compatible replacements, and determining the values to store in them. Unfortunately, discrepancies between hand annotations and values recorded in the manuals led to stray interrupts, anomalous start-up errors, and improper shutdowns of the ground terminal hard drive.

The payload powered up with no faults, and an emulation was started successfully. This was an encouraging result, as a failure in any of the payload's fundamental subsystems will block emulations from proceeding. Without the ground terminal, however, it was impossible to verify most of the payload, although a spectrum analyzer was used to verify that the downlink RF signal was hopping across the SHF band.

As the Harris team was making preparations for receiving FASSET at their laboratory facilities, they requested a letter of volatility. The letter contained a listing of the part numbers of all electronic memory devices in FASSET, their capacities, and a statement of their volatility. A search through documentation and a process of opening various subsystems yielded the required information. Notes on the letter of volatility can be found on page 2 of the logbook [4].

2.5.2 August 2002 – September 2002

With all of the improper shutdowns that were experienced during the reprogramming of BBRAMs, the ground terminal hard drive became corrupted. Time was spent in restoring an old image from tape backup to the ground terminal hard drive. The process was successful, but data on the formerly operational hard drive was lost. A series of spare hard disks were procured for use in backups and troubleshooting. Only a select list of hard drives are compatible with the MVME147-SB-1, and most require a read only memory (ROM) modification that was eventually duplicated in the laboratory. A procedure was also developed for efficiently making duplicate copies of ground terminal hard drives.

Both the payload and ground terminal were already fitted with removable hard drive caddies when they came out of storage. However, when new caddies were installed in the development system and LGT2, problems surfaced. Those issues, in addition to ongoing faults in the ground terminal led to the procurement of StarTech removable drive bays and caddies. More consistent performance was achieved with the caddies from StarTech.

Another surprise during refurbishment was an electrical fault in the display section of LGT2's X-terminal. This fault was compounded with the fact that prior to the failure, the performance of the X-terminal had been sluggish and unreliable. Since the remaining X-terminal display was of the same age as the one that failed, the decision was made to replace the entire X-terminal with an emulator. The Hummingbird Exceed software emulator was installed and configured on a laptop with the network interconnection provided by a Bayonet Neill-Concelman (BNC) to RJ45 media converter. The emulator was tested successfully on both Windows 2000 and XP laptops.

With serious hard drive problems, failure of the X-terminal display, and developing problems with the ground terminal's single board computers, effort was invested in setting up a standalone VME (VERSA module Eurocard) development system for troubleshooting and conducting tests. This development system was comprised of a standalone VME chassis with a single board computer, a hard disk, and no extraneous peripherals. Ultimately, the VME test system proved to be invaluable throughout refurbishment in simplifying and isolating a variety of problems. It was used to duplicate ground terminal hard drives and for testing single board computers. It was also useful in testing Attachment Unit Interface (AUI) to thinwire transceivers, the VMETRO bus analyzer card, and the reliability of Small Computer System Interface (SCSI) hard drive caddies.

2.5.3 September 2002 – December 2002

Among other failures that occurred unexpectedly, the failure of a transceiver in the TCS impeded a lot of momentum that had been built in the refurbishment process. When network problems were suspected, an investigation using a local area network (LAN) protocol analyzer determined that the fault was internal to the TCS. A detailed analysis of the Payload Controller Interface (PCI) module ensued, eventually leading to the isolation of the fault in the AUI to thinwire transceiver. Other identical transceivers failed later in refurbishment, but after studying the failure of the transceiver in the TCS, the symptoms were better understood.

A concern that was present from the beginning and throughout refurbishment was the unreliability of the Downlink Frequency Expander in the payload's TCS. For no apparent reason and with no noticeable pattern of occurrence, a fault message would be produced on the screen indicating a problem in the output of the Downlink Expander. This fault was sometimes present for hours, and other times it was sporadically present over several days. The Downlink Expander problem was not dealt with until late in the refurbishment process because it was possible to work around the problem in the early stages. Eventually, the intermittent symptoms of the fault were better understood, and it became a higher priority, so the spare expander was installed. The fault

in the expander was determined to be the switch matrix, but since the spare worked, the switch matrix was never repaired or replaced.

In the midst of studying the Downlink Expander fault, the Transmitter Subsystem failed to power-up. It was not unusual for any payload subsystem to occasionally fail a power up, but it was not normal to experience multiple consecutive power-up failures. In an effort to isolate this more critical fault, the subsystem's transceiver and Digital Interface Unit (DIU) were independently swapped with spares. The Transmitter Subsystem still did not power-up properly, indicating that the failure was not necessarily internal to the DIU. An investigation of the DIU's boot sequence revealed that it was not being properly reset during initialization. Once the reset circuit was modified to accommodate the aging power-up characteristics of the power supply, the Transmitter Subsystem was able to power-up properly.

As refurbishment continued, problems were encountered with the programmable waveguide attenuators on the uplink side of the payload in the Receiver Subsystem. The fault manifested itself with abnormally low RF power levels on the uplink. The command words from the Receiver DIU to the programmable attenuators were suspected, but the investigation was inconclusive, and other tasks became a priority. The programmable waveguide attenuator in the Receiver Subsystem was bypassed with a coaxial cable and two waveguide to coaxial transitions, and left in that state. Manual waveguide attenuators were used in the external connection to the Receiver Subsystem to control uplink levels.

2.5.4 December 2002 – February 2003

Being such a relatively large and complex system, it is not surprising that a variety of faults were encountered in the TCS. When a surge protector in a quadruple-voltage switching power supply of the TCS burned out, it was relatively simple to isolate because its effects were consistent and far spread. In a more difficult case, a stuck bit in the Downlink Delay Module was not causing a hard failure, but rather it created degradation in the bit error rate of all data communications. Although the impaired bit error rate was a known problem, the source of the problem was indirectly discovered while investigating a different error message.

Although there were no hardware faults in the LDR Baseband Subsystem, a significant effort was invested in fixing and improving its operation. A software bug in the LDR Symbol Processor was identified and fixed to alleviate a symbol readback fault that was impairing operation of the TT&C port. Another software bug in the symbol processor was fixed to correct the processing of C2 access control messages. The ability to specify a fixed or random C3 message was also introduced in the payload. Other improvements in the LDR Baseband Subsystem included the ability to have flexible uplink to downlink mapping for TT&C and C1 messages. Furthermore, the user interface and primary software control unit was modified to have flexible uplink to downlink mapping for multiple C0 assignments.

A series of test matrices was developed that provide a functionally representative set of emulations that can be used as benchmarks for FASSET operation. The matrices were divided into groups that test C0 user data, C1 secondary user data, TT&C user data, multi-user data, C2 uplink access control, and C2 downlink access control. Two main benchmark tests, STANDARD and SYSBENCH, were created to provide baseline tests of FASSET's fundamental functions.

Being one of the first steps in setting up communications, downlink synchronization is one of the earliest pieces of feedback provided to the user for gauging the synchronization progress. While investigating problems with downlink synchronization, it was discovered in the ground terminal that sometimes the local oscillator (LO) in the down-converter would shutoff because the hop strobe rate of the Frequency Synthesizer was 2.5 times faster than nominal. As the commands were accumulating too fast, the ground terminal became effectively paralysed

when the LO shutdown. A spare Direct Digital Synthesizer (DDS) Controller board appeared to alleviate the problem, so the fault on the suspected defective board was never resolved.

Problems with the ground terminal became more prevalent at one point, so a method for independently testing the payload became a priority. A built-in diagnostic mode called the external loopback proved to be an adequate method for testing a limited set of uplink capabilities. Initially, only the loopback acquisition tests worked properly, but after some fixes, the C2 access control and TT&C loopback tests also worked.

2.5.5 February 2003 – September 2003

After many fixes and improvements, and after a period of reliable operations, FASSET passed all of the tests in the emulation test matrices. Having passed all of the acceptance criteria, the decision was made to deliver FASSET to Harris Corporation. Although some of the shipping had been previously planned, most of the logistic work could not proceed until close to delivery. The shipping of FASSET from DRDC Ottawa to Harris in Melbourne was successful, but during the process a number of lessons were learned. Some of the lessons were learned from negative experiences, while others were learned from examples of good preparation and execution. It was found that some of the packaging was not adequate, and as a result some physical damage was incurred due to mechanical stress during shipping. Furthermore, a delay in the shipping of the classified documents made some of the troubleshooting more difficult at the Harris facility.

With the benefit of planning during the dismantling of FASSET, it was relatively simple to reconstruct the system after delivery. There were some logistical problems to overcome, including AC power, air conditioning, test equipment, and various information technology (IT) and security issues but all of the obstacles were dealt with as they arose. During shipping, a power supply in the TCS was damaged, and a ROM in the Fireberd test set was jarred loose. Also, a design flaw in the Payload Controller Interface (PCI) of the TCS was in some way exacerbated by the shipping process. It became a hard fault, but having previously studied the PCI for other reasons, the FASSET team was able to isolate and resolve the problem efficiently. During emulations carried out at Harris, a fault developed in the payload Uplink Expander that impaired uplink synchronization. The cause of the fault was not definitively determined, however, some electrical connections in the module required some touch-ups. The only apparent shipping problem induced in the ground terminal was a loose circuit board (Hyperbus), although there may have also been some residual contribution to a fault that later developed in the Frequency Reference Generator.

Finally, after months of working-in the system with training and testing, FASSET was reliably running emulations. Harris was able to test their TRANSEC devices, and run multiple daylong emulations with both the payload and ground terminal.

3. FASSET Items Repaired

This section contains a compilation of hardware and software faults that were repaired during the refurbishment of FASSET for Harris Corporation. The topics are categorized into three subsections: system, payload, and ground terminal. The subsections may be read independently, and as such some background information has been repeated to aid the reader.

Although the troubleshooting for some failures started at the system level, most of them were eventually isolated to individual subsystems within either the payload or ground terminal. Most of the overall time and effort was spent in repairing faults in the payload, and that is reflected in the number of topics in that subsection. Being one of the more complex and critical subsystems, the Timing Control Subsystem (TCS) turned out to contain many of the major hardware faults. Serious errors in software were located and fixed, primarily in the code downloaded to the LDR Baseband Subsystem.

3.1 System

System level faults are those that include equipment and items that either span both the payload and ground terminal, or are part of neither. Repairs to system faults that are discussed in this subsection include the user data patch panel and the bit error rate test set. The issue of excessive uplink attenuation is introduced in this section, with the payload portion detailed in Section 3.2.3 and the ground terminal portion detailed in Section 3.3.5.

3.1.1 User Data Patch Panel Connectors

Loose connectors on the patch panel for user data I/O were found to be causing many intermittent data errors. During the first few months of refurbishment, a persistent residual error rate (with no coding and no interleaving) of 1.3×10^{-6} was present during all testing. It was not until an overlaying error source was removed that it was possible to observe this residual error rate by itself. Eventually, this residual error rate was identified as the result of a defective clock line in one of the RS-232 interfaces to the patch panel, and corrected.

To increase reliability of the patch panel interfaces, all connectors were screwed firmly onto the back of the patch panel. All cabling for the unused LGT1 ground terminal (LGT1 was not working and therefore not used) was disconnected and removed to make the area clearer for troubleshooting. Also, a modification was made to the patch panel to allow an external BER test device to be connected instead of the FIREBERD 4000.

The associated logbook entries for this work can be found on pages 32-36 of [5].

3.1.2 Bit Error Rate Test Set – Failed Self Test

The mechanical stress experienced during shipping of FASSET to Harris Corporation had negative effects on the hardware, including the Fireberd communications analyzer.

For shipping, the Fireberd was packed carefully in bubble wrap and foam, and placed in a solid wooden shipping crate. An air-ride truck was used for transport in order to minimize the jarring due to bumps on the road. Upon first powering-up the Fireberd at Harris, the instrument failed its ROM check, reported the fault on the display, and did not start-up properly. To ensure that no ROMs were loose, the Fireberd was opened and the ROMs were re-seated in their sockets. The ROM validation at power-up continued to fail consistently. Since no spare bit error rate

analyzer was available at the time, the main circuit card and the ROMs were re-seated. This final re-seating was enough to make the required connections solid, and the Fireberd passed its boot-up ROM check.

The associated logbook entries for this problem can be found on pages 1-3 of [6].

3.1.3 Excessive Uplink Attenuation

With both payload and ground terminal systems fully powered up, FASSET emulation tests were conducted. Numerous problems occurred during emulation testing, and can be broadly categorized as either:

a) Downlink synchronization failure (see Sections 3.2.10 and 5.2),
b) Uplink coarse synchronization failure (see below and also Sections 3.2.3 and 3.3.5),
c) Uplink fine synchronization & tracking failure, or
d) User data bit error rate problems (see Sections 3.2.8, 3.2.9, and 5.4).

The following discussion concerns significant failures in the uplink coarse synchronization category that were encountered. Problems in this area can frequently be traced to low or null RF levels in the 44 GHz uplink from the ground terminal Transmitter to the payload Receiver Subsystem. Payload and ground terminal specific problems related to this subject can be found in Sections 3.2.3 and 3.3.5, respectively.

The excessive uplink attenuation problem appeared after the ground terminal reported "Downlink sync acquired", but would not progress to uplink coarse synchronization. Probing with a spectrum analyzer on the RF uplink waveguide showed that the ground terminal transmitter was enabled, and was properly generating an uplink hopping pattern as determined by the settings of the ground terminal and payload TRANSEC devices. The Transceiver Emulator (TRANSEM) is a hardware component of the ground terminal, which includes the final upconversion stage and the first downconversion stage of the RF front end. The RF level of the hopping pattern was determined to be approximately 20 dB too low at the output of the ground terminal TRANSEM, and was not measurable at the input monitor port of the payload receiver.

Investigation led to the discovery of significant problems within the RF uplink circuit. First, the ground terminal's TRANSEM programmable output attenuator appeared to be intermittently producing excessive attenuation. A bypass with an external attenuator was implemented as a solution to this problem (see Section 3.2.3). The second problem was found in the programmable attenuator of the payload's Receiver Subsystem (see Section 3.3.5). A simple bypass was used to eliminate the failed component from the system. Another minor source of attenuation in the payload receiver was also located in the RF downconverter, and accounted for with an external manual attenuator.

With the problems resolved in the uplink RF circuit, subsequent emulation testing was able to progress fully through downlink and uplink synchronization into the "Uplink Tracking" state, which allows user data to be passed. More details on this series of problems can be found on pages 8-17 and 31-32 of [5].

3.2 Payload

The items repaired that are described in this subsection involve faults associated with the payload portion of FASSET. The payload repairs described below include the network transceivers, the Transmitter reset circuit, and the Receiver Subsystem. Also discussed in this subsection are the Payload Controller Interface and power supply faults in the Timing Control

Subsystem, and several items that were repaired in the LDR Baseband Subsystem. Finally, repairs to the Downlink Delay Module and the Uplink and Downlink Expanders are detailed.

3.2.1 Network Transceiver Faults

The payload uses a 10base2 thinwire coaxial network to connect the sequencing computer to each of the subsystems, as in Figure 2. Every subsystem that is connected to the network has a Digital Interface Unit (DIU) and an AUI-to-thinwire transceiver. The DIU is responsible for ethernet communications, subsystem monitoring and reporting, and internal controls. The Timing Control Subsystem (TCS) has a specialized DIU, called the Payload Controller Interface (PCI), which has some unique functions.

When a transceiver or DIU fails, the subsystem is not able to communicate over the local area network (LAN), so the sequencing computer automatically shuts down the subsystem to protect it. Common internal failures are detected by the DIU and reported to the Configuration Computer through the LAN, however some faults can result in no response to the power-up queries.

During refurbishment, the first transceiver failure began with a persistent power up error in the TCS that was reported by the sequencing computer. The effects of the fault were not obvious, however a systematic tracing through the subsystem eventually isolated the problem. A LAN protocol analyzer was used to verify that query packets were leaving the sequencing computer while response packets from the TCS were not being produced. A network connection was made directly from the sequencing computer to the TCS to ensure that the fault was not due to a broken link in the ethernet cable. Once the physical cable connection was proven, the transceiver was temporarily swapped with a working one, but it made no difference. Further investigation located an incorrect termination on the LAN cable that had masked the results of swapping the transceiver.

After detailed analysis of the PCI source code, circuit diagrams, and logic analyzer plots, it was determined that the main processor was executing properly. It was running through its algorithm until it found problems communicating on the LAN, then it turned on the subsystem status light emitting diode (LED) to indicate failure. The lighting of this LED was observed when the safety shutdown was disabled. Eventually, the TCS transceiver was retested and identified as the fault. Other transceiver faults occurred in the Sequencing Computer and LDR Processor Subsystem, but after the transceiver fault in the TCS, the symptoms were recognized.

An ongoing issue with the payload LAN that may be contributing to the transceiver failures is the fact that the LAN's bulkhead connectors are not insulated from the subsystem housing. Currently, the outer conductor of each coaxial bulkhead connector is physically and electrically attached to the subsystem housing as in Figure 3. Since the subsystem housing is grounded, any ground currents from other sources can interfere with LAN communications.

More details on the transceiver fault in the TCS can be on pages 60-74 of [4].

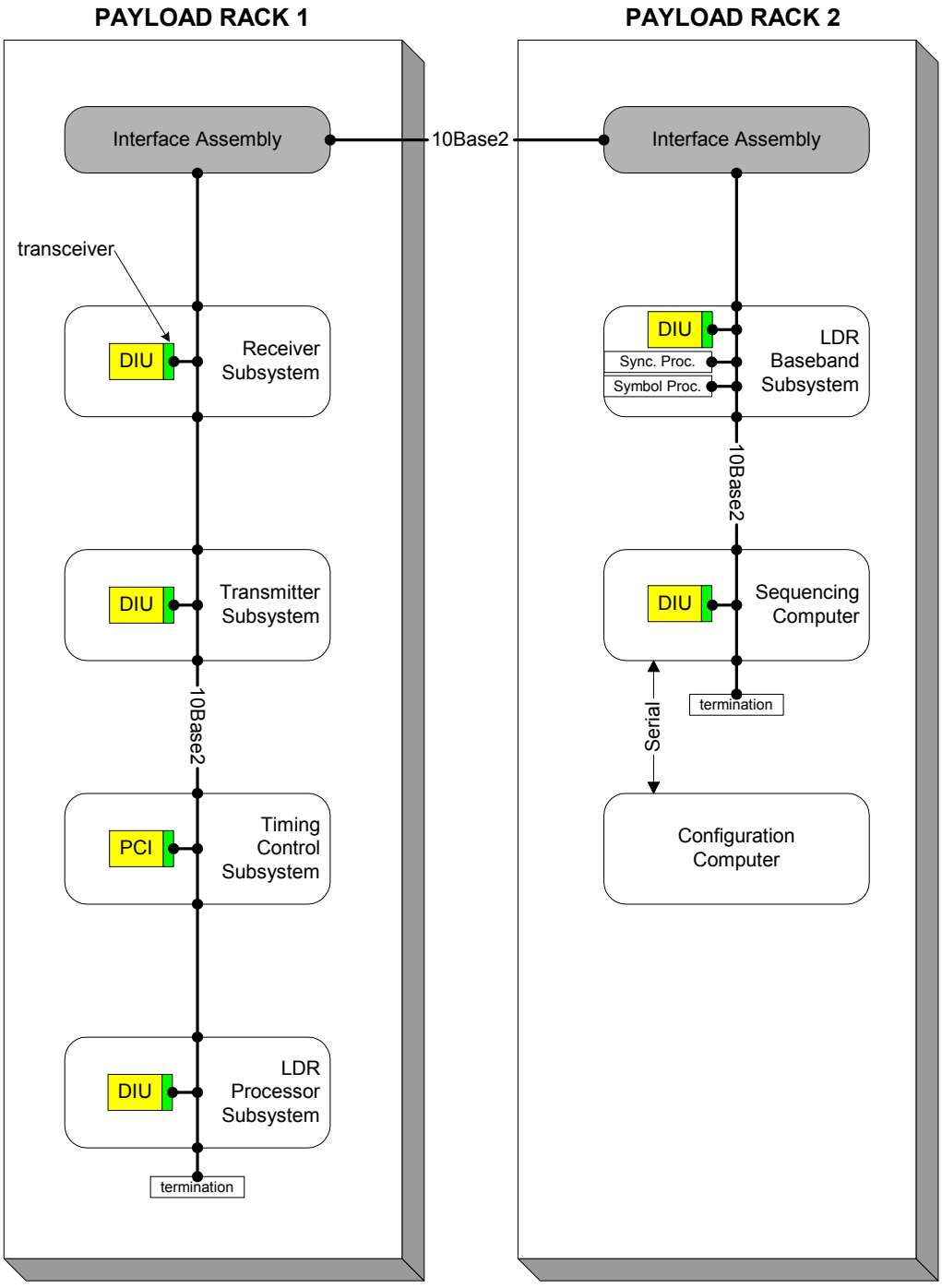


Figure 2 Payload network connections

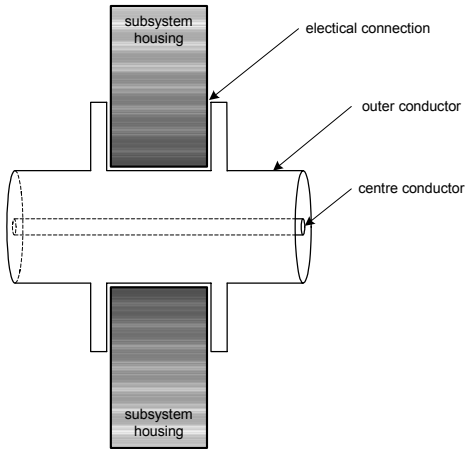


Figure 3 Payload network bulkhead connectors

3.2.2 Transmitter Subsystem Not Responding

Each subsystem in the payload is powered-up independently by the Sequencing Computer. After subsystems have stabilized, the Sequencing Computer polls them. If any critical errors are detected or if there is no response after repeated attempts, the Sequencing Computer then shuts down the problem subsystem.

During refurbishment, there were several occasions when the payload Transmitter Subsystem would not respond after power-up to the sequencing computer. Usually, after cycling the power on the payload, the subsystem would then respond. On one occasion it did not respond despite many attempts at power-up.

To isolate the problem, the payload subsystems were powered-up with the safety shutdown disabled (which meant the Sequencing Computer would not shut down unresponsive subsystems). When power was applied to the Transmitter Subsystem, there was no response and none of the status lights in the front panel were lit. First the AUI-to-thinwire transceiver was replaced, resulting in no improvement. Then the digital interface unit (DIU) was swapped with a spare, again with no improvement. The cables and connecters involved were inspected and some minor repairs were made to the cabling. This did not correct the problem. The LAN protocol analyzer was then used to verify that there were three packets sent to the Transmitter Subsystem with no response. The Transmitter Subsystem was opened up and the DIU cover was removed. It was then verified that the address lines of the DIU were properly set (00010). Various input clocks were inspected to ensure that they were being properly received in the DIU. During this phase, it was determined that the Transmitter Subsystem DIU requires the hop clock to initialize properly. All necessary clocks were found to be present.

A logic analyzer was used to probe the RAMs of the processor in the DIU. This allowed inspection of the data and addresses used during execution. It was determined that the DIU processor was in a non-code area of the address space. This could have been caused by a bug in a properly running program, failed hardware or by an improper reset. Since the DIU was previously functioning properly, the reset hypothesis was tried first. By grounding out the power-up capacitor, the board was artificially reset. This caused the DIU to work normally. It was felt that the aging power supplies were coming up slower than originally designed and the time constant of the reset was no longer long enough.

By quadrupling the capacitance, the time constant of the circuit was lengthened. This solved the power-up problem. The final diagram of the reset circuit is shown in Figure 4.

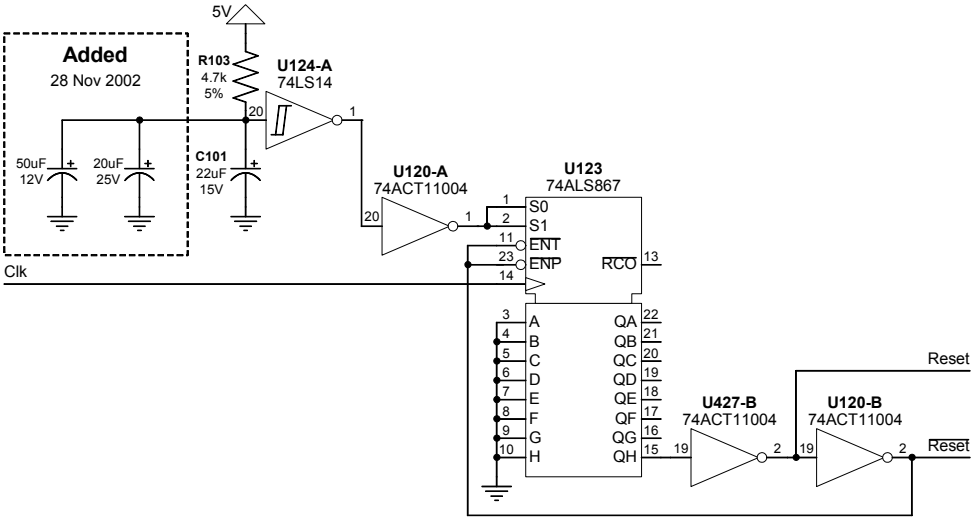


Figure 4 Payload Transmitter Subsystem DIU reset circuit

Since power-up problems have been seen occasionally in other subsystems, this problem may have occurred in other DIUs. If other subsystems fail in the same manner, then their reset circuit time constant should be lengthened accordingly.

It is possible that this problem is related to the PCI EPROM socket that was not properly matched to the EPROM device used (The PCI is the specialized DIU used by the Timing Control Subsystem). See Section 3.2.5 for further discussion on this topic. The associated logbook entries for the Transmitter reset problem can be found on pages 121-128 of [4].

3.2.3 Receiver Attenuator Malfunction

As described in Section 3.1.3, a number of problems were encountered while running full emulation testing on the FASSET system. This subsection concerns significant failures that were experienced during the uplink coarse synchronization process. Symptoms related to this type of problem often appear at the system level, and can be traced to low power levels in the RF uplink signal from the ground terminal to the payload. Information on the preliminary analysis of these uplink synchronization problems is provided in Section 3.1.3.

A significant problem in the RF uplink chain was identified during an investigation and eventually isolated in the payload's Receiver Subsystem. Although it was first suspected, the low noise amplifier (LNA) of the receiver was tested with external continuous wave (CW) signals and found to be functioning within specification with a measured gain of 13 to 15 dB. The design of the subsystem utilized a programmable input attenuator to simulate uplink fading conditions during emulation. After independently testing the attenuator at a laboratory bench, it was found to be defective. Rather than repairing or replacing the failed unit, it was bypassed permanently.

In the RF downconverter of the payload receiver, a secondary source of attenuation was also identified, and was found to be operating at an uncontrolled level in the range 0 to 15 dB. This level appeared to take a random value at power on, and was apparently not receiving any update commands from the Receiver's DIU. There was no way to bypass this attenuator, so it was decided to note the problem for possible future work, and to compensate for the 0-15 dB variation by backing off on the manual uplink attenuator on the ground terminal's TRANSEM output.

With the TRANSEM and payload receiver problems resolved, subsequent emulation testing was able to progress fully through downlink and uplink synchronization into the "Uplink

Tracking" state, which allows user data to be passed. More details on this series of problems can be found on pages 8-17 and 31-32 of [5].

3.2.4 Timing Control Subsystem Power Supply Failures

The Timing Control Subsystem (TCS) has one switching and six linear DC power supply units (PSU) that provide power internally to the subsystem. Table 4 summarizes the models and specifications of the PSUs. The Payload Controller Interface (PCI) module is a circuit card assembly within the TCS that is responsible for checking power supply levels, and for communications and control signals between the TCS and other subsystems. After boot-up, the PCI starts reporting the telemetry status of the TCS back to the Configuration Computer.

In December 2002, PSU#1 failed and it was replaced with 2 independent supplies (PSU#1A and PSU#1B). The PSUs in the TCS are densely packed into an upper and lower layer at the front of the subsystem. The top layer must be removed to access the bottom layer. PSU#1A, PSU#1B, and PSU#4 are mounted on the top layer, while the others are on the lower layer.

Power Supply Number / Model		Rail Voltage (V)	Remarks (location)
1A	+5, ±15 V Switching PSU (GLT-03-200, Sola/Hevi-duty)	+5 +15 -15	PSU #1 (LFQ-26-1) was replaced by PSU #1A and #1B (upper layer)
1B	-5.2 V Linear PSU (model not known)	-5.2	(upper layer)
2	+28 V Linear PSU (LDS-Y-28)	+28	(lower layer)
3	+5 V Linear PSU (LDS-Y-OV)	+5	(lower layer)
4	-5 V Linear PSU (LDS-X-5-OV)	-5.2	(upper layer)
5	+8 V Linear PSU (LDS-Y-02)	+8	(lower layer)
6	-12 V Linear PSU (model not known)	-12	This supply replaced the failed unit (LDS-Y-12, 3.7A), but still meets requirement (lower layer)

Table 4 Power supplies in the TCS

After FASSET was disassembled at DRDC Ottawa, shipped to Harris Corporation, and reassembled in their laboratory, one of the initial failures encountered was the –12 V PSU.

After thoroughly checking cable connections and power sources, the first power-up yielded errors early in the start-up sequence. The sequencing computer was not able to communicate with the TCS following two attempts after power-up, so the subsystem was powered off by the safety shutdown. The fact that no useful messages were reported on the

payload Error Messages Screen indicated that the PCI was not healthy enough to respond with meaningful packets.

A direct ethernet connection was made between the sequencing computer and the TCS to ensure that packets were not being lost due to a problem in the network cable. A LAN analyzer was not available to verify if either the query packets were not leaving the sequencing computer, or if response packets were not leaving the TCS. The TCS was removed from the rack and opened to allow swapping of the AUI-to-thinwire transceiver unit that is used by the PCI to respond over the network. The new transceiver had no effect, indicating that the problem was further up the chain, closer to the PCI. Since the TCS was unable to respond on the LAN with status messages, a more fundamental fault was suspected such as power, the LAN coprocessor, the central processing unit (CPU), or the dual port random access memory (RAM).

The PCI module was opened and setup on two individual VME-like extenders so that the board could be probed. A 25-pin ribbon cable was employed to extend the sense lines for the PSUs and the light emitting diode (LED) drive lines. The ±15 V and the +5 V levels were present across the PCI board, although the +5 V was sagging to 4.78 V as measured at the top of the board near the connectors. A documentation search and a verification of the PSU sense lines on the 25-pin extension cable determined that the –12 V PSU had failed. The –12 V PSU was removed from the TCS, tested at a laboratory bench, and determined to be in a failed state.

The –12 V PSU provides power to the Frequency Synthesizer module of the TCS that distributes clocks and LOs internally and externally to the subsystem. The failure of the –12 V PSU, however, did not cause the TCS to not respond on the LAN. That symptom of failure was caused by an intermittent problem with EPROM sockets in the PCI module. This second failure within the TCS, compounded with the –12 V PSU failure, made isolating the fault difficult, and in fact, no symptoms of the –12 V PSU failures were ever observed because of the problem in the PCI.

A new –12 V PSU was not readily available in the required form factor, but Harris was able to immediately provide a partially used PSU. The replacement PSU was tested at a laboratory bench under loaded and unloaded conditions. Having passed all testing successfully, the PSU was reinstalled in the TCS and loaded back into the rack. The TCS was then able to power-up properly and full emulations were conducted on the FASSET system.

The cause of the fault in this PSU is unknown, although it may have been a result of mechanical wear due to shipping, age, storage, or insufficient thermal dissipation within the TCS. The replacement –12 V PSU was later suspected to have failed (see Section 3.2.11), but the problem turned out to be a faulty load that was dragging down the voltage level of the supply. The partially used power supply was replaced with a new unit, regardless.

More details on the problems associated with PSU#1 can be found on pages 136-145 of [4], while information on the –12 V PSU can be found on pages 1-3 of [6].

3.2.5 Timing Control Subsystem Payload Controller Interface Failure

A number of payload power-up sequencing errors were encountered which involved failures in the Timing Control Subsystem (TCS). Some of these problems were identified by TCS status lights as power supply failures, and were resolved by power supply replacements. In one case, an LED status error persisted after replacing a failed power supply, and could not be quickly resolved. It was necessary to remove the Payload Controller Interface (PCI) module from the TCS box and extend it from the backplane using VME extender boards (individual boards, 2 singles were needed because the spacing was not VME). In the extended configuration it was possible to connect HP16500A logic analyzer probes to the address and data lines of the I960 CPU, and to trace the program execution. After a significant effort was invested in tracing the PCI program through various start-up routines, it was determined that the LED status error was

the result of a LAN communications failure. The problem was traced to an improper termination on the LAN analyzer, and was corrected.

In the course of connecting the logic analyzer probes to the PCI processor bus lines, a discrepancy was noted between the pin-out shown on the PCI diagram for the AM27C010 EPROM address lines and the pin-out shown on the data sheet for the device. One of the address lines (A14) on pin 29 of each of the four EPROMS was apparently not connected to anything. Since the EPROM code was executing correctly at that time, this discrepancy was noted but not pursued to determine which diagram was correct.

Several months later, in the course of checking out the payload operation after delivery to Harris Corporation in Florida, yet another power supply failed in the TCS box. When the power supply was replaced, it was found that the TCS box would no longer power up correctly, but would fail with all status lights on. Investigation showed that all power supplies were working correctly in the TCS, and that the payload LAN appeared to be communicating properly with all payload boxes except the TCS. Once again the TCS was removed from the FASSET payload rack and reconfigured to power up in test mode with the box open. The PCI module was again extended, and logic analyzer probes were attached to monitor program execution on the PCI module.

Subsequent tracing of program execution revealed that the I960 CPU was receiving incorrect op-code bytes during fetch cycles from the four AM27C010 EPROMs containing the firmware code. After verifying that all of the EPROMs still contained the correct program codes, and that the I960 CPU was functioning correctly, further investigation was focussed on the A14 address line discrepancy noted earlier in this section.

Before discussing this problem further, it must be noted that the PCI memory map requires that the lowest two address lines of the CPU (A0 and A1) be used to determine which of the four EPROMs are selected for an instruction fetch, or read cycle. The remaining CPU address lines A2 – A18 are connected to A0 – A16 respectively on each of the four EPROMs. This reflects the “CPU byte address divided by 4” mapping required for the EPROM address lines.

It was found that the assignment of pin 29 to EPROM address line A14 was indeed correct as per the data sheet for the device, however there were corresponding errors in the PCI schematic for each EPROM socket. There was no connection made from CPU address line 16 to pin 29 of any of the EPROMS, corresponding to EPROM address line A14. It was also found that two other address lines were incorrectly connected to the EPROMs. CPU address line A16 is shown incorrectly connected to pin 3 on each EPROM, which is actually EPROM A15, instead of the intended EPROM A14 on pin 29. CPU address line A17 is incorrectly connected to pin 31 on each EPROM, which is actually the /PGM pin, instead of the intended EPROM A15 on pin 3. Figure 5 is an extract from [15], drawing #112709, sheet 4 of 8. It shows the problem connections highlighted on a diagram representing the four EPROMs with a single sketch. Note that EPROM U401 is associated with data lines D0-D7 plus chip select line BE0. Similarly, U402 is associated with D8-D15 plus BE1, U403 with D16-D23 plus BE2, and U404 with D24-D31 plus BE3.

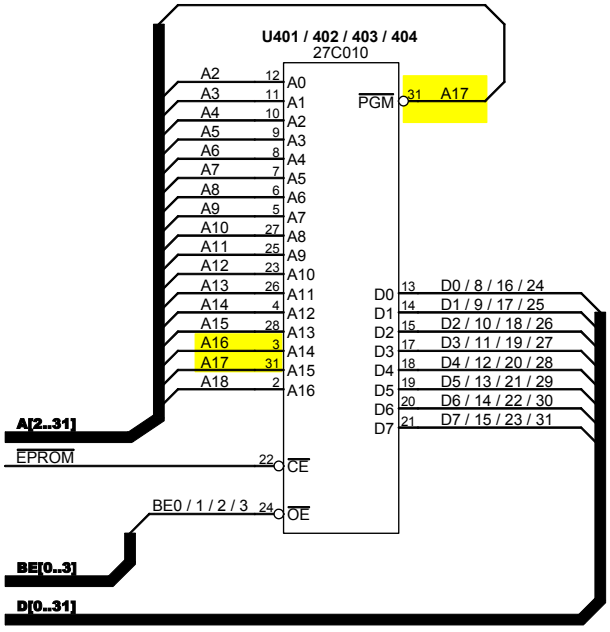


Figure 5 Extract from original PCI EPROM wiring diagram

To temporarily bypass the problem, the four floating EPROM A14 pins were connected to ground to force a logic “0” on this address line. This was an acceptable repair since the upper portion of the EPROM address space represented by A14 = ”1” was unused, and could be permanently disabled.

Analysis showed that the error in connecting CPU address line 17 to the /PGM pin of each would have no effect on the functioning of the EPROMs as long as no programming voltages were supplied to the chips.

Further analysis showed that the error in connecting CPU address line 16 to the A15 pin of each EPROM did not cause a malfunction because the upper portion of the EPROM address space represented by A15 = “1” was unused. It could never be addressed because the CPU A16 line was always a “0” when addressing the EPROMs. Figure 6 is a similar diagram to Figure 5, and shows the wiring errors that were found, with corrected labels applied to pins 3, 29 and 31. The modification to ground the floating A14 line on pin 29 of each EPROM is also shown.

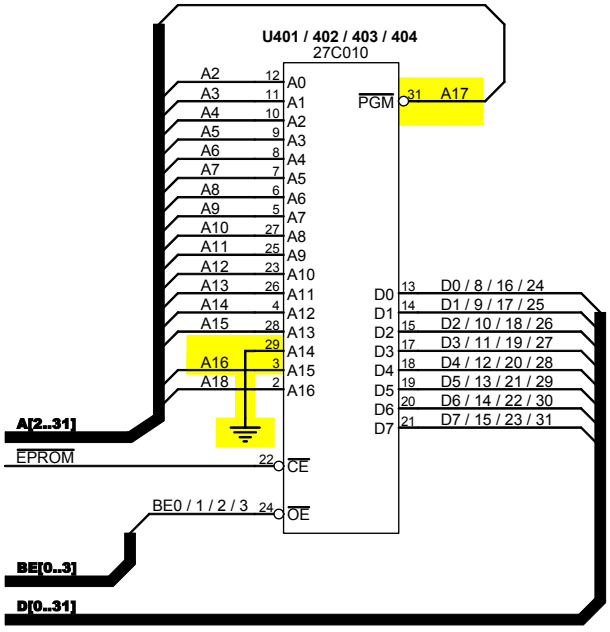


Figure 6 Modified EPROM diagram showing corrected labels and A14 grounded

The TCS was verified to be operating normally after this repair, and was reinstalled in the FASSET payload rack to continue checkout testing.

It should be noted that the PCI EPROM circuit repairs discussed in this section are only temporary, and do not actually correct the address line wiring errors. This could represent a serious problem if the PCI firmware had to be modified, and expanded to a point where it occupied a larger portion of the EPROM address space. The proper and permanent modifications required for the PCI circuit board are shown highlighted in Figure 7, and are included in Section 5.6 as recommendations for future work.

More information on this problem can be found on page 68 of [4], and on pages 2-3 of [6].

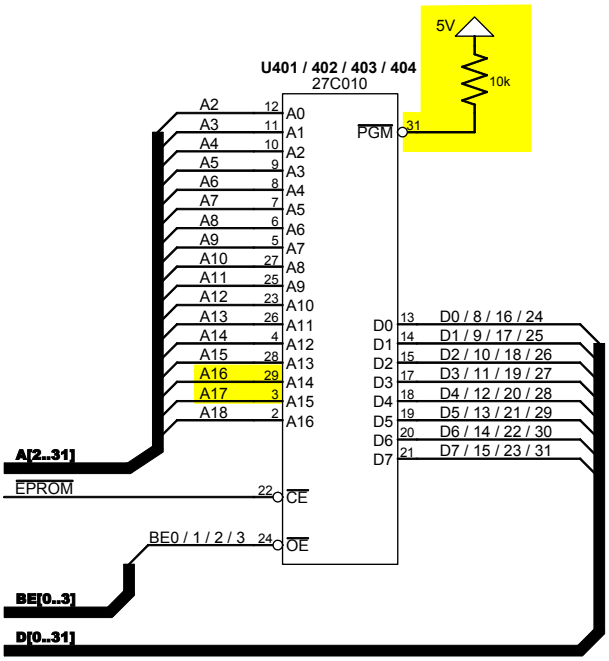


Figure 7 EPROM diagram with recommended final modifications

3.2.6 LDR Baseband Subsystem – TT&C Not Working

Prior to interoperability trials at Lincoln Lab in 1997, the decision was made to concentrate on a single C0 data channel to the exclusion of all other forms of communications because C0 capability was essential for interoperability trials. When modifications were made to FASSET to support the trials, the other types of communications (multiple C0's, TT&C, and C1) were considered to be less important and no tests of these types were run. The modifications were made only to support flexible mapping for a single C0 channel and the software code for these other types was not accordingly updated. Over the years, this fact was forgotten and only rediscovered during refurbishment.

When payload loopback tests were first run, all attempts to run TT&C tests failed because of a symbol readback fault from the Sequencing Computer. This error results when the Sequencing Computer sends configuration information to the LDR Symbol Processor, then requests a readback of this information and the two values do not match. Various combinations of TT&C switch settings (the switch replaced the old TT&C plug) were tried with no improvement. It was found that even the non-loopback DIAGSB1 (formerly SYSBNCH1) configuration could not be run because of this symbol readback fault.

The LDR Symbol Processor board was replaced with the spare (the ROMs were swapped too). The TT&C loopback was still not working, but occasionally DIAGSB1 (which includes a TT&C assignment) worked on the uplink only. An examination of the code revealed the consequences of the changes made prior to interoperability – the TT&C access was not being properly serviced within the software.

The download portion of the LDR Symbol Processor was changed. Specifically LRDATAFO.C, which formats the downlink data, was modified to treat TT&C as it used to prior to interoperability modifications. The old fixed mapping TT&C code was uncommented and added to the C0 section on the condition that the access type was TT&C.

There were some problems regenerating the download software because the date on the development machine had been reset to 1980 and the nmake utility believed that all files were current. The method for preparing the download software in [11] was cumbersome, error prone and required an external EPROM programmer (even though no EPROMs are programmed for the download). The ABStoBIN utility (see Annex D.1) was developed to automate the preparation. The entire process for regenerating the LDR Symbol Processor download software can be found in Section 6.3.2. After regeneration, the downlink TT&C capability was restored (but still using the fixed mapping). This was later enhanced as documented in Section 4.2.5.

Testing showed that there was still a problem. The symbol readback fault was still occurring most often when TT&C was turned on after another emulation was run. Turning off coding, interleaving and permutes also enable TT&C emulations to run. At this point it was suspected that there was an initialization problem.

Trace information was added to the kernel to observe the differences between good and failed configurations. The trace lines were inserted into the file (CONFIGSY.C) that processes the configuration and readback messages. While attempting to regenerate the kernel, an error occurred in the batch file SYM_ROM.BAT. Several of the lines were supposed to copy build files to a higher directory but used the destination directory "..\" which DOS (disk operating system) did not accept. This was changed to a destination of ".." and the build completed error free. The process for regenerating the LDR Symbol Processor kernel EPROM can be found in Section 6.3.1.

Subsequent tests showed that the values returned by the kernel did not match those sent in the configuration message. An examination of the code revealed that the kernel was not saving the downlink portion of the TT&C configuration messages. Also, it was not sending the downlink portion back when responding to a readback message. This was corrected in CONFIGSY.C and the kernel was regenerated. This solved the symbol readback fault problem and TT&C worked properly (with fixed mapping).

The associated logbook entries for this TT&C problem can be found on pages 20-27 of [5]. The logbook entries for the symbol readback fault problem can be found on page 45 of [5].

3.2.7 LDR Baseband Subsystem – C2 Messages Not Working

C2 messages are the uplink access control communications sent from the ground terminal to the payload. Because the message appendices of [1] were not available to the FASSET contractors, only the CRC portion of the C2 was designed into the payload (because the CRC details were included in the main body of [1]). The ground terminal normally generates random bits for the uplink access control message and appends an appropriate CRC.

During refurbishment, C2 tests were run and all failed. The only indication of failure available to the user is a counter on the payload screen that counts C2 messages received that contain errors. Knowing that the LDR Symbol Processor is responsible for processing C2 messages, the first step was to add trace information to PROULAC.C. The process for regenerating the LDR Symbol Processor download software can be found in Section 6.3.2. The VMETRO Bus Analyzer was installed in the payload Baseband Subsystem immediately adjacent to the LDR Symbol Processor. Examination of the trace information showed that the decoded C2 message was all zeros.

While using the bus analyzer, there were several failures of the VMETRO. It was believed that the on-board battery was the problem. Because the battery was spot welded, it was difficult to replace the new one with a solder connection. This was done and the VMETRO reinserted. It still was unreliable and examination of the board showed scraping damage to both the solder side and the component side.

Another VMETRO was borrowed for the duration of these fixes. More detailed trace information was added in ACDECODE.C to observe the exact inputs and outputs of the hardware decoder (which is hosted on the custom board Data Routing Switch). One problem was identified immediately. In the documentation for the decoder interface, it states that the decoded data is available after the 21st read (hard coded) from the decoder output, but the software was taking the data immediately. Changes were then made to ACDECODE.C to ignore the first 21 reads and take the data starting on the 22nd read.

This change then allowed the proper decode message to be read. The ground terminal was set up to transmit a fixed C2 message and the traces were examined. It was noticed immediately that the expected C2 message (the one sent from the ground terminal) was reversed when it came out of the decoder. A close examination of the ground terminal documentation showed the ground terminal sends out most significant bit first and in fact it is the ground terminal user interface screen that is reversed. This must be taken into account when generating fixed C2 messages from the ground terminal.

Once the desired C2 bits had been properly set on the ground terminal (by reversing the original bits) the desired message came out of the decoder. The ground terminal was then switched to random C2 message generations (with valid CRC). The CRC check still failed. Additional trace information was put in PROULAC.C to show the intermediate steps prior to the CRC check. The internal data was somehow partially corrupted.

Extensive examination of the code showed that a bit copy routine in BITSTUFF.C was flawed in that the accumulator was not zeroed prior to a byte-aligned transfer (non-byte-aligned transfers were done properly). Once fixed, the payload C2 message CRC check passed.

The associated logbook entries for the problem with C2 messages can be found on pages 57-60 of [5].

3.2.8 Downlink Delay Module – Excessive Bit Error Rate

While running emulations using the DIAGSB1 (formerly SYSBNCH1) configuration, it was noted that uplink data from LGT2 to the payload TT&C port had a near-zero bit error rate yet the downlink data from the payload TT&C port to LGT2 had an uncoded error rate greater than 10^{-2}. When the STANDARD configuration was used, the end-to-end error rate with coding on was 10^{-3}.

Coding was then turned off and different BERT patterns were selected: mark, space, 1:1, and 7:1. The test port PLD-XMTR-IP-MON, which shows the data prior to modulation, was examined. It was determined that one out of eight bits was stuck at ‘0’. Such a problem would exhibit errors 1/16 of the time for random data (1/8 of the bits are affected, 1’s go to 0, 0’s stay at 0), which corresponds to an error rate of 6×10^{-2}.

The VMETRO board was placed in the LDR Baseband Subsystem to monitor the data at the LDR Symbol Processor. (See A.1.1.1 for trace details.) The data at the LDR Symbol Processor appeared to be correct with no stuck bits. Thus, the problem was located somewhere between the LDR Symbol Processor and the transmitter test port. By examining each point in the chain, the problem was isolated in the Downlink Delay Module. A logic analyzer was used to verify that the data on the input to the Downlink Delay Module was valid, while the output contained stuck bits.

The Downlink Delay Module is used to provide the downlink delay that would normally be associated with propagation time travelling to and from satellites (distance from ground terminal to satellite divided by the speed of light). The input is a formatted serial bit stream that is converted into bytes on-board. A ninth parity bit is generated for each byte. These are then written to memory and read back for output at the appropriate time. If the error occurred in the memory, then a parity error should have been generated. This was not the case, so it is likely that

the error is occurring at or immediately following the serial to parallel conversion. Two programmable logic devices may be involved. The most likely candidate is WTACC.PLD that does the conversion from serial to parallel. The other candidate is WTCTL.PLD that generates related timing and control. Because four pages of Downlink Delay Module diagrams could not be found, fixing the module hardware became too time consuming to pursue.

The module was replaced with the spare that worked properly. The residual error rate (with no coding and no interleaving) was measured at 1.3×10^{-6}. This residual error rate was later identified as the result of a defective clock pin, and corrected (see Section 3.2.9).

The original module is now the spare and needs to be fixed. The associated logbook entries for this problem can be found on pages 28-31 of [5].

3.2.9 TT&C Loopback Modification and Connector Repairs

As delivered, the FASSET system included the capability of extracting and injecting C0 data at the payload by means of a Tracking Telemetry and Command (TT&C) port provided on the front of the LDR Baseband Subsystem. One of the unresolved problems noted in the FASSET documentation ([10] and [11]) was that a special TT&C Loopback Plug had to be attached to the TT&C data port as part of the initial emulation start procedure. Emulations failed if this procedure was not followed, for reasons briefly explained in [11]. In order to facilitate this process, the TT&C Loopback Plug was replaced with a TT&C Loopback Switch installed on the front panel of the LDR Baseband Subsystem beside the TT&C data port. The switch enabled the user to easily select the loopback position as part of the emulation start procedure, and then return to normal data mode during the test period. This eliminated the problem of constantly manipulating and keeping track of the small loopback plug originally provided, and improved the reliability of the TT&C loopback mode.

As discussed in Section 3.2.8, a significant improvement to the downlink bit error rate (BER) performance of the FASSET system was achieved by correcting a “stuck bit” fault associated with the Downlink Delay Module in the Timing and Control Subsystem. However, further testing after this fault was corrected led to the conclusion that the downlink C0 channel still had a reduced, but measurable BER when data was injected or looped back at the payload TT&C port. This problem was investigated by attaching two Fireberd bit error analyzers, one at the ground terminal C0 user data port, and the second at the payload TT&C data port. The payload and ground terminal were configured for single user uncoded TT&C testing, and more emulations were conducted. After analyzing the Fireberd clock and data signals, it was noted that the residual TT&C downlink BER appeared to be related to incorrect clocking of the downlink, or TT&C C0 transmit data. Investigation revealed a broken transmit clock pin on the TT&C data connector, and the connector was subsequently replaced. Further testing after the repair confirmed that uncoded TT&C data was then running error-free in both uplink and downlink directions.

Further discussion of these TT&C issues can be found on pages 34-37 of [5].

3.2.10 Downlink Expander Fault

The payload Frequency Synthesizer, located in the Timing Control Subsystem, provides hopped and fixed frequency local oscillators (LO) to the payload Transmitter and Receiver Subsystems. The LOs (PLD-RCVR-LO1, PLD-RCVR-LO2, PLD-RCVR-LO3, PLD-XMTR-LO1, PLD-XMTR-LO2) are transferred to the target subsystems through semi-rigid SMA cables on the rear side of the TCS.

The Frequency Synthesizer is composed of three separate components, the Reference Generator, and the Uplink and Downlink Expanders. The Frequency Reference Generator takes a

10 MHz input and generates a fixed LO, a stepped LO, a DDS clock, and LOs for up-conversion. The Frequency Expanders each use frequency control words and LOs to generate up-converted frequency hopped LOs for the transmitter and receiver.

Four key areas within the Frequency Synthesizer Subsystem are protected by automatic fault indication. A parity checker in an erasable programmable logic device (EPLD) validates the input serial command words carrying the uplink and downlink frequency hop words. The average output power level of the uplink and downlink hopped LOs in the expanders are measured and compared to acceptable thresholds by built-in test equipment (BITE). Also, the average output power levels of the LOs in the Frequency Reference Generator are measured by BITE and compared with upper and lower thresholds. Furthermore, the Payload Controller Interface monitors the DC power supplies that feed the Frequency Synthesizer, and any faults are reported to the Configuration Computer.

From the first emulations achieved during refurbishment, the Downlink Expander fault was intermittently present. The fault was difficult to isolate and analyse since it was intermittent and often masked by other faults. Since the effects of the Downlink Expander fault were known, it was possible to troubleshoot higher priority problems, while working within certain constraints. Part way through refurbishment, problems with achieving downlink synchronization eventually raised the priority of the Downlink Expander fault.

In order to further isolate the Downlink Expander fault, its inputs were first verified against the documented specifications. Plots of the Downlink Expander output (PLD-XMTR-LO2) were compared with those of the Uplink Expander. The entire hopping band on the downlink was divided into approximately six frequency blocks, or steps of different amplitude. The Downlink Expander output had low amplitude across blocks of frequencies within the required hopping band. With the failure isolated to the Downlink Expander, the module was replaced with its spare, and the fault was no longer reported. Plots of the spare Downlink Expander's output had reasonably uniform amplitude across the hopping band that resembled the output of the Uplink Expander.

The faulty Downlink Expander module was examined and tested at DRDC Ottawa, and its RF switch matrix was determined to be the failure. A set of appropriate pin diodes and an RF technician are required to troubleshoot the circuit. An operational spare for the payload Frequency Expanders does not currently exist.

Later in the refurbishment process, after many of the overlaying problems were alleviated, a further explanation of the Downlink Expander fault was identified. When emulations are run with Rotate 6 TRANSEC, the power distribution across the hopping bad is not uniform and causes a Downlink Expander fault. Previously, the intermittent failure of the RF switch matrix was compounded with this feature of Rotate 6, making the problem difficult to diagnose.

The associated logbook entries for the Downlink Expander problem can be found throughout [4] and [5], with key analysis on pages 3-8 of [5].

3.2.11 Uplink Expander Fault

3.2.11.1 Background

The payload Frequency Synthesizer, located in the Timing Control Subsystem, provides hopped and fixed local oscillators (LO) to the payload's Transmitter and Receiver Subsystems. The LOs (PLD-RCVR-LO1, PLD-RCVR-LO2, PLD-RCVR-LO3, PLD-XMTR-LO1, PLD-XMTR-LO2) are transferred from the frequency synthesizer (see Figure 8) to the target subsystems through semi-rigid SMA cables on the rear side of the TCS.

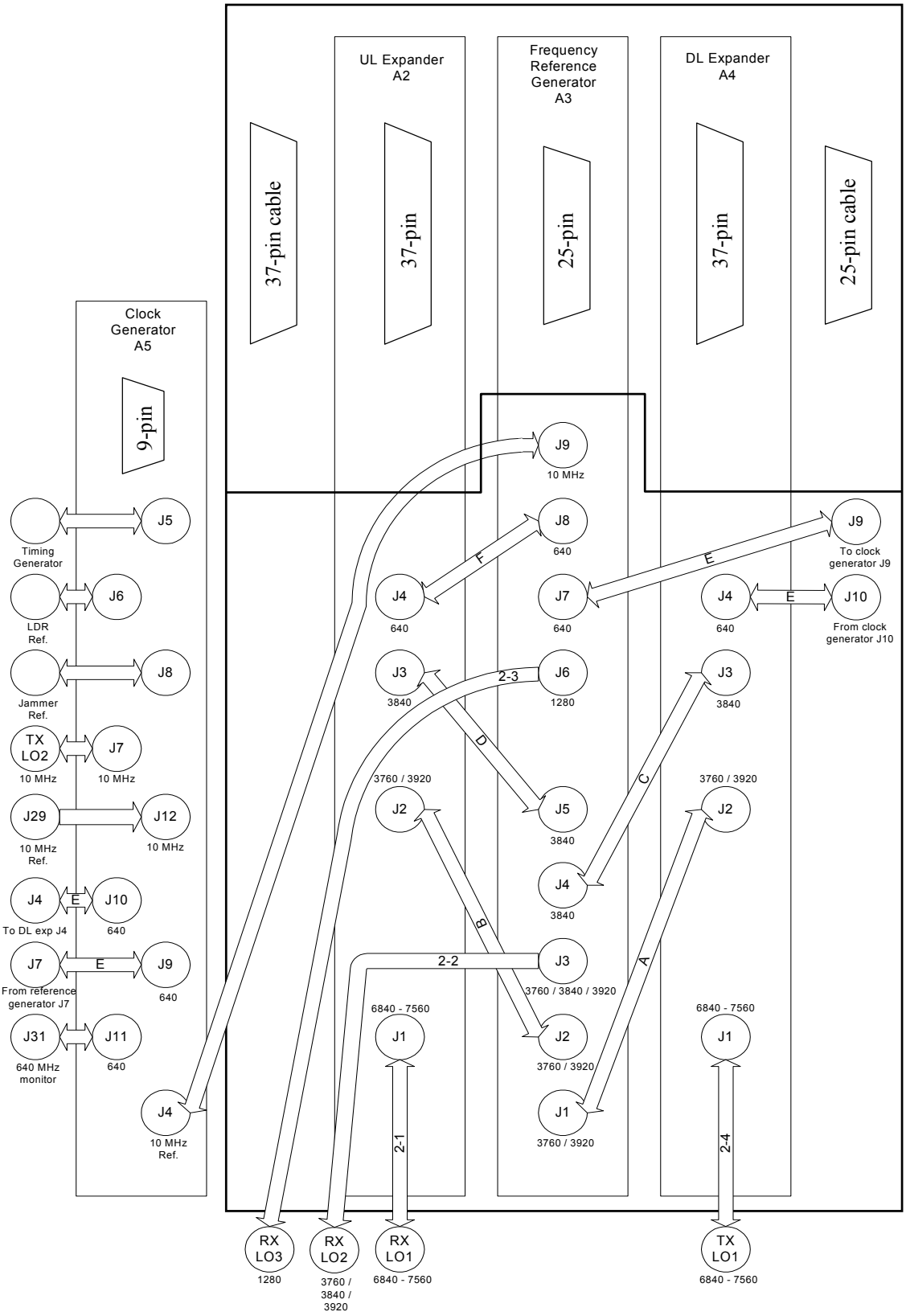


Figure 8 Payload Frequency Synthesizer connection diagram

The Frequency Synthesizer is composed of three separate components, the Reference Generator, and the Uplink and Downlink Expanders. The Frequency Reference Generator takes a 10 MHz input and generates a fixed LO, a stepped LO, a DDS clock, and LOs for up-conversion. The Frequency Expanders each use frequency control words (digital commands) and LOs to generate up-converted frequency hopped LOs for the transmitter and receiver.

DC power and serial commands for the DDS are distributed in the Frequency Synthesizer through an I/O interface box. This box is mounted on the top of the synthesizer, with each of the three modules and the inputs connected on the bottom side of the box. In order to access any of the three modules, the I/O interface box and its attached cover panel must first be unscrewed from the top and removed.

Four key areas within the Frequency Synthesizer Subsystem are protected by automatic fault indication. A parity checker in an EPLD validates the input serial command words carrying the uplink and downlink frequency hop words. The average output power level of the uplink and downlink hopped LOs in the expanders are measured and compared to acceptable thresholds by the built-in test equipment (BITE). The average output power level of the LOs in the Frequency Reference Generator are measured by BITE and compared with upper and lower thresholds. The Payload Controller Interface (PCI) also monitors the DC power supplies that feed the Frequency Synthesizer, and any faults are reported to the Configuration Computer.

3.2.11.2 Fault Analysis

An Uplink Frequency Expander fault began surfacing on the payload during emulations with the standard test file (STANDARD.PAY). Accordingly, the ground terminal reported that it was stuck in UL acquisition. Similar to the Downlink Frequency Expander, the TCS will report a fault if the average power across the band of the uplink hopped LO (PLD-RCVR-LO1) is below a specific threshold. The Payload Controller Interface (PCI) transmits the fault reports across a LAN to the Configuration Computer where it is recorded in the alarm logging system.

The Uplink Expander fault was present during multiple tests over two days, during which the ground terminal was stuck in uplink acquisition. Part way through testing, a TCS power bit #6 fault occurred. This power bit fault indicates that the voltage level of the TCS –12V power supply unit (PSU) is outside the rated threshold window. Typically, for power bit faults the voltage level drops below the lower threshold due to a failure in the PSU. Alternatively, the load may increase, drawing more current than the PSU can provide, causing the voltage level to drop.

In order to verify the BITE, measurements were taken on six of the LO outputs (PLD-RCVR-LO1, PLD-RCVR-LO2, PLD-RCVR-LO3, PLD-XMTR-LO1, PLD-XMTR-LO2, 640 MHz monitor), and the frequency reference input (10 MHz). The resulting measurements were compared with plots taken previously, and the results showed that both the uplink and downlink hopped LOs were low by 50 dB. Later testing showed that a fault in the –12 V PSU caused the hopped uplink and downlink output levels to drop, while also masking a secondary fault in the Uplink Expander.

Since the TCS was continually receiving faults on power bit #6, the decision was made to replace the –12 V PSU with a new PSU from International Power. The new PSU was tested at the bench in a loaded and unloaded state prior to installation. With the new PSU installed, the monitor port documented in Section 4.2.3 was added to allow easy measurement of internal power supply voltages.

The LO measurements were repeated, and a new set of plots was captured. The results showed that all of the outputs were within specifications, except PLD-RCVR-LO1 was low by 14 dB. The power bit #6 fault was no longer present, but an Uplink Frequency Expander fault was consistently reported. An HP bench amplifier was used to increase the power level of the ailing hopped LO at the output of the TCS, and emulations with full tracking were successfully performed. In order to further isolate the problem, the Uplink and Downlink Expander boards

(which are identical) were swapped, and emulations showed that the problem moved accordingly to the downlink side.

On the assumption that the DDS board was damaged by the –12 V PSU, the DDS boards of the Uplink Expander and the spare expander were swapped. Swapping had no effect on the LO power levels, so it was decided to leave the spare DDS board in the Uplink Expander module.

With the TCS in the equipment rack and the Uplink Expander connected by extensions outside the rack, the power supply voltages were measured across the board. The voltage levels were within specifications, although some were slightly low, possibly due to the cable extension. In order to extend the Uplink Expander module while keeping the remaining Frequency Synthesizer components in their normal locations, a two-foot section of 37-pin ribbon cable was utilized (see Figure 8). The RF chain (see Payload Frequency Expander drawing #SDG537202 in [16]) was then measured using a makeshift RF probe and a spectrum analyzer. The power level measurements were consistently low until point SR12 was probed. Point SR12 is a transition type of connection between the DDS printed circuit board (PCB) and the analog micro-strip board. After touching up that connection and also bulkhead connector J2, the uplink hopped LO output improved to nominal levels. All of the LO outputs were measured again, and full tracking was achieved with the standard emulation.

3.2.11.3 Post-Repair Analysis

It is not possible to conclude if the failure of the –12 V PSU was related to the failure within the Uplink Expander. When the problem was sufficiently diagnosed to allow troubleshooting to proceed, the symptoms of the –12 V PSU failure were already masking those of the Uplink Expander. The only definite conclusion is that the BITE automatic fault indicator worked successfully in identification and isolation of the problem.

It is possible that running the TCS too hot over an extended period of time caused these faults. Alternatively, they could have been due to age or humidity. Another factor may have been mechanical stress incurred during physical relocation of the system to Melbourne.

Of the three Frequency Expander boards in existence, two are fully operational and one is in need of repairs and reassembly. A failure in the 3x2 switch matrix has been identified, and currently awaits repairs by a qualified RF technician. A single payload Frequency Reference Generator is installed in the TCS, while its spare is currently in storage and its status is unknown.

During testing, a discrepancy between schematics and actual connections was noted with regards to the payload Frequency Synthesizer. In Figure 8 (this figure depicts the connections as they currently exist), the SMA coaxial cable connected to J1 on the Frequency Reference Generator (module A3) should be connected to J2, and the cable at J2 should be connected to J1. Similarly, the cables connected at J4 and J5 should also be swapped. From the point-of-view of the Uplink and Downlink Expanders there is no difference because the signals on the swapped connectors are functionally identical.

In the process of investigating the Uplink Expander failure some recommendations for improving the TCS have resulted. The addition of 6” sections of flexible cabling on the 37-pin and 25-pin connectors of the Frequency Synthesizer will improve accessibility for future troubleshooting within and near that module. Replacement of the SMA semi-rigid cabling on the top of the Frequency Synthesizer has been recommended as these cables have been worn by use, and have caused some problems. Another recurring problem has been the loosening of SMA bulkhead connectors on the rear panel of the TCS. Non-rotating bulkhead connectors are recommended to prevent twisting and bending of internal cabling.

Power supply failures in the TCS have become an increasingly important issue and a recurring problem. It is recommended that to improve accessibility in the subsystem and reduce thermal emissions, a separate external power supply rack-mounted tray should be considered.

More detail on the Uplink Expander fault can be found on pages 36-43 of [6].

3.3 Ground Terminal

In this subsection, the repaired items that are discussed involve faults associated with the ground terminal portion of FASSET. The topics covered include battery backed-up RAM, hard disk corruption, the X-terminal, and the Hyperbus interconnect. Repairs made to the uplink attenuator and the Frequency Reference Generator are also described.

3.3.1 Replacement Battery Backed-up RAM for Single Board Computers

Since the FASSET system was placed in storage in 1997, one of the inevitable obstacles to refurbishment was the failure of battery-backed-up RAM (BBRAM) or non-volatile RAM (NVRAM). In both the payload and ground terminal, BBRAM is used to retain values in memory. In some cases the BBRAM is used, in others, it resides on the boards unused. During the initial phase of refurbishment, it became apparent that the BBRAMs of the MVME147SB boards in the ground terminal are used to store vital boot-up parameters.

The original BBRAM of the MVME147SB boards was Mostek MK48T02B-25. These rectangular devices have a lithium battery mounted on the top of a timekeeper and RAM module. Their lifespan is approximately 5-10 years depending on usage. These particular devices were manufactured in the 1992 to 1997 timeframe, thus explaining why they had expired by 2002.

At the start of refurbishment, during initial attempts to start-up the ground terminal, bus errors were encountered during the boot phase. The bus errors interrupted the booting, and forced the user to shutdown the ground terminal while the system was still on. The source of the bus errors was unknown, although the BBRAMs were suspected.

Attempts were made to program each of the BBRAMs using the instructions provided on pages 11-14 of the Ground Terminal Technical Manual [12]. It was not possible to reliably write to memory then read the values back, indicating that the BBRAMs had failed. Each BBRAM exhibited the same symptoms, so the decision was made to replace them with a newer and faster model of the same timekeeper, the M48T02-200PC1.

The replacement BBRAMs were successfully tested and were able to hold their memory. Once they were programmed according to [12], subsequent boots resulted in various bus errors including a stray interrupt. The cause of the stray interrupt was suspected to be either a hardware failure or an error in the BBRAM settings. There were some inconsistencies between the instructions and some hand annotations found in the manual. To alleviate this uncertainty, a VMETRO bus analyzer and the VME specification documentation were used to study the bus-mastering scheme that is implemented in the ground terminal.

In an attempt to eliminate any hardware failures and bus conflicts, all of the ground terminal boards were removed from the VME bus, and only the system controller was installed. The ground terminal was still unable to boot. The bus errors eventually disappeared when the hard drive was replaced with a restored set of software from tape backup. Finally, after programming the BBRAMs following the procedures outlined on pages 100-101 of [3], the system was then able to successfully boot to the login prompt. The final BBRAM configuration is documented in those procedures.

The associated logbook entries for the problems with the BBRAMs can be found on pages 1-57 of [4].

3.3.2 Hard Disk Corruption

As described earlier, the FASSET ground terminal software is installed on a single 500 MB SCSI hard disk drive (HDD) mounted in a removable caddy in the ground terminal VME chassis (See [3], Section D.3). At system start-up the System Controller (SC) board, one of three

MVMV147SB-1B single board computers in the chassis performs a ROM-based bootstrap of the Unix System V operating system (OS) installed on the SCSI HDD. When the Unix OS finishes loading, it initiates an X-Windows server task, and progresses to execution of the main FASSET ground terminal software.

When the ground terminal chassis was initially reassembled after being in storage for a number of years, there were a number of problems encountered in attempting to get the Unix system to start up on the original HDD. After a number of repairs, including the replacement of the BBRAM modules on all of the MVME147SB-1B computer boards (Section 3.3.1), it was determined that the software installed on the 500 MB SCSI HDD had become partially corrupted, and would no longer run either the Unix System V OS or the FASSET application program. After consulting references [17-21], a number of replacement HDDs were located and procured which were compatible with the Motorola Unix System V/68 OS. Some drives had to have modified control ROM's installed so that they would be compatible with the boot ROMs of the MVME147SB-1B computer boards.

A procedure was developed to perform a complete HDD system generation of both the System V/68 Unix OS, and all of the FASSET application modules working from a set of QIC-150 backup tape cartridges (see Section 6.4). Once this process was completed, the newly generated HDD was designated as a GOLD, or reference system, and was used as a source drive to make multiple additional copies of the system software on other compatible SCSI HDDs. The HDD copy procedure is documented on pages 102-112 of [3], and also in [22]. It depends upon the availability of a VME development system with a set of removable hard drive trays which are compatible with the drive trays installed in the FASSET ground terminal chassis. This equipment is further discussed in Sections 4.3.1 and 4.3.2 below.

3.3.3 X-terminal Failure

Each ground terminal includes a corresponding X-terminal for controlling setup and execution of emulations. The X-terminal runs X-windows with a keyboard and mouse for the user to manipulate settings and communicate with the server running on a Unix system.

Early in refurbishment, with no warning, the LGT2 X-terminal began to produce smoke upon power-up. The system was switched off and unplugged from its power receptacle in order to limit the damage. Inspection revealed that the monitor of the X-terminal was no longer operational, but the main body of the system was still functioning properly. An attempt was made to obtain repairs or a replacement monitor, with no success. In the process, the failed monitor was lost, and a replacement monitor was not available.

The X-terminal emulator called Hummingbird Exceed V8.0.0 was procured and setup on a laptop with a network connection to the Unix system. The Exceed software provides all of the necessary functional features to emulate an X-terminal. The block diagram in Figure 9 shows where the X-terminal emulator is connected in the ground terminal system layout. In order to connect the laptop to the ground terminal network, a media converter was required to switch from 10base2 (thinwire) to the 10baseT (RJ45) standard that is required by the laptop. The X-terminal monitor from LGT1's X-terminal was connected to LGT2's main body, and the settings were recorded for use in setting-up Exceed. Each ground terminal is identified by a unique IP address (LGT1-192.199.120.182, LGT2-192.199.120.183), so the laptop running the X-terminal emulator had to be setup accordingly.

Emulations were successfully run using the X-terminal emulator, validating the replacement system and its setup. Since the emulator runs on a laptop, the convenience of Windows utilities were also available for screen captures, printing, and word processing.

The associated logbook entries for the X-terminal fault can be found on pages 32-33 and 76-87 of [4].

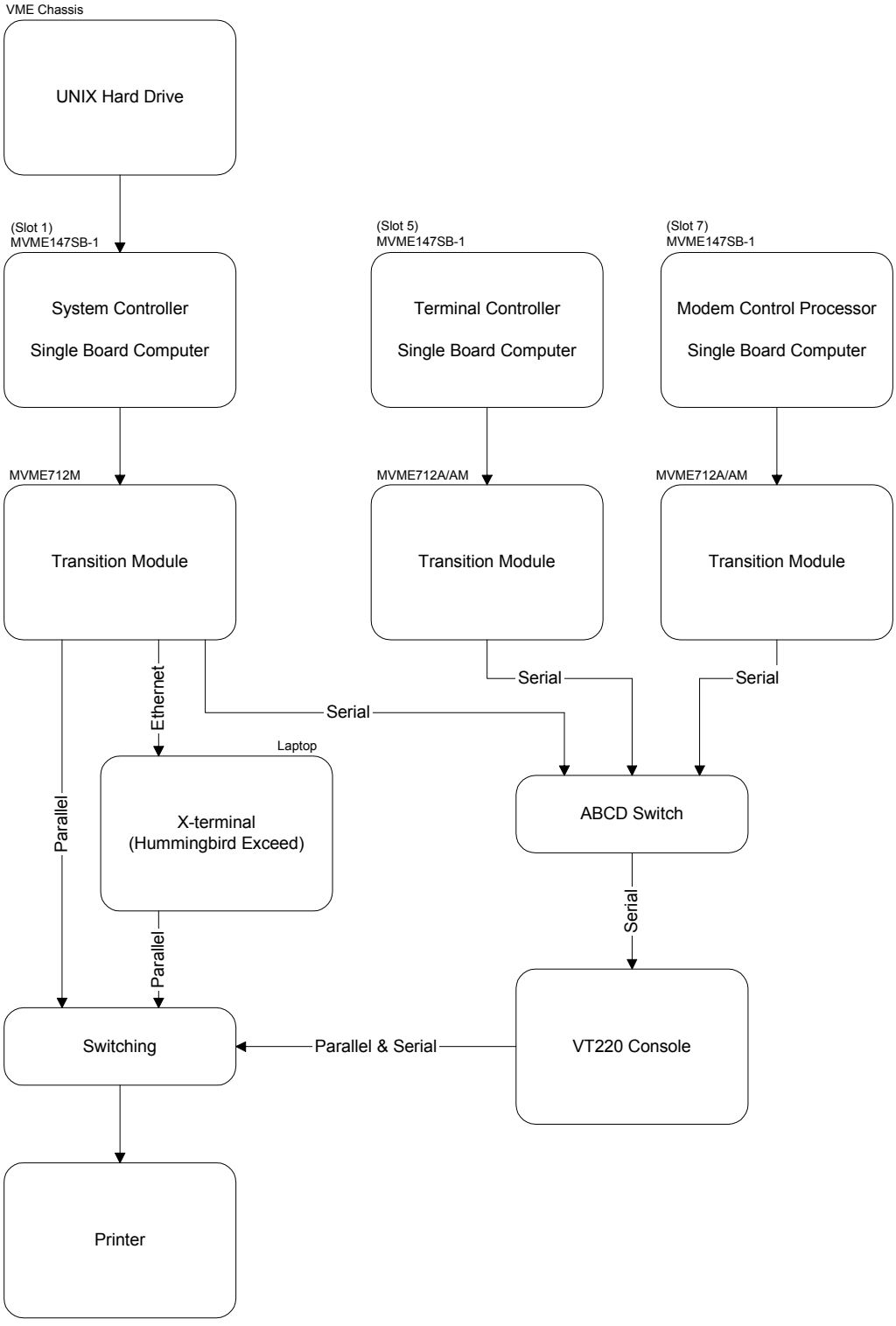


Figure 9 Ground terminal system layout

3.3.4 Hyperbus Interconnect – Short Circuit

The Hyperbus is a high-speed data communications bus between three of the FASSET ground terminal VME boards: the Downlink Processor#1 (DLP#1), the Downlink Processor#2 (DLP#2), and the Downlink Buffer (DLB). The bus is physically connected in a cutout at the bottom edge of the boards, as in Figure 10 below, completely independent of the VME bus. The Hyperbus board must be connected before the three boards are installed in the chassis. Once the Hyperbus is connected, the three boards are handled as a single entity when being installed and removed from the chassis.

When FASSET was being prepared for delivery to Harris Corporation, all of the ground terminal boards were removed and catalogued. To read the serial numbers of the Downlink Processors and the Downlink Buffer, the Hyperbus had to be disassembled at a laboratory bench. Prior to the cataloguing, the three Hyperbus connections were solid, as they had been attached for several years. After being disturbed during cataloguing and during shipment, the Hyperbus connector became loose. Upon first power-up, the ground terminal produced a slight burning odour that lasted a couple of minutes. The source of the smoke was difficult to locate, but after a thorough search, the Hyperbus circuit board was found to be loose. Although no burn marks were evident upon inspection of the Hyperbus, it was suspected as the source of the smoke, because once it was tightened the overheating stopped and the ground terminal start-up proceeded normally. Successive power-ups yielded no further problems from the Hyperbus, although a loose connector was later suspected when downlink synchronization problems began to become prevalent during emulations (see Section 3.3.6). That problem was eventually linked to a failure in the Frequency Reference Generator.

The associated logbook entries for the failure in the Hyperbus interconnect can be found on pages 0-28 of [6].

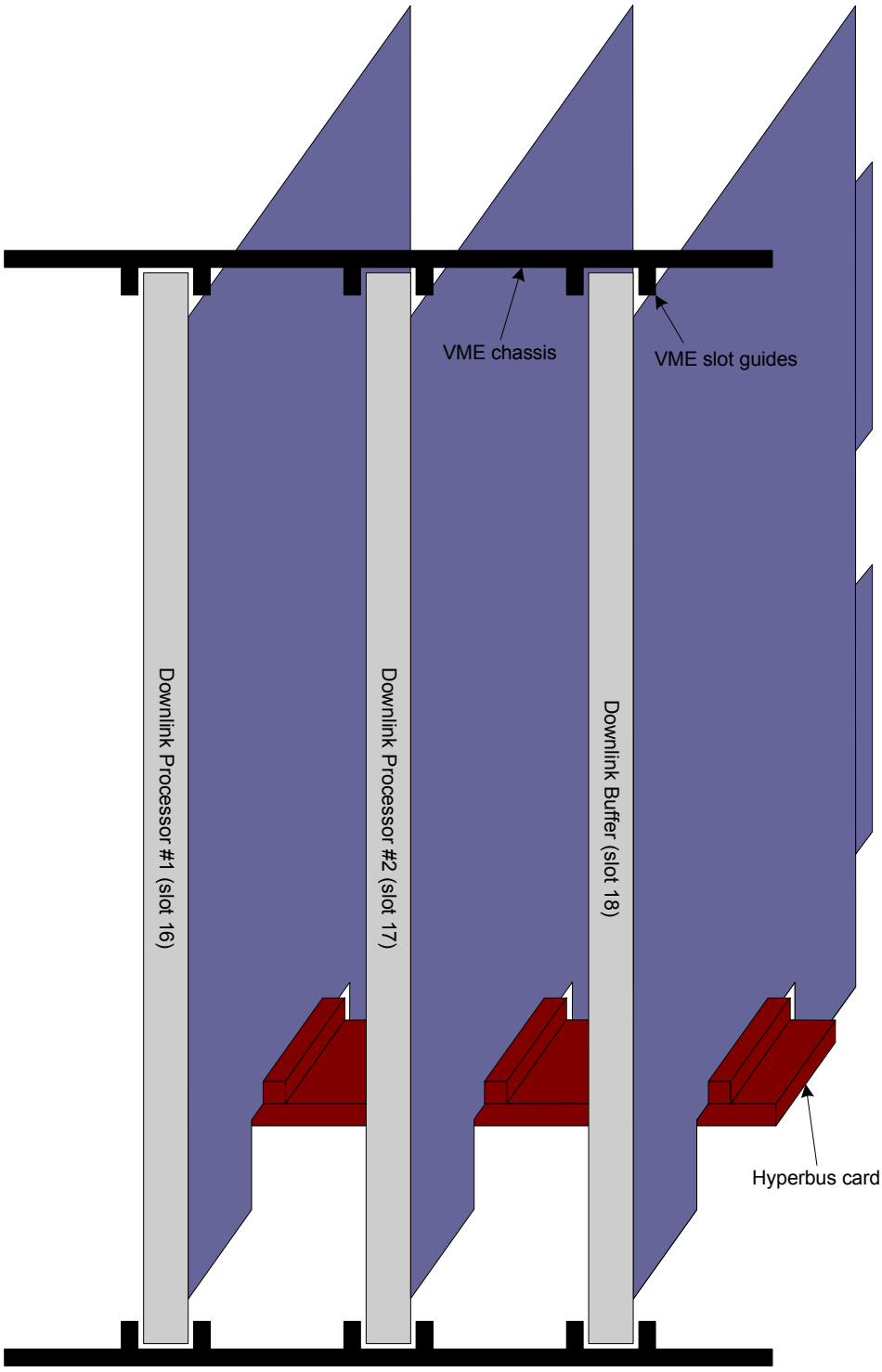


Figure 10 Physical layout of the Hyperbus (front view)

3.3.5 TRANSEM Uplink Attenuator Failure

Several problems were encountered during emulation testing with the FASSET system. The discussion below concerns significant failures that were experienced during the process of uplink coarse synchronization. Symptoms related to this type of problem often appear at the system level, and can be traced to low power levels in the RF uplink signal from the ground terminal to the payload. Information on the preliminary analysis of these uplink synchronization problems is provided in Section 3.1.3.

The investigation of power levels in the uplink RF chain led to the discovery of a problem in the transmitter section of the ground terminal Transceiver Emulator (TRANSEM). The TRANSEM is a hardware component of the ground terminal, which includes the final upconversion stage and the first downconversion stage of the RF front end. A programmable output attenuator in the TRANSEM appeared to be intermittently producing excessive attenuation. To alleviate this problem, the failed attenuator was replaced and a manual waveguide attenuator was attached to the output of the TRANSEM's transmitter port. The manual attenuator was adjusted to a nominal setting and locked for normal emulations.

With the problems resolved in the uplink RF circuit, subsequent emulation testing was able to progress fully through downlink and uplink synchronization into the "Uplink Tracking" state, which allows user data to be passed.

More details on this series of problems can be found on pages 8-17 of [5].

3.3.6 Frequency Reference Generator – Out of Lock

After delivery of FASSET to Harris Corporation in Florida, the ground terminal was occasionally stuck in downlink acquisition. Usually a restart or power down resolved the issue enough to temporarily enable emulations so further investigation was not immediately required. At one point, the ground terminal was stuck in downlink acquisition continually and a trip was made to resolve this issue.

A number of common hardware problems were checked first – the integrity of the hard disk, connections involving the Hyperbus, the X-terminal configuration (which laptop and which version of the operation system), verifying the cryptographic sources, and cycling the power on both the ground terminal and payload. None of these tests yielded a solution.

The VMETRO Bus Analyzer was installed in the ground terminal and it was triggered by synchronization events. Traces and event logs were collected and examined with no conclusions.

The 70 MHz intermediate frequency (IF) signal at the ground terminal was examined on the spectrum analyzer using the logic analyzer synchronization trigger at the payload. The synchronization hop was observed only rarely (it should be there more often). The ground terminal synthesizer command words were examined using the logic analyzer and were found to be present roughly 50% of the time. It was not known if that was normal. Because of suspicious behaviours, the ground terminal synthesizer controller board (DDS) was examined. Minor touch-ups were made to the solder and insulation on the board. The board was reinstalled and tests run, resulting in no improvement.

It was noted in the ground terminal event log that there were alarms from the time-of-day (TOD) board first-in, first-out (FIFO) units. The TOD board holds FIFOs that send data to the DDS board and it was suspected that the FIFO was overflowing. The TOD board was extracted and put on a VME extender (and the SMA connecters on the front panel were also extended). The logic analyzer was used to probe the FIFO (U11 on the board) and trigger on the FIFO full bit. An emulation (stuck in downlink acquisition) was run but there was no trigger and therefore no overflow. The logic analyzer was then set to trigger on FIFO empty. It was again stuck in downlink acquisition but the trigger occurred, indicating that the FIFO underflowed.

Boards were swapped including the cryptographic controller board (Key Generation Controller) and the demodulator. TRANSEC simulators were swapped between ground terminal and payload. None of these swaps helped.

Traces and event logs were obtained for good (achieved tracking) and bad (stuck in downlink acquisition) emulations. Captured trace data was interpreted using the information found in Annex A.2. Examination of the capture traces showed that the FIFO underflow was a consequence of the downlink keystream being disabled. It was noted that there was a problem with the documentation of the FIFO interface on the TOD board. The register value for FIFO underflow was reversed, and incorrectly indicated an overflow error.

The VMETRO was moved to slot 11 to allow bus grant signals to be monitored. Several traces were captured for bad emulations. In an earlier emulation, a Terminal Controller (TC) error was found in the ground terminal event log, so this board was swapped, resulting in no improvement. The downlink attenuator RGT-DL-LEVEL on the payload was set to 990 (maximum attenuation, so no signal was received by the ground terminal) and traces were captured. It was also set to 0 (minimum attenuation) and various points in between, resulting in no improvement.

By chance, a good emulation was captured and compared to previous bad ones. Comparison of the traces highlighted that fact that the synchronization hops were not being detected in bad emulations. It was only at this point that the FASSET Classified Notes became available (see Section 7.2 for details) which provided the frequency data for the Rotate 6 TRANSEC.

The DDS command words were captured using the VMETRO and the hopped LO frequency was calculated. The ground terminal LO was examined with a spectrum analyzer on it, and was seen to be stepping and jumping. Hopping was turned off (which fixes the LO at the centre of the hopping band). The ground terminal LO was still off frequency and moving. The ground terminal Frequency Expander was swapped with no improvement. It appeared that the Frequency Expander was not locked. The Frequency Reference Generator provides several frequencies to the Expander, so these were examined with a spectrum analyzer. It was then noticed that the 8640 MHz LO from the Frequency Reference Generator was not locked and was stepping. Also it was noticed that the Downlink Expander LED on the front panel of the Jammer/Synthesizer was blinking rapidly with a 50% duty cycle. The other LOs were checked as well, and it was noted that they were not as far off frequency but still moving. The Frequency Reference Generator was swapped with spares but the problem remained.

The Frequency Reference Generator was opened up and the inputs probed. The 5 MHz reference and DC voltages were all correct. The phase locked loop (PLL) portion within the Frequency Reference Generator was not working, yet it had good inputs. The board was examined and touched up. It was then replaced and heated using a heat gun to thermally modify its lock range. After approximately 15 minutes, the 1080 MHz LO (along with the others) locked up. The 1080 MHz LO is available for observation on the back of the ground terminal through a coupler. Once the PLL was locked up, the ground terminal was able to synchronize properly and pass data.

Later an RF expert at Harris fixed this board by modifying the lock range for ambient and operating temperatures. It was installed in the ground terminal and worked successfully.

Similar attempts were made to fix the other boards (spare and LGT1), with less success. There is only one good board at this point.

The associated logbook entries for the PLL problem in the Frequency Reference Generator can be found on pages 12-23 (plus two pages of diagrams) of [6].

4. Modifications and Improvements

The topics in this section pertain to hardware and software modifications and improvements that were added to FASSET during the refurbishment of the system for Harris Corporation. The topics are categorized into payload and ground terminal subsections. As in Section 3, the subsections may be read independently.

Significant improvements are described that facilitated troubleshooting, increased maintainability, and enhanced FASSET capabilities to meet Harris' requirements. The primary goal of refurbishment was to bring FASSET back to the level of operation and functionality that it possessed prior to being placed in storage. Any modifications and improvements were implemented at the request of Harris, or if they were deemed imperative by the troubleshooting team.

4.1 System

System level improvements are those that include equipment and software that spans both the payload and ground terminal.

4.1.1 Standard Configuration Files Implemented

Configuration files are text files that specify the parameters for a given emulation or test. When delivered, there was a standard test that was run routinely to verify the working of the FASSET system. This test, called STANDARD, was stored in a payload configuration file of that name (and a corresponding configuration file on the ground terminal). While most tests were done using this configuration from delivery to mid-way through the refurbishment, it had one significant shortcoming. It used an FSK/1 mode on the uplink which was part of the FASSET extensions to the waveform and not part of standards [1] or [2]. As a consequence, it was determined that this would not by useful with the Harris equipment. This configuration was renamed OLDSTAN and a new STANDARD was developed that used FSK/2, which conformed to [2].

A version of STANDARD with coding turned off is called UNCODED. Another variant of STANDARD was created called STANDARDH. For the payload, this test is identical to STANDARD. For the ground terminal, this configuration uses the high hop-rate synchronization hop during acquisition instead of the low hop-rate synchronization hop. STANDARD, STANDARDH and UNCODED are used primarily when debugging, and are under DRDC Ottawa configuration control.

These configuration files and their unclassified parameters can be found in Annex C.

4.2 Payload

In this subsection, improvements made to payload equipment and software are described in detail. The payload improvements described below include the configuration file format, the external loopback, and the monitor port for power supplies in the Timing Control Subsystem. Also discussed in this subsection are the addition of uplink to downlink mapping for TT&C and C1 messaging, and the addition of support for multiple user data accesses. Finally, the implementation of flexible C3 message sources is addressed.

4.2.1 Configuration File Format Modified

As described in 4.1.1, payload configuration files are text files that specify the parameters for a given emulation. Default parameters are stored in one FASSET file "__EMDATA.FST" whereas the other files (*.PAY) are read in via the user interface.

As delivered, FASSET did not support any selectable downlink assignments. Downlink assignments were determined by the uplink access using a fixed mapping (see Section 2.1.4.3). As a consequence, there were no database entries (nor associated entries in the configuration files) for downlink access assignments. When modifications were made prior to interoperability testing at Lincoln Lab in 1997, fields were added for the downlink assignments for the four user data accesses (though three of the four were not yet used).

For flexible mapping of C0 there were no changes required in the file format. Since C1 data was an option on the C0 configuration, it too did not require any changes. For flexible mapping for TT&C, the following fields were added to the default parameter file __EMDATA.FST and to each *.PAY file:

a. EPD_LDR_TTC_DL_MOD
b. EPD_LDR_TTC_DL_SPARE
c. EPD_LDR_TTC_DL_HOPS (10 entries)
d. EPD_LDR_TTC_DL_RAW

To allow for increased database size, the size of the EPD_Supplement section in the database EPD_Supplement_Entries was increased from 52 to 65 in the file EPDOFSET.H. The maximum number of database entries, H_MAX_NUM_EMUL_REC, was increased from 907 to 920 in the file HMIDEFS.H. The database initialization routine InitEPD_LDR in the file INITEPDL.C was augmented by including support for the new TT&C entries. The process for regenerating the LDR Symbol Processor download software can be found in Section 6.3.2.

4.2.2 External Loopback Capability Added

One of the diagnostic modes originally supplied with the FASSET system was the ability to reconfigure the payload so that a limited number of payload tests could be performed without the involvement of the ground terminal. As described on pages 113-119 of [11], the payload loopback configuration alters the payload downlink waveform so that it emulates a simplified version of the normal ground terminal uplink waveform. When converted in frequency from the 20 GHz downlink band to the 44 GHz uplink band, this payload "loopback" signal is injected into the payload uplink receiver port, and is processed by the payload uplink processor as if it originated from the ground terminal. In order to achieve the necessary wide band frequency conversion, a special mixer is required. The mixer originally provided for this purpose (Watkins Johnson SMC-1844) could not be found, but a replacement unit with similar specifications (Miteq TB0400LW) was identified and used to implement the loopback configuration. Details on the loopback setup and operation are described on pages 92-95 of [3].

As originally configured and tested, three modes of operation can be selected under the loopback configuration:

a) LDR Acquisition Loopback – intended to test the payload uplink acquisition of coarse and fine synchronization probes.
b) LDR Access Control (AC) Loopback – intended to test the payload processing of uplink C2 messages by verifying the CRC field on all uplink C2 messages.

c) LDR C0 Loopback – intended to test the payload processing of both externally supplied and internally generated user C0 messages.

When loopback testing was initially performed on the payload, only the LDR Acquisition mode was found to be operational. The AC loopback mode reported 100% errors over the entire range of simulated uplink signal to noise levels, and the C0 loopback mode failed with a "TT&C Symbol Readback Fault" after every attempt to pass C0 messages. These problems were not unexpected, as some previous payload processing changes had been made to fix problems not associated with loopback mode, and the diagnostic modes had not been tested after those changes.

Since a number of payload processing problems had already been identified for correction under the FASSET refurbishment contract, the loopback problems were noted on pages 18-20 of [5], but put on hold until the higher priority work was completed on the payload.

During the subsequent few weeks, extensive changes were implemented and tested to ensure that the payload could properly receive and validate CRCs on uplink C2 messages generated by the ground terminal in normal (non-loopback) configuration. Additional work was then performed to correct a number of issues related to processing C0 messages through the payload TT&C port to and from the ground terminal, and to implement the multi-user data mode on the TT&C port (see Section 3.2.6 for details). When these payload modifications were completed and adequately tested with the ground terminal, the payload was again reconfigured for loopback diagnostic mode. Subsequent testing established that all three loopback test modes were operational, and that synchronization probes, C2 and C0 messages could all be passed through the payload with error rates consistent with emulation signal to noise ratio settings.

4.2.3 Timing Control Subsystem Power Supplies – Diagnostic Monitoring Implemented

In order to measure the voltage levels of internal power supplies without opening the Timing Control Subsystem (TCS), a monitor port was created. During both normal operations and troubleshooting, there is often a requirement to know the power supply voltage levels within the TCS. Normally, the Payload Controller Interface (PCI) monitors these critical voltage levels by comparing them with thresholds, but if the PCI and Configuration Computer are not fully functioning, direct measurements are necessary. Unfortunately, the TCS is a heavy and bulky unit that requires four people to lift. Even with care, moving the subsystem causes detrimental jarring of modules and cables within. Every time the TCS is removed from the rack, a series of cables must be disconnected at the rear, which adds to the cumulative wear of the connectors and cables.

The TCS is a densely packed subsystem with modules and cables that are worn by age and from past manipulation. The power supply section is especially troublesome as it is organised into two vertical layers. The top layer of power supplies can be removed, with difficulty, for access to the supplies below. Once the top layer is disconnected and removed from the TCS, it is no longer available for simultaneous testing in combination with the lower layer. In order to ease the monitoring of power supplies in the TCS, the monitor port in Figure 11 was installed on the rear panel of the TCS, and an intercept cable was placed in line with the J3 connector of the PCI module. This monitor cable allows the operator to measure all of the power supplies in the TCS with external probes. This is a significant improvement to the former situation where a significant amount of agitation was incurred on the internal harnesses and cabling each time a power supply failure was investigated.

The associated logbook entries for the power supply monitor port of the TCS can be found on page 39 of [6].

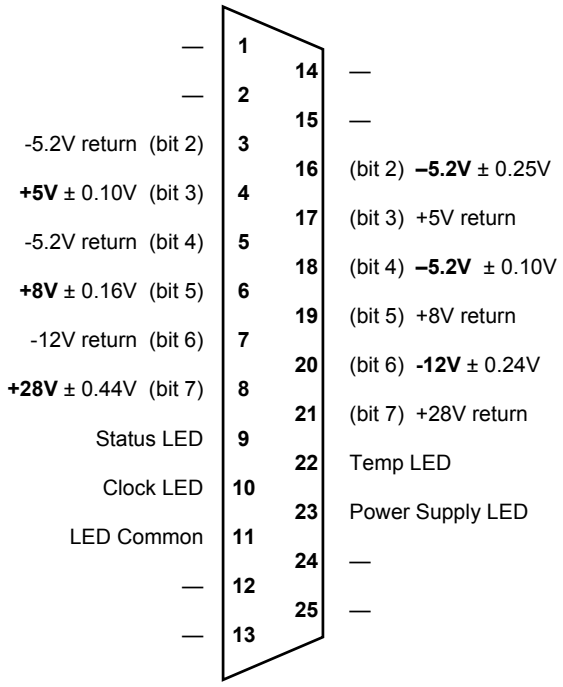


Figure 11 Payload TCS power supply monitor

4.2.4 Support for Multiple User Data Accesses Added

Prior to interoperability trials at Lincoln Lab in 1997, the decision was made to concentrate on a single C0 data channel to the exclusion of all others because C0 was essential for interoperability trials. When modifications were made to FASSET to support the trials, the other communications modes (multiple C0's, TT&C, and C1) were considered to be less important and no tests of these types were run. The modifications were made only to support flexible mapping for a single C0 channel and the software code for these other types was not accordingly updated.

Some work had been done previously to support multiple user data accesses. At the same time as the changes were made to support a single C0 channel, the data base entries were made for all channels but no other changes were made for multiple channels. One of the parameters included in the downlink entries was a diversity parameter that was never used. Rather than remove the parameter it was decided to make it a spare. The first change to the screen library was in file EPDOFSET.H to rename the parameter from EPD_LDR_AC1_DL_DIVERSITY to EPD_LDR_AC1_DL_SPARE (similarly for AC2 to AC4). In the file GSCCCNFG.C it was also necessary to rename EPD_LDR_AC1_DL_DIVERSITY to EPD_LDR_AC1_DL_SPARE (similarly for AC2 to AC4). The structure element .DL_Diversity was renamed to .DL_Spare. The structure renaming was also performed in SEQDEFS.H.

Changes were made to the user interface to allow configuration of the parameters necessary for multiple user data accesses. In the file HMIDEFS.H, it was necessary to increase the maximum number of fields used by the user interface screens. The techniques and associated pitfalls are detailed in Section 6.2.3. It was during these changes that the software tool CountFields (see Annex D.2) was developed. The name of the action token was also changed from the more general H_A_HOP_TO_BIN to a name specific for the first user data access H_A_HOPBIN_AC1.

New tokens were added for the other user data accesses (AC2-AC4). In the file ACTION_H.C, the token renaming was also done. The old "hopbin" action and associated

subroutines originally were hard-coded to implement conversion of text field hops to binary only for the first user data access. The routines were changed to accept a parameter specifying the access number, so that "hopbin" could be invoked for each of the four actions associated with the four accesses. The payload executable and screen library were then regenerated as described in 6.2.1 and 6.2.2.

The user interface screens were changed to allow independent uplink and downlink configuration for all four of the user data accesses. In the file __HMIDEF.FST, the screen for user data access #1 was modified to include uplink details, and to retain old downlink details. It was also reformatted and hints were added. Three similar screens were added for user data accesses #2-4 details. An unrelated change made at this time was to enhance the Emulation Active screen by the addition of a parameter to display the current configuration file name. The original LDR communications parameter screen was reformatted and modified to link to the other screens (including TT&C details), add entry hints, and move some details to other screens. No recompiling was necessary for the user interface screens.

The associated logbook entries for the multiple C0s can be found on pages 45-47 of [5].

4.2.5 Uplink to Downlink Mapping Enhanced for TT&C Messaging

Prior to interoperability trials at Lincoln Lab in 1997, the decision was made to concentrate on a single C0 data channel to the exclusion of all others because C0 was essential for interoperability trials. When modifications were made to FASSET to support the trials, the other types (multiple C0's, TT&C, and C1) were considered to be less important and no tests of these types were run. The modifications were made only to support flexible mapping for a single C0 channel and the software code for these other types was not accordingly updated.

Once TT&C was fixed (see Section 3.2.6), it was desirable to have flexible uplink to downlink mapping. First TT&C downlink entries had to be created for the database. The screen library was modified in file EPDOFSET.H by extending the length of the EPD_Supplement section of the database by changing EPD_Supplement_Entries from 52 to 65 (13 entries added for TT&C). The entries added after EPD_LDR_AC4 were: EPD_LDR_TTC_DL_MOD, EPD_LDR_TTC_DL_SPARE, EPD_LDR_TTC_DL_HOPS (10 entries), and EPD_LDR_TTC_DL_RAW. In HMIDEFS.H, changing H_MAX_NUM_EMUL_REC from 907 to 920 extended the total number of database entries. Also in that file, a new action token H_A_HOPBIN_TTC with a value of 44 was added.

Database initialization was modified in file INITEPDL.C in the routine InitEPD_LDR by adding TT&C downlink entries that were temporarily the same as the downlink entries for user data access #4 (AC4). The configuration messages sequencer routine GSCCCNFG.C was changed to use TT&C database entries when generating TT&C configuration messages. While making these changes the same fix was made for maintenance TT&C in the code (even though these have never been used by DRDC). A new action was added for H_A_HOPBIN_TTC in the file ACTION_H.C that converts text input for hops used to the binary format needed for the message. The screen library was then regenerated as described in 6.2.2.

Changes were then made to the LDR Symbol Processor download software. In the file data formatter LRDATAFO.C, the TT&C section was copied from the C0 section, renamed for TT&C and then simplified to only deal with 2400 b/s. A minor change was made to SYMMBOX.H, to rename DL_Diversity (which wasn't used) to DL_Spare. The process for regenerating the LDR Symbol Processor download software can be found in Section 6.3.2.

The user interface screens were changed to allow independent uplink and downlink configuration for TT&C. In the file __HMIDEF.FST, a simplified (because only 2400 b/s is supported) version of the C0 access #1 screen was used and renamed. The new action token was used. The screen changes did not require any recompiling.

The final modifications were made to the configuration files on the payload. A copy of the user data access #4 downlink entries was added and renamed for TT&C. The standard test configuration STANDARD.PAY was modified to include the new entries as well. All other test configuration files (named *.PAY) were modified similarly. More details can be found in 4.2.1.

The associated logbook entries for the TT&C flexible mapping can be found on pages 46-47 of [5].

4.2.6 Uplink to Downlink Mapping Enhanced for C1 Messaging

Prior to interoperability trials at Lincoln Lab in 1997, the decision was made to concentrate on a single C0 data channel to the exclusion of all others because C0 was essential for interoperability trials. When modifications were made to FASSET to support the trials, the other types (multiple C0's, TT&C, and C1) were considered to be less important and no tests of these types were run. The modifications were made only to support flexible mapping for a single C0 channel and the software code for these other types was not accordingly updated.

When Harris indicated a desire to have C1 access capability, a code inspection revealed two problems. The first problem was that the LDR Symbol Processor download software (LRDATAFO.C), which formats the downlink data, processed C1 data using the fixed mapping. This was changed to allow flexible uplink to downlink mapping. The second problem was that the module FORMATDP.C, which assigns downlink data bits to hops, did not support multiple hops for C1 using DPSK. This feature was added and verified. The process for regenerating the LDR Symbol Processor download software can be found in Section 6.3.2.

There are no associated logbook entries for the C1 flexible mapping. Those modifications were made at the same time as the C2 messages were fixed. The associated logbook entries for modifications to C2 messaging can be found on page 60 of [5].

4.2.7 Additional C3 Message Sources Implemented

As delivered to DRDC Ottawa, FASSET generated a random C3 message, added the terminal identification (ID) and appended a valid checksum. This message was sent on the downlink to the ground terminal and each subsequent time a new random message was generated. Just before delivery, Harris stated that they wanted a method for sending a specific message to their ground terminal.

Since it was desired to enable C3 messaging options as quickly as possible, it was considered to be too much work to modify the configuration messages. This would have required changes to the LDR Symbol Processor (both kernel and download), the sequencing computer (kernel) and the payload computer. A solution presented itself when it was realized that one could not set up for C3 testing without also including at least one C0 access. This access was not used during the testing yet the parameters were passed from payload computer to LDR Symbol Processor. By using the spare field, and taking over the last three words of the hop field, the C3 information was piggybacked on the C0 access configuration. This limited the C0 access to not using hops in the last three words. This was not a problem because the C0 access was not being used.

Changes were made to the user interface screens in the file __HMIDEF.FST. The screen specifying the AC/AROW was reformatted into uplink and downlink columns. A source field was added for the C3 message (which used the DL_Spare parameter of access #1). The message itself is specified by modifying the last three hop fields in the User Data Access #1 screen.

Changes were also made to the LDR Symbol Processor download software. The three 32-bit fields (using the last words of the hop field for access #1) necessary for the message and terminal ID along with the C3 message source (using .DL_Spare from access #1) had to be passed

down to the routine which generated the downlink access control message in the file GENDLAC.C. These parameters had to be passed in the file SYMDATAP.C, the file PRODLAC.C and the definition file DLEXTERN.H.

Within the file GENDLAC.C three sources were allowed. The first, corresponding to a source value of 0, causes the generation of a random message with a valid ID and checksum. This was the same mode used when FASSET was delivered. A source value of 1 allows the generation of a specified message and terminal ID with a valid checksum appended. In this case, the message bits are taken from the first two words, and the terminal ID is taken from the third. A source value of 2 corresponds to a raw message with all bits specified in the three words. The process for regenerating the LDR Symbol Processor download software can be found in Section 6.3.2.

While making these changes, an unrelated change was made to one of the payload user interface screens. A version date was also added to the Setup Local Emulation screen to allow the user to immediately determine the screen version.

There are no associated logbook entries for the implementation of user-defined C3 message sources. Those modifications were made at the same time as the C2 messages were fixed. The associated logbook entries for modifications to C2 messaging can be found on page 60 of [5].

4.3 Ground Terminal

In this subsection, improvements made to FASSET ground terminal equipment are described. The topics covered include the VME development system and the issue of unreliable removable hard disks.

4.3.1 VME Development System Built

As work proceeded during the early stages of recommissioning the FASSET system, a number of problems developed while trying to use the ground terminal VME chassis for system maintenance and debugging of spare boards. An arrangement was needed which would facilitate working independently on system software and maintenance procedures while leaving the entire FASSET ground terminal available for independent operation.

It was also realized, because any improper shutdown of the ground terminal can corrupt a hard disk, that an efficient, repeatable method was required for copying a complete, bootable ground terminal software disk from a reference copy. To make this idea viable, several more HDDs had to be found to serve as reference, and as additional system disks to allow the preparation of multiple backups for the FASSET ground terminal software.

It became apparent that a separate VME “development system” chassis was required which could boot the basic ground terminal System V Unix OS, and support the attachment of multiple HDDs so that system maintenance procedures such as system disk generation could be developed without requiring the use of the entire FASSET ground terminal. Furthermore, at least some of the HDDs involved had to be mounted in removable drive caddies so that they could be easily interchanged with the VME chassis in the FASSET ground terminal system.

A separate VME chassis was set up with a power supply, MVME147SB-1 single board computer (SBC) card, MVME712 transition module, LCP2 Adapter board, VT220 console terminal, HP LaserJet II printer, QIC-150 SCSI tape drive, CDC Wren-V 500 MB SCSI boot drive, CDC Wren V SCSI reference drive and two trays for removable Fujitsu SCSI M2624FA or M2694FA HDDs. The SBC card and the LCP2 adapter board for the MVME712 transition card set was taken from the unused FASSET ground terminal chassis. Note that this SBC card was not

fully functioning, but would still run the Motorola System V OS without the FASSET application software. Extensive use was made of references [17 – 21] in configuring the development system.

A number of procedures were developed which took advantage of the VME development system to provide backup system disks and test configuration files for continued support of FASSET system operations. These procedures are documented in [3], Annex D3, and [22]. The development system also proved useful by providing an independent means of isolating and testing for faulty or degraded disk drives and MVMESB-1 SBC boards without compromising the operational FASSET ground terminal equipment.

4.3.2 Removable Hard Disks Installed

After using the development system for a few weeks, a problem was identified with the SCSI hard disk drive (HDD) plug-in trays that caused intermittent operation of the ground terminal Unix operating system, and the accompanying FASSET application software. It was eventually determined that the HDD drive tray connector appeared to make unreliable contact with the mating chassis connector when the drive was inserted into either the ground terminal or the development system VME chassis. A HDD tray from StarTech (SNT127S) that has a more secure connector design was used to replace the faulty units. With the new HDD drive tray system installed, no further problems were encountered with drive tray connections.

5. Problems Identified But Not Resolved

The following section contains a collection of hardware and software problems that were encountered during the refurbishment of FASSET, but were never resolved. As in Sections 3 and 4, the subsections may be read independently.

Not all of the unresolved problems encountered during refurbishment are recorded in this section. Some of the minor issues are discussed throughout the document, but the significant unresolved problems that require some explanation are included in this section. In some cases during refurbishment, replacing failed boards with spare modules alleviated faults, and due to time restraints fixing the failed boards were never pursued. The priorities during refurbishment were governed by the requirements of Harris Corporation, and in general the level of redundancy and sparing in FASSET was determined to be adequate considering the situation.

5.1 Line Printer Not Printing

One of the tools provided by the FASSET ground terminal software for tracing automatic fault indicators is an "event log" which records all console messages reported to the operator during emulation execution. As described in [12], Section 4.3.1.4, the operator may select event logging trace messages to be directed to any combination of console window, line printer, and disk file.

It was found that selecting trace messages to the line printer was not working. Investigation into this problem revealed that the Unix print spooler was off-line, and could not be successfully placed online again. This problem appeared to be caused by some conflict between the FASSET application software attempting to access the printer, and the normal Unix print spooler software. The result was that no trace messages or other information could be successfully printed on the ground terminal while the FASSET application software was running.

As a temporary work around, the option to direct trace messages to a disk file was selected in the FASSET application. This file is always given the name "EVENT.LOG" and is created in the directory "/USR/FASSET/" during FASSET application start-up. The "EVENT.LOG" file can be copied to the Unix console port with the "cat /USR/FASSET/EVENT.LOG" command. As described on pages 113-118 of [3], this information can also be copied to a printer if the Unix console terminal function is being supplied by a personal computer (PC) or laptop computer running "HyperTerminal" software.

The inability to print event logs during FASSET emulations remains an outstanding problem, which could be addressed by further examination of the ground terminal FASSET application software, but might require considerable effort to find the relevant code, fix it, and successfully regenerate the correct module of the FASSET code.

More details on this series of problems can be found on pages 4-8 of [5].

5.2 Ground Terminal DDS Controller Causing Downlink Sync Failure

The ground terminal can become effectively paralysed when one of the LOs in the down-converter is shut off. An investigation determined that the LO was being shut off because the hop clock was 2.5 times faster than nominal (referred to as the 2.5 hop strobe fault), and commands were accumulating too fast. Without a hop clock at the proper rate, and thus no LO, it is not possible to achieve downlink synchronization.

Being one of the first steps in setting up communications, downlink synchronization is one of the earliest pieces of feedback provided to the user for gauging synchronization progress. All of the fundamental subsystems must be functioning properly for a FASSET ground terminal to achieve downlink synchronization. Therefore, it is not surprising that in troubleshooting downlink synchronization, there were often multiple problems overlaid and concealing each other.

The cause of the 2.5 hop strobe fault is not currently known due to time limitations and prioritization. A spare board (S/N C30002) appeared to eliminate the problem, and with a working DDS Controller available, repairs to the original board (S/N C30004) became a lower priority and the source of the fault was never determined.

Some symptoms from the initial failure, however, have been recorded in the logbook on pages 53-57 of [5]. An attempt was made to extend the DDS Controller board so that it could be probed during emulation, however the problem was intermittent, making testing difficult. A plastic leaded chip carrier (PLCC) extender was setup for probing U26 on the DDS Controller board to monitor the LDR-RF-DLCMD serial interface. The results were inconclusive, as the board would not power up properly with the PLCC extension.

The 2.5 hop strobe failure was seen in a later incident during troubleshooting at Harris' facilities. The fault occurred while running with the only proven working board, but it is possible that the ground terminal was in an abnormal failure state. The 2.5 hop strobe fault was not seen after that isolated incident. No spare DDS Controller exists, so if the current DDS Controller fails, then debugging will be required. One of the spare DDS Controller boards in storage is unpopulated and untested, but must be considered as a possible replacement in the case of another failure.

5.3 Remaining Issues of Excessive Uplink Attenuation

As described in Section 3.1.3, a number of steps were taken to substitute manual attenuator controls for the three programmable attenuators located in the ground terminal TRANSEM, the payload Receiver, and the payload Downconverter Subsystems. In all cases, it was not possible to determine with certainty if the programmable attenuators had failed completely, gone out of calibration, or were not properly being commanded by software control.

These attenuator problems can be investigated further if time and resources become available. In each case, the attenuator would have to be isolated, and tested with a calibrated signal source and spectrum analyzer. The digital control unit for each attenuator would also have to be evaluated with a logic analyzer to determine if the correct command sequences were being produced. Any necessary software corrections would then have to be applied to both the payload and ground terminal. Finally, a series of uplink calibration tests would need to be performed using the attenuation controls provided in both the ground terminal and payload setup screens (see [11] & [12]).

More details on this series of problems can be found on pages 8-17 of [5].

5.4 Excessive Bit Errors in Uncoded Multi-user Accesses

As the FASSET system was being subjected to final testing at DRDC, before being shipped to Harris Corporation, a number of last minute anomalies were found in the multi-user emulation mode. The particular test configuration was MUDATA03, as described in Annex C.4. As long as tests used the standard (default) settings for configuration MUDATA03, 2400 b/s, rate ½ coding/decoding, the emulations worked successfully and error free data was passed (after full FASSET synchronization was achieved). If the standard test was modified to run in uncoded

mode, at 4800 b/s, the traffic became completely unusable, with no pattern synchronization. This same problem was observed on the TT&C user running in uncoded mode and was evident during end-to-end traffic with the payload TT&C port and during payload TT&C loopback to the TT&C user.

Lack of time prevented a further investigation of this apparent "multi-user uncoded" traffic problem. If time and resources become available, this problem should be further investigated. It is suspected that a problem may still exist in the payload processing of multi-user data modes. The problem might be either a limitation of payload capacity to handle multiple 4800 b/s uncoded data streams, or it might be due to a conflict in correctly switching individual user data streams between coded and uncoded processing.

More details on this series of problems can be found in [5] on page 67.

5.5 Spares

The level of sparing in FASSET is a critical issue because of the age and complexity of the hardware. When a serious fault occurs, testing can be halted for days or weeks depending on the severity of the failure. Sparing, in the short-term can minimize the time and effort required for troubleshooting problems. In general, sparing is a good risk reduction strategy for FASSET because both the payload and ground terminal were designed with a modular approach.

When FASSET was delivered to DRDC Ottawa, spares were included for the critical items (see Section 2.3.2), while products available for purchase as commercial-off-the-shelf (COTS) were not spared. Some COTS items are no longer readily available, and the spares for critical items have been depleted in some areas. During refurbishment, the main focus was to re-establish a working system, and sparing was a minor priority. Those modules that were exchanged with a line replaceable unit (LRU) were not repaired. A preliminary diagnosis of the symptoms was conducted in some cases, however all failed units remain in disrepair. If the resources become available, the failed modules should be prioritized according to the circumstances and repaired.

The following modules or boards have already been replaced with spares and thus have reduced or no spares currently:

a. Ground terminal MVME147SB-1 single board computer
b. Ground terminal DDS Controller
c. Ground terminal Frequency Reference Generator
d. Payload Timing Control Subsystem Downlink Delay Module
e. Payload Timing Control Subsystem Downlink Expander
f. Payload Timing Control Subsystem PCI Module
g. Several AUI-to-thinwire transceivers
h. Several ground terminal hard drives (Fujitsu M2624FA and M2694FA)
i. RS-232 switch module

5.6 Incorrect Wiring in Payload Controller Interface

After months of testing FASSET at Harris' facilities, the Payload Controller Interface (PCI) module in the Timing Control Subsystem suddenly failed at power-up (see Section 3.2.5 for details). For some unexplained reason, the fault (eventually isolated to the PCI) did not manifest itself until after FASSET was delivered to Harris' laboratory in Florida. It was felt that either the shipping or the Florida environment in some way affected the circuits within the PCI, and caused it to behave differently from when it was operated at DRDC Ottawa.

To temporarily alleviate the PCI EPROM circuit problem, a quick and simple repair was done (see Figure 6) that did not actually correct the address line wiring errors. This could represent a serious problem if the PCI firmware had to be modified, and expanded to a point where it occupied a larger portion of the EPROM address space. The proper and permanent modifications required for the PCI circuit board are highlighted in Figure 7, Section 3.2.5. If the resources become available, the PCI EPROM circuit should be modified and repaired accordingly.

6. Maintenance Procedures

This section is a collection of maintenance procedures for FASSET that were developed and refined, out of necessity, during the refurbishment. The procedures may be read independently, and as such some background has been repeated to aid the reader.

Many of the procedures developed during refurbishment have already been documented in the logbooks [4, 5, 6] and in the FASSET training report [3]. The topics covered in this section include changing passwords, modifying the payload user interface, modifying the LDR Symbol Processor software, and ground terminal tape backup and restoration.

6.1 Changing Passwords

Passwords exist on both the ground terminal and payload to limit access. In practice, the physical security (and intimidating complexity) of FASSET is sufficient to protect it. So the passwords were chosen to be simple and easy to remember.

The security department at Harris Corporation desired the ability to change the passwords to more secure ones. This was deemed necessary for both the payload and ground terminal. Procedures for changing the passwords are given in the following sections.

6.1.1 Payload Password

The operating system for the payload is DOS and therefore does not have any integrated password protection. The FASSET designers instead, used a small utility for the password. After delivery, this utility was deemed to be unnecessary and removed. The only password protection is in the FASSET executable user interface screens. This executable compares the password entered by the user with one stored in the following file.

```
C:\FASSET\CONFIGSW\PLDCONF\__PASSWD.FST
```

This file can be easily edited by using any text editor. The password for the payload, as delivered to Harris, is:

```
fasset
```

6.1.2 Ground Terminal Passwords

The operating system for the ground terminal is Unix and thus uses standard Unix passwords. For those unfamiliar with Unix, this section presents the method for changing passwords for both the root account and the fasset account.

To change either password, it is easiest to log on as root on the console (most likely through a HyperTerminal window on the laptop). This allows any operating system restrictions on choice of password to be bypassed. It also ensures that the normal fasset screens do not start up (as would happen if one logged on as fasset). The following lines allow either the fasset user account or the root password respectively to be changed.

```
Passwd fasset
passwd root
```

Unix then requests the old password, and then twice asks for the new password. The passwords for the ground terminal fasset and root accounts, as delivered to Harris, are respectively:

```
fasset
root
```

6.2 Modification of Payload Computer Application

The payload computer software has three main components: the payload screens, the screen library and the overall controller. The screen library contains the generic screen engine and the action handlers. The payload screens details are defined in a text file read in at run-time. The following sections detail how to regenerate the executable and the library as well as how to change the payload screen details.

6.2.1 Regeneration of Payload Computer Executable

The source files for the payload executable can be found on the red disk (see Section B.3). A secure environment is necessary for using the red disk. After modifications are made to the source files, the executable is then rebuilt and copied (via floppy disk) onto the payload computer. The steps are:

a. Boot the red disk in a secure environment

b. Make any modifications to the source files

c. Ensure the screen library HMILIB.LIB is up to date (Section 6.2.2)

d. Start Windows 3.1 by typing WIN and then open a DOS command window. This step is necessary to get the proper memory management for NMAKE to be used in the next step

e. Rebuild the executable by typing the following:
```
cd \FASSET\CONFIGSW\PLDCONF
GENPLDC.BAT
```

f. It is necessary to wait a long time because there are a lot of modules to compile and the computer is relatively slow

g. Put the executable on floppy disk by typing the following:
```
copy PLD_CONF.EXE A:
```

h. Move to the payload computer

i. Save the old version of the executable by typing the following:
```
cd \FASSET\CONFIGSW\PLDCONF
rename PLD_CONF.EXE PLD_CONF.EX4
```

j. Install the new version of the executable by typing the following:
```
copy A:\PLD_CONF.EXE
```

k. Reboot the payload computer to use the new software

6.2.2 Regeneration of Screen Library

Regeneration of the screen library is only necessary if the maximum definitions change (such as limiting the number of screens) or if the action handler changes (such as adding a new action). There is no need to regenerate if the screen details are the only things changed because __HMIDEF.FST is read in at run-time and is not part of the library.

The maximums are stored in HMIDEFS.H, which ends up being incorporated in the library, and later the executable. It is important to note that the executable program is very close to the DOS maximum program size. If the maximums are increased significantly, then it is likely that the executable will be unable to run.

Once the screen library is regenerated, it will be necessary to regenerate the payload executable (see Section 6.2.1). The following step causes the screen library to be regenerated:

a. Boot the red disk in a secure environment
b. Make any modifications to the source files
c. Start Windows 3.1 by typing WIN and then open a DOS command window. This step is necessary to get the proper memory management for NMAKE to be used in the next step
d. Rebuild the library by typing the following:
```
cd \FASSET\CONFIGSW\PLDCONF
GENLIB.BAT
```
e. Wait a long time, there is a lot to compile and the computer is relatively slow
f. It is now necessary to regenerate the payload executable to incorporate the updated screen library

6.2.3 Modification of Payload Screens

The payload user interface screen details are stored in a text file __HMIDEF.FST that is read in at run-time. Two portions of the user interface screens are not in this file. The first is the action handler that performs all the actions commanded by the screens. The action handler is part of the screen library. The second portion of the user interface screens is the limit on number of screens and fields. These limits can be found in HMIDEFS.H, which gets compiled into the screen library.

Changes that don't necessitate modifying the maximums, or adding a different action, can be made with a text editor on the payload computer (with no need to recompile). There is a software tool CountFields detailed in Annex D that can determine the required maximums after __HMIDEF.FST has been modified. This should be compared with the table below to see if the maximums must be changed. The table shows the parameter name, the original value, the required value as determined by CountFields, and the values currently in HMIDEFS.H.

Screen Parameter from HMIDEFS.H	Original	Required	Current
H_MAX_NUM_SCREENS	55	57	60
H_MAX_NUM_ DISPLAY_FIELD	550	525	550
H_MAX_NUM_ PARAM_FIELD	370	261	290
H_MAX_NUM_ ACTION_FIELD	160	150	170
H_NUM_DISPLAY_FIELDS_PER_SCREEN	25	25	26
H_NUM_PARAM_FIELDS_PER_SCREEN	20	20	21
H_NUM_ACTION_FIELDS_PER_SCREEN	12	11	13

Table 5 Field counts in payload screens

Assuming the maximums are not exceeded, adding new screens or fields can be done by following the format of existing screens. If a new action is required, then an associated token must be defined in HMIDEFS.H and the action must be included in the action handler ACTION_H.C.

6.3 Modification of LDR Symbol Processor Software

The LDR Symbol Processor is a single board computer in the payload LDR Baseband Subsystem. The symbol processing code is responsible for dealing with data that is received from the payload demodulator and then sent to the payload modulator for transmission to the ground. This is the software that was fixed and modified to allow multiple C0, flexible mapping for C0/C1/TT&C and special generation for C3 messages.

The LDR Symbol Processor has two components: the kernel and the download portion. The kernel, which deals with the low-level functionality including message handling, resides in firmware on an EPROM. The download software, which includes the high level functionality, is stored on the payload Configuration Computer hard disk and is downloaded at power-up.

6.3.1 Regeneration of LDR Symbol Processor Kernel Firmware

Regeneration of the kernel was required to allow for proper handling of configuration messages that included the downlink assignments. Previously, this information was obtained from a mapping table based on the uplink assignment.

The kernel source files can be found in the C:\FASSET\A2A11\SYM\CODE\KSRC directory. Note that header files used by both the kernel and download software can be found in the \FASSET\A2A11\SYM\CODE\INC directory.

The regeneration of the kernel requires a computer equipped with a SuperPro EPROM programmer or equivalent. The following steps are necessary to generate a new EPROM for the kernel:

a. Boot the red disk in a secure environment

b. Move to the build directory
```
cd \FASSET\A2A11\SYM\CODE\BUILD
```

c. Rebuild the kernel
```
SYM_ROM.BAT
```

d. Insert a blank floppy when asked, SYMROM.ABS and .MAP are automatically written to it (the current size is roughly 110 kbytes)

e. Switch to the EPROM programming computer

f. Move to the ROMS directory
```
cd \fasset\roms
```

g. Copy the files to the programming computer
```
copy a:\SYMROM.*
```

h. Set up ABStoBIN.txt as follows:

```
; abstobin.txt
;
; configuration file for abstobin
;

c:\fasset\roms\symrom.abs     ; Intel hex format source file
c:\fasset\roms\symrom.map     ; Link map used to obtain checksum address for ROM
c:\fasset\roms\symrom.bin     ; Binary output file
0                             ; Fill value for unused locations
0xF0C00000                    ; Base address for ROM=0xF0C00000 and for DL=0x0
ROM                           ; Download or ROM
```

i. Create binary output file SYMROM.BIN (the program automatically converts format, fills holes with 0's, and inserts checksum – see Annex D.1)

```
ABStoBIN
```

j. Make EPROM label SYMROM with:
   current time and date
   4 byte validity checksum (last one was 001F2AAB)
   2 bytes programmer checksum (last one was 2B9F)

k. Switch on SuperPro EPROM programmer and start software

```
SP2P
```

l. Select AMD 27C020

m. Fill entire buffer with 0's

n. Read in SYMROM.BIN (ensure checksums match)

o. Insert erased EPROM into socket on programmer

p. Perform blank check

q. Program EPROM and put on label when compete

r. Install EPROM on LDR Symbol Processor board

6.3.2 Regeneration of LDR Symbol Processor Download Software

The LDR Symbol Processor download software does most of the work receiving uplink communications and preparing the downlink information to be sent to the modulator. When delivered, FASSET used a fixed mapping based on uplink access to determine when data goes in the downlink. This was modified to allow configuration of the downlink access independent of the uplink.

The download source files can be found in the C:\FASSET\A2A11\SYM\CODE\DSRC directory. Note that header files user by both the kernel and download software can be found in the \FASSET\A2A11\SYM\CODE\INC directory.

The regeneration of the download software requires a computer to allow conversion of the compiled output into the format necessary for download. The following steps are necessary to regenerate new download software:

a. Boot the red disk in a secure environment

b. Move to the build directory

```
cd \FASSET\A2A11\SYM\CODE\BUILD
```

c. Rebuild the download software

```
SYM_DL.BAT
```

d. Insert a blank floppy when asked, SYMDATAP.ABS and .MAP are automatically written to it (the current size is roughly 85 kbytes)

e. Switch to the EPROM programming computer

f. Move to the ROMS directory

```
cd \fasset\roms
```

g. Copy the files to the programming computer

```
copy a:\SYMDATAP.*
```

h. Set up ABStoBIN.txt as follows

```
; abstobin.txt
;
; configuration file for abstobin
;

c:\fasset\roms\symdatap.abs   ; Intel hex format source file
c:\fasset\roms\symdatap.map   ; Link map used to obtain checksum address for ROM
c:\fasset\roms\symdatap.bin   ; Binary output file
0                             ; Fill value for unused locations
0x0                           ; Base address for ROM=0xF0C00000 and for DL=0x0
Download                      ; Download or ROM
```

i. Create binary output file SYMDATAP.BIN (the program automatically converts format and fills holes with 0's – see Annex D.1)
```
ABStoBIN
```

j. Copy source, map and output files to floppy
```
copy SYMDATAP.* a:
```

k. Switch to payload Configuration Computer

l. Move to the CODE directory
```
cd \FASSET\A2A11\SYM\CODE
```

m. Save old versions on payload by renaming them

n. Copy new version from floppy
```
copy a:\SYMDATAP.*
```

o. Move SYMDATAP.BIN to software download area
```
copy SYMDATAP.BIN \FASSET\CONFIGSW\PLDCONF\__SYM0__.FST
copy SYMDATAP.BIN \FASSET\CONFIGSW\PLDCONF\__SYM1__.FST
copy SYMDATAP.BIN \FASSET\CONFIGSW\PLDCONF\__SYM2__.FST
```

6.4 Ground Terminal Tape Backup and Restore

A short time after FASSET was delivered to DRDC Ottawa, backup tapes were made of the ground terminal's hard disk. In early refurbishment for Harris, it was determined that the hard disk on the ground terminal had been corrupted. First, the technique used to backup the disk will be presented. Then the restore technique will be explained. This same technique is necessary if the "gold" disk (or equivalent reference disk) is not available.

There is additional information in Annex D.3 (FASSET Hard Disks) of the FASSET Training Report [3], such as the slice table that is necessary during the format operation.

6.4.1 Installation of Tape Drive

Before backing-up or restoring, it is necessary to connect the SCSI tape drive. The steps are as follows:

a. Ensure the ground terminal is powered down

b. Ensure the SCSI tape drive address to 4

c. Properly connect the SCSI tape drive with any necessary terminator

d. Turn on the tape drive

e. Turn on the ground terminal

f. Halt the debugger of the System Controller by pressing break at the console or the reset button on the front panel

g. Update the SCSI configuration by using the following command at the console:
IOT;T
(Respond to prompts with Y and Y)

6.4.2 Backup

The ground terminal disk backup is split into two parts: backup of /root and backup of /usr. The root can fit into one tape, but three tapes were required for the /usr partition.

6.4.2.1 Backup Root

The root files include all of the operating systems files. Unix keeps mount points of other devices as directories below the root and it is not desirable to backup these file systems. The -mount switch ensures that other mounted file systems are not backed up. The root is backed up with the following steps:

a. Insert a blank tape and close the latch
b. Backup the root files by using the following commands:

```
cd /
find . –mount –depth –print | cpio –oBvc > /dev/r40t
```

c. Remove the tape and label appropriately including the time and date

6.4.2.2 Backup Usr

The usr files include all of the FASSET files. This backup likely requires many tapes and the operator will be prompted when it is time to change tapes. The usr partition is backed-up with the following steps:

a. Insert a blank tape and close the latch
b. Backup the usr files by using the following commands:

```
cd /usr
find . –depth –print | cpio –oBvc > /dev/r40t
```

c. Remove the tape and label appropriately including the time and date

6.4.3 Restore

The backup tapes include all of the files for the operating system and FASSET executables. If the disk is blank, it is necessary to install the base operating system before one can restore the backup tapes.

6.4.3.1 Install Base Operating System

The base operating system can be found on the Motorola System V/68 Base Operating System (BOS) tape, which can be found in a box along with the installation manual. These steps have been adapted from the information in the manual.

a. Insert the BOS tape and close the latch
b. Boot the tape by using the following command at the console:

```
BO 4,0
```

c. Allow the tape to install to the hard disk
d. Boot the hard disk by using the following command at the console:

```
BO
```

e. Run setup by using the following command at the console:

```
setup
```

(Respond to prompts with N N Q Y and lgt2)

6.4.3.2 Restore Root

Once an operating system is available, the root can be restored from the backup tape. The following steps are necessary for restoring the root:

a. Insert the root backup tape and close the latch

b. Restore the root files with the following commands:

```
cd /
cpio -iBvud < /dev/r40t
```

6.4.3.3 Restore Usr

With the base operating system installed and the root restored over it, the ground terminal is now ready for restoration of the usr files. The following steps are necessary for restoring the usr:

a. Insert usr backup tape #1 and close the latch

b. Restore the root files with the following commands:

```
cd /usr
cpio -iBvud < /dev/r40t
```

(When prompted, insert next tape and enter /dev/r40t)

c. Do not reboot at this point. See the next section for how to disable the FASSET start-up to allow for quick reboots.

6.4.3.4 Optional Post-Restore Steps

A normal boot sequence includes starting all the Unix services and the FASSET executables. Some of the post-restore steps require rebooting and it is onerous to wait the 10 minutes necessary for FASSET to run each time. The FASSET start-up is initiated by the file /rc2.d/S99FASSET. This file is executed whenever Unix mode 2 (normal multi-user mode) is started. This file can be renamed to R99FASSET, which will ensure it is not executed. Later, when normal FASSET functionality is desired, the R99FASSET should be renamed back to S99FASSET.

It may be desirable to remove the tape drive. This can be accomplished by powering down the system, disconnecting the tape drive, and moving the terminator appropriately. To update the SCSI configuration, the following command should be entered at the console debugger:

```
IOT;T
```

(Respond to prompts with Y and Y)

The command necessary to shut down the ground terminal prior to power off has many switches and is hard to remember. It is easier to assign the alias “die” which can be accomplished by adding this line to /.profile:

```
alias die=cd/;shutdown -go -I0 -y
```

Since the disk drives for the ground terminal are removable, it is desirable to know which disk is running when logging-in as the root user. To allow the disk to be easily identified, an echo is added to ./profile including the disk type and number:

```
echo Fujitsu M2624 (D3)
```

The default delete key for this Unix installation is the "#" key which is not intuitive. This can be changed to the delete key. It is necessary to examine the .profile for the user to find the CERASE entry. This should then be changed to the character "^?" (delete).

It may also be desirable to add a printer. This is done in the normal Unix method and is not covered in this report. Note that problems with the normal printer spooler occurred when logging to printer is enabled in the FASSET screens (see Section 5.1 for details). For that reason, it is recommended that logging to printer not be used.

7. Lessons Learned

This section is a compilation of lessons learned during the refurbishment of FASSET. The various subsections are independent, and may be read accordingly.

The purpose of this section is to make recommendations that will help avoid serious obstacles that are similar to those encountered during refurbishment. The topics covered in this section include reliability issues caused by shipping, transfer of classified documents to industry, long-term storage of equipment, and the long-term availability of COTS products.

7.1 Reliability Issues Caused by Shipping

The shipping of FASSET from DRDC Ottawa to Harris Corporation in Melbourne was successful, but during the process a number of lessons were learned. Some of the lessons were learned from negative experiences, while others were learned from examples of good preparation and execution.

Delivery of FASSET to Harris involved shipping of the hardware by truck over 3000 km of road. With such a significant amount of mechanical stress incurred by the hardware, it was not surprising that problems developed despite the precautions taken.

Shipping crates for the system were available from when FASSET was taken to Lincoln Labs for interoperability testing (see Section 2.1.5). All of the shipping crates provided an adequate amount of protection except the one designed for the Timing Control Subsystem (TCS). That crate, being designed differently from the others, does not have a shock-absorbing layer of foam. Instead, the crate includes a strong wooden platform to which the TCS is attached with tight straps. A solid wooden cover fits on top of the TCS and screws into the platform forming a complete shipping crate. Since there is no absorbing layer of foam between the TCS and the base of the crate, vibrations are transferred more easily from the truck to the TCS hardware. For future shipping, the TCS will require a new crate, or a modified one that has a relatively thick layer of foam.

The ground terminals were wrapped in multiple layers of bubble wrap and strapped to a wooden skid. This is not ideal, but it is better than removing all of the VME cards and packing them individually in a crate. A better solution is to build large crates with a thick layer of formed foam to absorb shock. A less expensive solution is to ship the ground terminals on wooden skids, followed by a rigorous inspection and checkout process. The former solution is more ideal, more expensive, and more proactive than the latter.

The US customs broker hired by Harris to handle the shipping of FASSET from DRDC to Melbourne performed well. They were well prepared prior to the shipment, and processed the customs work efficiently. Future shipping of FASSET should involve an experienced customs broker that is well known to the receiving party.

The method of labelling some crates as those to be opened first was very beneficial. Even with well-catalogued shipping crates, it can be difficult deciding which equipment to unpack and setup first. With a simple dual-level priority system it was clear which crates contained the most important components for initial setup.

7.2 Transfer of Classified Documents to Industry

The security classification of hardware, software, and documentation was an issue when the topic of shipping FASSET across the border was raised. Many of the classified documents contain critical information that is required for debugging FASSET. Normal operations do not require any of the classified or unclassified documents, but when detailed understanding is required, the classified documents are invaluable.

As expected, some hardware faults were encountered during setup, and the classified documents would have been helpful. However, they were also required at DRDC until just prior to packing and shipping, so it was not possible to ship them any earlier. Since the classified documents were shipped by a different and more time consuming route than the system hardware, the best effort was made to work without them. Within a few weeks, the classified documents arrived and were used immediately.

Since Harris Corporation is a private company and not a Canadian or US government agency, the shipping of classified documents was somewhat complicated. Although more work is involved, it is recommended that in the future, a test shipment of less frequently used classified documents is sent as early as possible to setup and understand the shipping process.

7.3 Issues Associated with Long-term Storage

In the late 1990's, the FASSET system was put into a secure storage building at DRDC Ottawa. Since laboratory space must be prioritized, the least important equipment is typically shelved and eventually moved into nearby storage facilities. When the need arises, as in the refurbishment for Harris Corporation, the equipment can be brought back to active laboratory areas for work.

The storage building where FASSET was stored for five years is partially climate controlled. Extreme cold is not a problem as the building is heated, but there is no cooling. During extreme summer conditions the effects of heat, humidity, and oxidization contributed to the aging of the FASSET hardware. Some of the equipment, including the payload subsystems and spares, was wrapped and sealed in antistatic bags and bubble wrap and stored in crates. The ground terminals were stored with the cards mounted in the chassis, although the spares were wrapped in plastic and stored in crates.

With so many components, and so much time passed since delivery, it is difficult to properly catalogue the FASSET system. It is recommended that prior to being stored next, FASSET be catalogued with a best guess description of all components and cables. A brief status report for each component will provide future scientists with some background and a summary that will help in making decisions.

7.4 Long-term Availability of COTS Not Guaranteed

Some portions of FASSET were delivered without spares because they were considered to be commercial-off-the-shelf (COTS). While this may have been true at the time FASSET was being developed, it certainly was not true 5-10 years later. An example of such a problem was the payload hard disk. This disk was considered to be of adequate capacity at the time of assembly, but 10 years later, disks of that capacity are now unavailable through normal purchase.

The work-around, for disks and some single board computers, was to purchase used parts through auction services. With this purchasing method, there is a risk that the parts are not quite what is needed or that they are broken. To mitigate this risk, it is necessary to purchase more than the number required in hopes that enough working ones can be found either through luck or cannibalization.

As prototype development moves more towards COTS to save money, it must be remembered that something off-the-shelf now may be unavailable in 10 years. In future, sparing of COTS items should be done at the time of assembly to ensure they are available over the lifetime of the equipment.

8. Conclusions and Future Work

8.1 Summary

The FASSET system is an EHF LDR satellite communications payload and ground terminal test bed that was developed in the early 1990's for the Department of National Defence. In 1996, FASSET was delivered then debugged and modified for one year at DRDC Ottawa prior to successful interoperability testing at Lincoln Laboratory in 1997.

FASSET went into storage for several years until July of 2002, when Harris Corporation requested the use of FASSET to assist in the development of their EHF modem. A contract between Harris and DRDC Ottawa resulted in the refurbishment and delivery of FASSET to Harris in Melbourne, Florida. This technical memorandum has described the repairs and improvements that were conducted for Harris in support of the contract.

The primary goal of the refurbishment was to reconstitute FASSET to the state of functionality that it possessed prior to being stored for many years at DRDC Ottawa. With refurbishment completed by March 2003, the system was delivered to Harris, and after some follow-up troubleshooting FASSET became a functioning tool that was available for exploitation. Briefings and hands-on training were provided to Harris personnel covering FASSET's background, functionality, and operation.

In this report, significant refurbishment events are described in detail including items repaired, modifications and improvements, and unresolved problems. A list of test equipment used during refurbishment is also provided, along with descriptions of various maintenance procedures that were developed. A series of lessons learned and potential areas for future work are also outlined.

The section in the report covering items repaired is the largest, as it was one of the main areas of refurbishment work. Significant effort was invested in fixing critical faults in the Timing Control Subsystem including the Payload Controller Interface, various power supplies, and the Uplink and Downlink Expanders. Other items repaired include the X-terminal, the bit error rate test set, the network transceivers, the Transmitter reset circuit, and the Frequency Reference Generator.

The other major section in the report covers modifications and improvements that were implemented in FASSET during refurbishment. Although some minor work was done in the ground terminal, most of the improvements focused on payload software in the LDR Baseband Subsystem. Uplink to downlink mapping was enhanced for TT&C and C1 messaging, support for multiple user data accesses was added, and additional C3 message sources were implemented. Other modifications and improvements include the payload's external loopback, the emulation configuration files, the TCS monitor port for power supplies, the VME development system, and the removable hard disks.

8.2 Conclusions

With a significant investment of time and effort, the refurbishment of FASSET was completed and its original functionality restored. New system capability was added at the request of Harris Corporation to aid in integration and testing of their modem. All of Harris' requirements were met, and the contract moved into the leasing phase of testing and exploitation. Harris personnel were provided with briefings and hands-on training that covered FASSET's background, functionality, and operation. Advanced training on FASSET's capabilities was

offered, while training for long-term maintenance and troubleshooting techniques were reserved for future sessions.

Restoration of functionality was the main priority during refurbishment, while long-term reliability and subsystem redundancy were secondary concerns. The modifications and improvements done during refurbishment were primarily at the request of Harris, and depending on the requirements of future clients, other improvements may be necessary for projects involving FASSET.

8.3 Future Work

DRDC Ottawa is currently providing on-site and remote technical support to Harris Corporation during the lease of FASSET. As this is an ongoing contract, DRDC Ottawa will continue to provide support, however without access to the equipment, that support becomes more difficult as time passes. In addition to troubleshooting that may arise, there are many areas of FASSET that can be improved if there is a requirement to increase reliability, subsystem redundancy, or functionality.

Possible future work may include another session of the previous training to instruct new personnel, and to provide more detailed training for others. Other future work might include the development of training necessary for someone to maintain and troubleshoot FASSET over the long-term. This requirement may become a greater priority if the client becomes more dependent on the system, and makes longer-term plans.

During refurbishment, several areas were identified as priorities for improvement and enhancement should the resources become available. In addition to the work identified in Sections 2.3.3 and 5.5, other areas include the changing of all aging power supplies (see Section 3.2.4), modification of the reset circuit within all DIU modules (see Section 3.2.2), replacement of SMA cabling in the Frequency Synthesizer (see Section 3.2.11), and the correction of the address line wiring in the PCI module (see Section 3.2.5). Furthermore, it is recommended that the rear connectors on the TCS be replaced with non-rotating bulkhead connectors as some have become worn and damaged over time (3.2.11).

Another area of future work might include a detailed assessment and maintenance inspection of the FASSET hardware. In addition to helping maintain current working knowledge of FASSET, this task could also provide a comprehensive snapshot of working system metrics that would be available for troubleshooting and technical support.

9. References

1. "Military Standard Extremely High Frequency (EHF) Low Data Rate (LDR) Satellite Data Link Standard (SDLS): Uplinks and Downlinks", MIL-STD-1582B, Department of Defense, USA, 18 Apr 1990.
2. "Military Standard Extremely High Frequency (EHF) Low Data Rate (LDR) Satellite Data Link Standard (SDLS): Uplinks and Downlinks", MIL-STD-1582C, Department of Defense, USA, 10 Dec 1991.
3. Liao, B., Addison, R., "A Compilation of Training Procedures for the FASSET EHF SATCOM Simulator", DRDC Ottawa Technical Memorandum #TM 2003-239, December 2003.
4. "FASSET Log Book Volume 3," DRDC Ottawa Log Book, 24 July – 13 December 2002.
5. "FASSET Log Book Volume 4," DRDC Ottawa Log Book, 16 December 2002 – 26 February 2003.
6. "FASSET Log Book Volume 5," Harris Log Book, 17 March – 18 September 2003.
7. Liias, G. G., "Simulation of MIL-STD-1582C TRANSEC Functions", MIT Lincoln Lab Report SC-96, October 1993.
8. "FASSET Classified Technical Notes", classified communications, document #555, DRDC Ottawa, March 2003.
9. Schefter, M., J., "Low Data Rate Data Link Specification", DRDC Ottawa Contractor Report – MPR Teltech Ltd., 17 January 1997.
10. Schefter, M., J., "System Technical Manual for FASSET Phase II", DRDC Ottawa Contractor Report – MPR Teltech Ltd., 14 February 1997.
11. Malarky, A., "Payload Technical Manual for FASSET Phase II," DRDC Ottawa Contractor Report – MPR Teltech Ltd. & COM DEV Ltd., 9 August 1996.
12. McKerracher, R., "Ground Terminal Prime Item Technical Manual," DRDC Ottawa Contractor Report – MPR Teltech Ltd., 20 August 1996.
13. "Engineering Change Notice / Waiver / Deviation Summary Report: Volume 1," DRDC Ottawa Contractor Report – MPR Teltech Ltd., 26 August 1996.
14. "Engineering Change Notice / Waiver / Deviation Summary Report: Volume 2," DRDC Ottawa Contractor Report – MPR Teltech Ltd., 26 August 1996.
15. "Design Description Document for the Payload Clock Generator – Part 2 of 3", DRDC Ottawa Contractor Report – MPR Teltech Ltd. & COM DEV Ltd., 18 June 1996.
16. "Design Description Document for the Payload Timing Control Subsystem", DRDC Ottawa Contractor Report – MPR Teltech Ltd. & COM DEV Ltd., 25 June 1996.
17. "MVME147S MPU VME Module User's Manual", Motorola Inc. Computer Group, document No. MVME147S/D3, March 1992.
18. "MVME147BUG 147Bug Debugging Package User's Manual, Motorola Inc. Computer Group, document No. MVME147BUG/D4, June 1992.
19. "Installation Instructions for SCSI Drives", Motorola Inc. Computer Group, document No. MVME8XXK/K5, April 1993.

20. “MVME712-12, MVME712-13, MVME712A, MVME712AM, and MVME712B Transition Modules and LCP2 Adapter Board User’s Manual”, Motorola Inc. Computer Group, document No. MVME712A/D2, January 1993.

21. “System V/68 Release 3 Version 7.1 Base Operating System Object Release and Source Code Provision Software Release Guide, Software Release FE03.71”, Motorola Inc. Computer Group, document No. QIC-8230TBV6871A and QIC_8230TSV6871A, 1992.

22. Lambert, J.D., “System Generation and File Transfer Procedures for the FASSET EHF Simulator”, DRDC Ottawa Technical Note #TN 2004-006, December 2003.

Annex A VME Bus Trace Information

The FASSET ground terminal and several payload subsystems use the VME bus for inter-board communications. During construction of the ground terminal, the original software developers left some debugging software in the executable program that allows tracing of the ground terminal processing. When the DRDC team was modifying and fixing the software for the payload LDR Symbol Processor, debugging information was added. There are also other transactions (non-debugging) that can be monitored in both the ground terminal and payload that are of interest.

In this annex, information relevant for tracing the payload and the ground terminal is detailed.

A.1 Payload

The payload consists of many subsystems that include processing elements. The only one with debugging information is the LDR Baseband Subsystem. Within this subsystem is a VME backplane hosting several boards including the LDR Symbol Processor (a single board computer) and the Data Routing Switch (a custom board that hosts the Viterbi decoder as a secondary function).

A.1.1 Debugging Information

The LDR Symbol Processor (single board computer) runs the symbol processing code that is responsible for dealing with data received from the payload demodulator and then sending it to the payload modulator for transmission to the ground. This was the software that was fixed and modified to allow selectable multiple C0, flexible mapping for C0/C1 and special generation for C3 messages.

During these modifications, debugging information was added by writing values to unused addresses. The following sections detail the meaning of the various addresses and values used. Values are one of Boolean, integer or unsigned integer.

Note: The single board computer used for the LDR Symbol Processor uses the opposite endian format to normal VME bus transactions. As a consequence, all the data values written are byte-reversed when read by the VMETRO bus analyzer.

A.1.1.1 LDR Data Formatter Routine

The LDR data formatter routine takes downlink messages and inserts them in the proper place prior to permutation, and leading to transmission.

The source file is \FASSET\A2A11\SYM\CODE\DSRC\LRDATAFO.C.

Address	Value	Meaning
10000500	12345678 11111111 22222222 33333333	Start of processing TT&C access C0 access C1 access
10000510		DL Modulation
10000520		DL Diversity
10000530		UL Diversity

Address	Value	Meaning
10000540		Number of DL Hops
10000550		DL Data Rate
10000560		DL Hop Index
10000570		DL Symbol Index
10000580		DL Hop Number
10000590		DL Data (1 to many)

Table 6 LDR data formatter addresses

A.1.1.2 Configure Symbol Processor Routine

The configure symbol processor routine receives access configuration messages (for C0, TT&C, C2 and C3) from the Configuration Computer.

The source file is \FASSET\A2A11\SYM\CODE\KSRC\CONFIGSY.C.

Address	Value	Meaning
10000600	01010101	Load Symbol Configuration
	02020202	Load C0/C1 Access
	03030303	Load AC Access
	81818181	Transmit Symbol Configuration
	82828282	Transmit C0/C1 Access
	83838383	Transmit AC Access
10000610	00001111	Normal C0 in Transmit C0/C1 Access
	00002222	TT&C in Transmit C0/C1 Access
	11111111	Normal C0 in Load C0/C1 Access
	22222222	TT&C in Load C0/C1 Access

Table 7 Configure symbol processor addresses

A.1.1.3 Decode C2 Message Routine

This routine decodes C2 messages using the hardware Viterbi decoder on the Data Routine Switch.

The source file is \FASSET\A2A11\SYM\CODE\DSRC\ACDECODE.C.

Address	Value	Meaning
10000800	11112222	Start of processing
10000810		16-bit input data (with index in upper bytes)
10000820		8-bit output data (with index in upper bytes)
10000830		Output timeout (zero value)
10000840		Count
10000850		Control Count

Table 8 Decode C2 message addresses

A.1.1.4 Process C2 Message Routine

This routine processes C2 messages. It decodes, performs a CRC check and compiles error statistics for C2 messages.

The source file is \FASSET\A2A11\SYM\CODE\DSRC\PROULAC.C.

Address	Value	Meaning
10000700	11223344	Start of processing
10000710		Frame number
10000720	1 2 3 4	Robust C2 Short C2 Long C2 C2 Message complete
10000730		Slot number
10000740		FNO
10000750		Index
10000760		De-interleave flag – Boolean
10000770		Uncoded data (with index in upper bytes)
10000780		CRC bad flag – Boolean
10000790	0 1 2	Short C2 Long C2 Robust C2
100007A0		Channel
100007B0		Number of bits
100007C0		Data2 (with index in upper bytes)
100007D0		Slot2
100007E0		Repeat
100007F0		Diversity

Table 9 Process C2 message addresses

A.1.2 Additional Information

In addition to the debugging information, there is useful data being passed between the LDR Symbol Processor and Data Routing Switch for hardware Viterbi decoding. The VMETRO that is installed semi-permanently in the payload's LDR Baseband Subsystem sits on the bus and can monitor the VME bus traffic internal to that subsystem. C2 messages can be monitored by examining the bus transactions as the LDR Symbol Processor writes the coded message to the decoder, and then reads back the C2 message without coding.

The data written in 16-bit words to the decoder is comprised of the encoded message followed by flush bits. Because of the delay inherent in this type of decoder, the data (decoded message) is read back in 8-bit bytes and is not valid until after the 22nd write.

The data in this case has been byte-swapped by the LDR Symbol Processor prior to the write, so the endian format matches with the VMETRO analyzer. The addresses to monitor are:

VMETRO Address (24 bit)	Data
xx508004	Decoder write (16-bit)
xx508006	Decoder read (8-bit)

Table 10 Decoder endian format swap addresses

A.2 Ground Terminal

The digital processing portion of the ground terminal resides in a single VME chassis. All transactions between boards (with the exception of Hyperbus communications) can be monitored on this VME bus. Transactions of interest are inter-board transfers and interrupts.

A.2.1 Debugging Information

The Modem Control Processor (MCP) is one of many single board computers in the ground terminal. It is responsible for most of the processing of data once it has been demodulated and prior to modulation.

The original developers left debugging information in the MCP software that writes values to unused addresses. The following sections detail the meaning of the various addresses, values, and parameters used. Since most of the addresses are the same (027FFFF0), any different addresses will be italicized. Values are one of Boolean, integer, unsigned integer or float. Parameters follow on subsequent writes to the same address.

Because the values overlap between routines and are not necessarily in numerical order, the last section has a cross-reference table in numerical order to determine in which section to find more details.

A.2.1.1 Calculate RMS Frequency Track Error Routine

This routine calculates the root mean squared (RMS) frequency track error for LHR and HHR coarse synchronization hops. The source file is cal_r_yt.c.

Address	Value	Parameters	Meaning
027FFFF0	801	Filtered error – float	HHR normal
027FFFF0	802	Filtered error – float Filtered error – float	Switch to LHR track
027FFFF0	803	Filtered error – float	LHR normal
027FFFF0	804	Error count	Increment LHR error count
027FFFF0	805	Sum of errors – float	Exceeded count
027FFFF0	806	Sum of errors – float	Less than count
027FFFF0	807	RMS error – float Tracking_C – float	Prior to exit
027FFFF0	80A	Tracking_C – float	Switching to LHR
027FFFF0	80B	Tracking_C – float	Using HHR
027FFFF0	80C	Tracking_C – float	Using LHR
027FFFF0	80E	Previous adjust – float Filtered error – float Adjustment – float Doppler – float	Using LHR
027FFFF0	814		Using HHR coarse
027FFFF0	815		Increment sync hop count

Table 11 Frequency track error addresses

A.2.1.2 Calculate RMS Time Track Error Routine

This routine calculates the RMS time track error for coarse and fine synchronization hops. The source file is cal_r_yu.c.

Note the addresses that differ from the standard 027FFFF0 are italicized.

Address	Value	Parameters	Meaning
027FFD0			Scaled time error – float
027FFD4			Scaled filter error – float
027FFD8			DL freq word
027FFE0			DL time error – float
027FFE4			DL filtered error – float
027FFE8			DL freq word
027FFFF0	901	Time error – float Coarse a – float Coarse b – float Tracking_C – float	Coarse tracking
027FFFF0	902	Time error – float Fine a – float Fine b – float Tracking_C – float	Fine tracking
027FFFF0	903	Sum of errors – float	Too many estimates
027FFFF0	904	Sum of errors – float	Not enough estimates
027FFFF0	905	RMS error – float	Near end
027FFFF0	906	Prev filter error – float Previous error – float Filtered error – float	In fine tracking
027FFFF0	907	Prev filter error – float Previous error – float Filtered error – float	After estimate update
027FFFF0	908	DL freq word	In tracking
027FFFF0	909	Sum of ratios	Fine switch check

Table 12 Time track error addresses

A.2.1.3 Downlink Frame Interrupt Service Routine

This is the downlink frame interrupt service routine. The source file is dlf_isr.c.

Address	Value	Parameters	Meaning
027FFFF0	820		Copy freq error
027FFFF0	821		Copy time error
027FFFF0	822		Copy automatic gain control (AGC) error
027FFFF0	82A		Reset KG/DDS (1)
027FFFF0	82B		Increment (2)
027FFFF0	82C		Send delay event (3)
027FFFF0		dltod_days ref_estod_days days frame	Always occurs first

Address	Value	Parameters	Meaning
027FFFF0		frames (int_day) frames (all) quarter frames qtr_frames ***OR*** seconds qtr_frames	Either one always occurs second
027FFFF0		crd qtr mod crd crtod	Optionally occurs third
027FFFF0		tsnbr tsnbr tsnbr tsnbr	Always occurs last

Table 13 Downlink frame interrupt addresses

A.2.1.4 Downlink Tracking Routine

This routine processes events while in the downlink tracking mode. The source file is dl_trk.c.

Address	Value	Parameters	Meaning
027FFFF0	2030		Sent C2 xon to TC

Table 14 Downlink tracking addresses

A.2.1.5 Perform LDR Reset Routine

This routine performs a reset of the LDR portion of the ground terminal. The source file is per_r_xv.c.

Address	Value	Parameters	Meaning
027FFFF0	829		Received event
027FFFF0	82D		Before send

Table 15 LDR reset addresses

A.2.1.6 Process C2 Data Routine

This routine processes incoming C2 data from Terminal Controller (including the Xon/Xoff protocol). The source file is pr_c2.c.

Address	Value	Parameters	Meaning
027FFFF0	2030		Sent C2 xon to TC

Table 16 Process C2 data addresses

A.2.1.7 Receive C2 Data Routine

This routine receives incoming C2 data from the Terminal Controller. The source file is rcv_c2.c.

Address	Value	Parameters	Meaning
027FFFF0	2031		Sent C2 xoff to TC

Table 17 Receive C2 data addresses

A.2.1.8 Send Synchronization Status Routine

This routine sends the synchronization status to the Terminal Controller. The source file is snd_t_yr.c.

Address	Value	Parameters	Meaning
027FFFF0	300	Time accuracy Freq accuracy	Before send

Table 18 Send synchronization status addresses

A.2.1.9 Synchronization Routine

This routine contains the synchronization task. The source file is sync_csu.c.

Address	Value	Parameters	Meaning
027FFFF0	1000		DL acquisition
027FFFF0	1001		Spatial enhancement
027FFFF0	1002		DL tracking
027FFFF0	1003		UL acquisition
027FFFF0	1004		Tracking

Table 19 Synchronization addresses

A.2.1.10 Time and Frequency Control Routine

This routine contains the time and frequency control task. The source file is time_csu.c.

Address	Value	Parameters	Meaning
027FFFF0	0		Event received
027FFFF0	4		Done process event
027FFFF0	10		Event: doppler delay
027FFFF0	14		Done doppler delay
027FFFF0	18		After doppler delay
027FFFF0	1C		Done AGC set
027FFFF0	20	DL freq BUIF – Boolean (buffer in use flag)	Event: freq error msg
027FFFF0	24	DL freq BUIF – Boolean	Done freq error msg
027FFFF0	30	DL time BUIF – Boolean	Process time msg
027FFFF0	34	DL time BUIF – Boolean	Before RSL msg
027FFFF0	40		Process RSL msg
027FFFF0	44		Done RSL
027FFFF0	50		Process AGC msg

Address	Value	Parameters	Meaning
027FFFF0	54		Done AGC
027FFFF0	60		Event: Set DC offset
027FFFF0	64		Requested DC offset
027FFFF0	70		Event: Rx DC offset
027FFFF0	74		Done Set DC offset
027FFFF0	80		Event: Set AGC
027FFFF0	84		Done set AGC
027FFFF0	808	RMS freq error – float	After window exceeded
027FFFF0	809	Fine error – float Coarse error – float I fine – float I coarse – float Q fine – float Q coarse – float	
027FFFF0	80D	Number errors	Done RMS freq calc
027FFFF0	80F	Number errors	Done RMS time calc
027FFFF0	810	RMS time error – float	After window exceeded
027FFFF0	812	LHR error – float HHR error – float	In tracking
027FFFF0	830		DL freq flag true
027FFFF0	831		DL time flag true
027FFFF0	832		DL AGC flag true

Table 20 Time and frequency control addresses

A.2.1.11 Address and Value Cross Reference Table

This section contains and address and value cross reference to go from a given address and value to the associated routine in the previous sections. The table is sorted first by address and then by value. Note that some values occur in more than one routine and the ambiguity can only be resolved by context.

Address	Value	Para	Routine
027FFD0		A.2.1.2	cal_r_yu
027FFD4		A.2.1.2	cal_r_yu
027FFD8		A.2.1.2	cal_r_yu
027FFE0		A.2.1.2	cal_r_yu
027FFE4		A.2.1.2	cal_r_yu
027FFE8		A.2.1.2	cal_r_yu
027FFFF0	0	A.2.1.10	time_csu
027FFFF0	4	A.2.1.10	time_csu
027FFFF0	10	A.2.1.10	time_csu
027FFFF0	14	A.2.1.10	time_csu
027FFFF0	18	A.2.1.10	time_csu
027FFFF0	1C	A.2.1.10	time_csu
027FFFF0	20	A.2.1.10	time_csu
027FFFF0	24	A.2.1.10	time_csu
027FFFF0	30	A.2.1.10	time_csu
027FFFF0	34	A.2.1.10	time_csu
027FFFF0	40	A.2.1.10	time_csu
027FFFF0	44	A.2.1.10	time_csu
027FFFF0	50	A.2.1.10	time_csu
027FFFF0	54	A.2.1.10	time_csu
027FFFF0	60	A.2.1.10	time_csu
027FFFF0	64	A.2.1.10	time_csu
027FFFF0	70	A.2.1.10	time_csu
027FFFF0	74	A.2.1.10	time_csu
027FFFF0	80	A.2.1.10	time_csu
027FFFF0	84	A.2.1.10	time_csu
027FFFF0	300	A.2.1.8	snd_t_yr
027FFFF0	801	A.2.1.1	cal_r_yt
027FFFF0	802	A.2.1.1	cal_r_yt
027FFFF0	803	A.2.1.1	cal_r_yt
027FFFF0	804	A.2.1.1	cal_r_yt
027FFFF0	805	A.2.1.1	cal_r_yt
027FFFF0	806	A.2.1.1	cal_r_yt
027FFFF0	807	A.2.1.1	cal_r_yt
027FFFF0	808	A.2.1.10	time_csu
027FFFF0	809	A.2.1.10	time_csu
027FFFF0	80A	A.2.1.1	cal_r_yt

Address	Value	Para	Routine
027FFFF0	80B	A.2.1.1	cal_r_yt
027FFFF0	80C	A.2.1.1	cal_r_yt
027FFFF0	80D	A.2.1.10	time_csu
027FFFF0	80E	A.2.1.1	cal_r_yt
027FFFF0	80F	A.2.1.10	time_csu
027FFFF0	810	A.2.1.10	time_csu
027FFFF0	812	A.2.1.10	time_csu
027FFFF0	814	A.2.1.1	cal_r_yt
027FFFF0	815	A.2.1.1	cal_r_yt
027FFFF0	820	A.2.1.3	dlf_isr
027FFFF0	821	A.2.1.3	dlf_isr
027FFFF0	822	A.2.1.3	dlf_isr
027FFFF0	829	A.2.1.5	per_r_xv
027FFFF0	82A	A.2.1.3	dlf_isr
027FFFF0	82B	A.2.1.3	dlf_isr
027FFFF0	82C	A.2.1.3	dlf_isr
027FFFF0	82D	A.2.1.5	per_r_xv
027FFFF0	830	A.2.1.10	time_csu
027FFFF0	831	A.2.1.10	time_csu
027FFFF0	832	A.2.1.10	time_csu
027FFFF0	901	A.2.1.2	cal_r_yu
027FFFF0	902	A.2.1.2	cal_r_yu
027FFFF0	903	A.2.1.2	cal_r_yu
027FFFF0	904	A.2.1.2	cal_r_yu
027FFFF0	905	A.2.1.2	cal_r_yu
027FFFF0	906	A.2.1.2	cal_r_yu
027FFFF0	907	A.2.1.2	cal_r_yu
027FFFF0	908	A.2.1.2	cal_r_yu
027FFFF0	909	A.2.1.2	cal_r_yu
027FFFF0	1000	A.2.1.9	sync_csu
027FFFF0	1001	A.2.1.9	sync_csu
027FFFF0	1002	A.2.1.9	sync_csu
027FFFF0	1003	A.2.1.9	sync_csu
027FFFF0	1004	A.2.1.9	sync_csu
027FFFF0	2030	A.2.1.4	dl_trk
027FFFF0	2030	A.2.1.6	pr_c2
027FFFF0	2031	A.2.1.7	rcv_c2

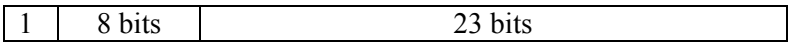
Table 21 Address and value cross reference table

A.2.2 Additional Information

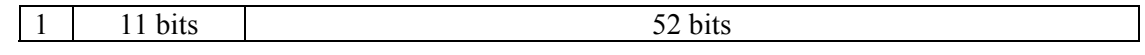
In addition to the debugging information, there is useful data being passed between the various boards on the VME bus. The VMETRO that is installed in the ground terminal can monitor the VME bus traffic. Useful trace information can be derived from the interrupts, board addresses and some inter-board communications buffers. Some of the data passed between boards is in Institute of Electrical and Electronics Engineers (IEEE) floating point format, so the formatting of floating point numbers is also detailed.

A.2.2.1 Floating Point Format

The Modem Control Processor (MCP), Uplink Processor (ULP) and Downlink Processor (DLP) all use floating-point numbers in the IEEE standard 32-bits format. Both the 32-bit float format and 64-bit double format are presented below.



$$\text{Float Value} = \begin{cases} (-1)^{Sign} \times 2^{(Exponent-127)} \times \left(1 + \left[\frac{Fraction}{2^{24}}\right]\right) & ; Exponent \neq 0 \\ \pm 0 & ; Exponent = 0 \end{cases}$$



$$\text{Double Value} = \begin{cases} (-1)^{Sign} \times 2^{(Exponent-1023)} \times \left(1 + \left[\frac{Fraction}{2^{52}}\right]\right) & ; Exponent \neq 0 \\ \pm 0 & ; Exponent = 0 \end{cases}$$

A.2.2.2 Interrupts

Many timing interrupts are generated to synchronize the processing within the ground terminal. The CTOD and emulated system time-of-day (ESTOD) 1 Hz interrupts are used to maintain the associated clocks. (Note that the ground terminal only has an estimate of ESTOD so the interrupt is called ESTOD Est 1Hz whereas the payload has ESTOD 1Hz.) The frame interrupts are used to process data whereas the quarter frame interrupts are used to process the TRANSEC data. The start emulation interrupt occurs at the beginning of the emulation, although it should be noted there are two occurrences of this interrupt.

The sources of interrupts are the Time-of-day (TOD) board and the KG Controller (Key Generation Controller) board. The Modem Control Processor (MCP) receives most of the

interrupts. The quarter frame interrupts are received by two Dataβ digital signal processing boards: Downlink Processor 1 (DLP1) and Uplink Processor (ULP).

To trigger on these interrupts, it is best to use the interrupt acknowledge (IACK) cycle on the VMETRO. The data should be xxxxxx***VV*** where ***VV*** is vector, Iack***I*** where ***I*** is interrupt.

Interrupt	Vector	Source	Destination	Meaning
4	82	TOD	MCP	CTOD Injection
	83			Start Emulation (2 occurrences)
	84			CTOD 1 Hz
	85			ESTOD Est 1 Hz/ESTOD 1Hz
	86			DLTOD Frame
	87			ULTOD Frame
5	91	KGC	DLP1	DLTOD Quarter Frame
6	90	KGC	ULP	ULTOD Quarter Frame

Table 22 Ground terminal interrupts

There may be some other interrupts (interrupt 2 vector 93, interrupt 4 vector A3) used by the serial and parallel boards to notify the processors of external events. There are clues in the software about these interrupts, but none were noticed when examining traces. It is believed that these interrupts are only for MDR, which was never completed.

A.2.2.3 Board Addresses

Below is a table of the various boards in the ground terminal and their positions in the VME chassis. Requirements for cooling, cabling, and space to accommodate the VMETRO bus analyzer cause some of the positions to remain unfilled. The table gives the name and type of board in each position. Also given is the A16 and A32 addresses used by the board. Finally, in the last column is a reference to the Ground Terminal Technical Manual [12] for more details on the board registers.

In a 32-bit address space, A16 addresses are of the format FFFF***aaaa*** where ***aaaa*** is the 16-bit address.

Position	Name	Type	A16	A32	GT Tech
1	System Controller (SC)	147SB-1	0000	0100 0000	
3	Parallel Interface	340A	0400		
4	Serial Interface	333S-2	3900		
5	Terminal Controller (TC)	147SB-1	0010	0200 0000	
7	Modem Control Processor (MCP)	147SB-1	0020	0300 0000	p217
8	Serial Interface	333S-2	3A00		
9	Serial Interface	333-2	3B00		
10	Uplink processor (ULP)	Dataß	F100	4000 0000	
12	Time of day Interface (TOD)	custom		F100 0000	p141, p217
14	Synthesizer Interface (DDS)	custom		2000 0000	p147, p217
16	Downlink Processor 1 (DLP1)	Dataß	F200	5000 0000	p217
17	Downlink Processor 2 (DLP2)	Dataß	F300	6000 0000	
18	Downlink Buffer (DLB)	custom		F200 0000	p157, p217
19	KG Controller (KGC)	custom		3000 0000	p151, p217

Table 23 Ground terminal VME addresses

A.2.2.4 MCP Inter-processor Communications Buffers

The Modem Control Processor controls much of the processing associated with user data and the manipulations necessary for the waveform. It communicates with the Uplink Processor, Downlink Processors and serial port controllers. For these communications, it uses buffers within the address space. The following table lists these buffers.

Buffer	Size	Start Address	End Address
ULPICC to ULPDSP Buffer	0400	03FFE500	03FFE8FF
ULPDSP to ULPICC Buffer	0100	03FFE900	03FFE9FF
ULP to MCP Buffer	0100	03FFEA00	03FFEAFF
MCP to ULP Buffer	0300	03FFEB00	03FFEDFF
DLP to MCP Buffer	0200	03FFEE00	03FFEFFF
MCP to DLP Buffer	0D00	03FFF000	03FFFCFF
TC LDR Test Setup Data	0300	03FFFD00	03FFFFFF

Table 24 Modem Control Processor buffer addresses

A.2.2.5 DLP to MCP Buffer

Communications from the Downlink Processor (specifically DLP1) to the MCP contain many of the results of interest especially when performing synchronization. The details of the buffer were extracted from the software, compared with the partial information in the Ground Terminal Technical Manual [12] and portions were verified by examination of traces. It was found that there was one error in the manual.

The buffer contains several areas that are used to communicate messages from the DLP to the MCP. Each area has an associated buffer-in-use flag (BIUF). In the table below, each address of the buffer and data type is given in the message areas. The BIUF associated with each message area is given in the last column.

Address	Meaning	BIUF Address
	Buffer-in-use Flags	n/a
03FFEE00	C3_Data_BIUF	
03FFEE04	DL_RSL_Data_BIUF	
03FFEE08	AROW_Data_BIUF	
03FFEE0C	DL_AGC_BIUF	
03FFEE10	DL_Frequency_BIUF	
03FFEE14	DL_Time_BIUF	
03FFEE18	DC_Offset_BIUF	
03FFEE1C	DL_Acquisition_Results_BIUF	
03FFEE20	DL_Alarms_BIUF	
03FFEE24	DL_Reset_Complete_BIUF	
03FFEE28	DL_BITE_Results_BIUF	
	C3 Data Message	03FFEE00
03FFEE2C	Word_1	
03FFEE30	Word_2	
03FFEE34	Word_3	
	DL RSL Data Message	03FFEE04
03FFEE38	DL_Frame_Of_Day	
03FFEE3C	Hop_of_Frame	
03FFEE40	RSL_Measurement – float	

Address	Meaning	BIUF Address
	AROW Data Message	03FFEE08
03FFEE44	Word_1	
03FFEE48	Word_2	
03FFEE4C	Word_3	
	DL AGC Magnitude	03FFEE0C
03FFEE50	DL_Frame_Of_Day	
03FFEE54	Hop_of_Frame	
03FFEE58	AGC_Measurement – float	
	DL Frequency Error	03FFEE10
	LHR Sync	
03FFEE5C	DL_Frame_Of_Day	
03FFEE60	Hop_of_Frame	
03FFEE64	Relative_Frequency_Error_Estimate – float	
	HHR Coarse Sync	
03FFEE68	DL_Frame_Of_Day	
03FFEE6C	Hop_of_Frame	
03FFEE70	Relative_Frequency_Error_Estimate – float	
	DL Time Error	03FFEE14
	Coarse	
03FFEE74	DL_Frame_Of_Day	
03FFEE78	Hop_of_Frame	
03FFEE7C	Time_Error_Estimate – float	
03FFEE80	I_Channel_Time_Track_Reference – float	
03FFEE84	Q_Channel_Time_Track_Reference – float	
	Fine	
03FFEE88	DL_Frame_Of_Day	
03FFEE8C	Hop_of_Frame	
03FFEE90	Time_Error_Estimate – float	
03FFEE94	I_Channel_Time_Track_Reference – float	
03FFEE98	Q_Channel_Time_Track_Reference – float	
	DC Offset Response	03FFEE18
03FFEE9C	I_Channel_Sample_Mean – float	
03FFEEA0	Q_Channel_Sample_Mean – float	
	DL Acquisition Results	03FFEE1C
03FFEEA4	Select_Next_Interval	
03FFEEA8	Detection	
03FFEEAC	Hypothesis_Selected (not used?)	
	DL Alarms	03FFEE20
03FFEEB0	LDR_Alarm_Flag_Number	
03FFEEB4	DSP_Alarm_Variable_1	
03FFEEB8	DSP_Alarm_Variable_2	
03FFEEBC	DSP_Alarm_Variable_3	
03FFEEC0	DSP_Alarm_Variable_4	
03FFEEC4	DSP_Alarm_Variable_5	
03FFEEC8	*DL Reset Complete*	03FFEE24

Address	Meaning	BIUF Address
	DL BITE Results	03FFEE28
03FFEECC	Mode_Selection_Test	
03FFEED0	Second_FIFO_Test	
03FFEED4	Hyperbus_Test	
03FFEED8	DLP_A_Bus_Test	
03FFEEDC	DLP_B_Bus_Test	
03FFEEE0	DLP_Hyperbus_Test	

Table 25 Downlink processor to MCP addresses

Annex B Summary of Payload File Changes

Modifications were made to the original software delivered with the FASSET payload. These changes were made to fix problems and enhance the capabilities of the payload especially for the interoperability trials, and for Harris requirements.

Changes were made to the LDR Symbol Processor (part of the LDR Baseband Subsystem) firmware kernel and downloaded software. Changes were also made to the payload screens and main program running on the payload computer.

The original software was delivered on a classified disk that was later painted red and was subsequently referred to as the red disk. This red disk is only used for software development in an appropriate secure room. The payload normally runs with the unclassified disk, which contains the executables and a sanitized version of the payload software (does not contain any classified information).

B.1 Firmware

The kernel for the LDR Symbol Processor resides on an EPROM on the board itself. This kernel is responsible for the basic functionality of the board including communications with the Configuration Computer (via the Sequencing Computer).

The changes made to the kernel were to modify the content and handling of configuration messages for the various C0/C1 and TT&C accesses. Initially, the messages did not include the downlink assignments because the payload used a fixed uplink to downlink mapping scheme. These messages were modified to include fields for downlink modulation and hop assignments.

The table below lists all the firmware changes by processor:

Processor	EPROM	File	Use	Changes
LDR Symbol Processor	AMD 27C020	SYMROM.BIN	Kernel supporting basic functions including message handling	Support TT&C downlink details in messages, trace information

Table 26 LDR Symbol Processor firmware

B.2 Payload Computer (Unclassified disk)

The unclassified disk is the one that is normally kept in the payload and used to boot and run the system. Along with the operating system, it contains the FASSET executable, user interface screens, supporting files and software downloads.

Most of the changes here were made to allow downlink assignments (instead of fixed uplink to downlink mapping). This changed the setup screens for the various accesses and the necessary additional database entries resulted in larger configuration files.

The table below lists all the files changed on the unclassified disk.

File	Use	Changes
\FASSET\A2A11\SYM\CODE\ SYMDATAP.ABS SYMDATAP.BIN SYMDATAP.MAP	Intermediate files for download software for LDR Symbol Processor	Support flexible TT&C mapping, additional trace information

File	Use	Changes
\FASSET\CONFIGSW\PLDCONF\ *.PAY	Test configurations	Add TT&C entries
\FASSET\CONFIGSW\PLDCONF\ __EMDATA.FST	Default values for database	Add TT&C entries
\FASSET\CONFIGSW\PLDCONF\ __HMIDEF.FST	Main program user interface screens	LDR Communications Parameters Screen supports multiple users, add hints TT&C, User Data Access #1-4 Details Screens added Local Emulation Active State Screen now shows file name Configure AC/AROW Parameters Screen add source field Local Emulation Select State Screen add version
\FASSET\CONFIGSW\PLDCONF\ __SYM0__.FST __SYM1__.FST __SYM2__.FST	Download software for LDR Symbol Processor	Three copies of SYMDATAP.BIN One for each robustness map (least, medium, most). Only least is used and then only partially.
\FASSET\CONFIGSW\PLDCONF\ PLD_CONF.EXE	Main program	All modification made to screen library, messaging and database

Table 27 Modified payload unclassified files

As shown in the above table, the user interface screens were stored on file __HMIDEF.FST. Details of screen modifications are given in the following table:

Screen	Title	Modification
16	Local Emulation Select State Screen Ver. 25 Feb 2003	Add version date
28	Configure LDR Communication Parameters Screen	Move TT&C details to screen 33 Add links to screens 33, 53-56 Update hints and reformat
33	TT&C Data Access Detail (formerly Access #1-4 UL Details)	Uplink details from screen 28 New downlink details New action token 44 Return to screen 28 Add hints
34	Configure AC/AROW Parameters Screen	Add source for C3 message Reformat
36	Local Emulation Active State Screen	Add configuration file name parameter
53	Data Access #1 Detail (formerly DL Details for Access #1)	Uplink details from screen 33 Use existing downlink details Use existing action token 40 Return to screen 28 Add hints

Screen	Title	Modification
54	Data Access #2 Detail (new screen)	Uplink details from screen 33 New downlink details New action token 41 Return to screen 28 Add hints
55	Data Access #3 Detail (new screen)	Uplink details from screen 33 New downlink details New action token 42 Return to screen 28 Add hints
56	Data Access #4 Detail (new screen)	Uplink details from screen 33 New downlink details New action token 43 Return to screen 28 Add hints

Table 28 Modification to payload user interface screens

B.3 Red Disk (Classified disk)

The red disk contains all the software (including classified portions of the code). Any changes to executable or downloadable software and firmware were made on the red disk. The compiled and linked version was then transferred to the payload computer for normal operation. This could be done because while the source code is classified, the associated executable is considered to be unclassified.

All changes affected the red disk, with one exception. The main program for the user interface screens are defined in a text file, residing on the unclassified disk, that is read in at run-time. Since it contains no classified information, all modifications were made on the unclassified disk.

The table below lists all changes to files on the red disk.

File	Use	Changes
\FASSET\A2A11\SYM\CODE\BUILD\ SYM_ROM.BAT	Generate symbol processor kernel	Fix bug in copy
\FASSET\A2A11\SYM\CODE\DSRC\ ACDECODE.C	Decode C2 messages	Output data only valid after decoder delay + 1 Add trace information
\FASSET\A2A11\SYM\CODE\DSRC\ BITSTUFF.C	Copy bits into a bit field	Zero accumulator when byte aligned output
\FASSET\A2A11\SYM\CODE\DSRC\ FORMATDP.C	Format DPSK data	Support multiple hop C1
\FASSET\A2A11\SYM\CODE\DSRC\ GENDLAC.C	Generate C3 messages	Add source and 3 x bits, support old random, new raw and new message for C3
\FASSET\A2A11\SYM\CODE\DSRC\ LRDATAFO.C	Downlink data formatting	Expand trace information, TT&C use flexible mapping, C1 use flexible mapping
\FASSET\A2A11\SYM\CODE\DSRC\ PRODLAC.C	Process C3 messages	Add source and 3 x bits to call to GENDLAC and PRODLAC for C3

File	Use	Changes
\FASSET\A2A11\SYM\CODE\DSRC\ PROULAC.C	Process C2 messages	Add trace information
\FASSET\A2A11\SYM\CODE\DSRC\ SYMDATAP.C	Download software for LDR Symbol Processor	Add source and 3 x bits to call to PRODLAC for C3
\FASSET\A2A11\SYM\CODE\INC\ DLEXTERN.H	Downlink external definitions	Add source and 3 x bits to call to GENDLAC and PRODLAC
\FASSET\A2A11\SYM\CODE\INC\ SYMMBOX.H	Message mailbox definitions	Rename DL_Diversity to DL_Spare
\FASSET\A2A11\SYM\CODE\KSRC\ CONFIGSY.C	Process configuration messages	Rename DL_Diversity to DL_Spare, add trace info, fix TT&C messaging
\FASSET\CONFIGSW\HMI\ INITEPDL.C	Initialize data base entries	Initialize TT&C entries
\FASSET\CONFIGSW\INCLUDE\ EPDOFSET.H	Data base offset definitions	Rename DL_Diversity to DL_Spare, add DL entries for TT&C
\FASSET\CONFIGSW\INCLUDE\ HMIDEFS.H	User interface screen definitions	Modify screen maximums, rename and add tokens for multiple accesses and TT&C, increase database size
\FASSET\CONFIGSW\INCLUDE\ SEQDEFS.H	Sequencer definitions	Rename DL_Diversity to DL_Spare
\FASSET\CONFIGSW\PLDCONF\ ACTION_H.C	User interface actions	Make hop-to-bin parameterized, rename token, add actions for multiple accesses and TT&C
\FASSET\CONFIGSW\PLDCONF\ GSCCCNFG.C	Sequencer for C0/C1 access messages	Rename DL_Diversity to DL_Spare, use AC1 parameters for TT&C maintenance access, use TTC parameters for TT&C access

Table 29 Modified payload classified files

Annex C Test Matrices

The test matrices were developed to provide Harris with a benchmark set of tests. These tests were used to validate acceptance testing and proper functioning, post-delivery. The core of the C0 tests were based on the tests used in the interoperability trials at Lincoln Lab. Additional tests of specific interest to Harris were added. These tests are not meant to be a comprehensive set of tests, rather they are meant to be representative tests that cover all of the key options at least once.

The data tests involve user data (C0), secondary user data (C1), and payload TT&C. There are also multi-user tests that support more than one access to the payload at a time. In addition, there are access control tests for uplink (C2) and downlink (C3) communications.

All of these tests have been successfully run on FASSET. Limitations to the capabilities of FASSET have been avoided in creating these tests. For example, 75 b/s C0 does not work with coding, therefore there is no test for that mode.

These test matrices have been used for training. With three exceptions, these tests are under Harris configuration control and may be later modified. The three exceptions are STANDARD, STANDARDH and UNCODED which are often used during debugging.

In the tables below, the file extension ".PAY", should be added to each filename for the payload. For the ground terminal, the file extension is ".GT".

C.1 Standard Test Matrix

The special user data (C0) test called STANDARD is used most often when testing FASSET and should be used when problems occur. A version of STANDARD with coding turned off is called UNCODED. STANDARDH is another variant of STANDARD that uses a high hop-rate (HHR) synchronization hop instead of a low hop-rate (LHR) synchronization hop during downlink acquisition. All standard tests are used primarily when debugging and are under DRDC Ottawa configuration control.

File Name	End-to-end			Uplink					Downlink
	Rate	Coding	Interleave	Mod	Super	Group	Chan	Mode	Mod
STANDARD	2400	Soft	Med Con	FSK/2	A	8	2	1	DPSK4+108
STANDARDH	2400	Soft	Med Con	FSK/2	A	8	2	1	DPSK4+108
UNCODED	4800	*None*	*None*	FSK/2	A	8	2	1	DPSK4+108

Table 30 Standard test matrix

C.2 User Data (C0) Test Matrix

These tests check the main user data ports for the ground terminal covering coded data rates from 75 b/s to 2400 b/s (uncoded rates 150 b/s to 4800 b/s).

It should be noted that STANDARD (see previous section) and C0DATA01 are identical tests and only differ by configuration control.

File Name	End-to-end			Uplink					Downlink
	Rate	Coding	Interleave	Mod	Super	Group	Chan	Mode	Mod
C0DATA01	2400	Soft	Med Con	FSK/2	A	8	2	1	DPSK4+108
C0DATA02	2400	Soft	Med Con	FSK/2	A	8	2	1	DPSK2+54
C0DATA03	2400	Soft	Med Con	FSK/4	A	8	0	5	DPSK4+108
C0DATA04	2400	Soft	*None*	FSK/8	A	8	3	7	DPSK2+54
C0DATA05	2400	Soft	Long Con	FSK/4	A	20	15	5	DPSK1+27
C0DATA06	2400	Soft	Short Con	FSK/4	A	20	7	6	DPSK2+12

File Name	End-to-end			Uplink					Downlink Mod
	Rate	Coding	Interleave	Mod	Super	Group	Chan	Mode	
C0DATA07	2400	Soft	Long Con	FSK/2	A	36	1	1	DPSK1+6
C0DATA08	2400	Hard	Med Con	FSK/2	B	8	1	3	DPSK2+54
C0DATA09	2400	Hard	Short Con	FSK/2	B	20	14	4	DPSK2+54
C0DATA10	2400	Soft	Med Con	FSK/2	C	8	2	4	DPSK2+54
C0DATA11	150	*None**	*None**	FSK/64	A	8	3	1	FSK/1
C0DATA12	150	*None**	*None**	FSK/128	A	20	4	6	FSK/1
C0DATA13	150	*None**	*None**	FSK/64	B	8	2	1	FSK/2
C0DATA14	150	Soft	*None**	FSK/32	C	36	17	2	FSK/1
C0DATA15	300	Soft	Short Con	FSK/16	B	36	31	1	DPSK2+12
C0DATA16	300	Soft	Short Con	FSK/16	B	20	13	3	DPSK1+6
C0DATA17	600	Soft	Short Con	FSK/8	A	20	1	2	DPSK2+12
C0DATA18	1200	Soft	Short Con	FSK/4	C	20	5	1	DPSK2+54
C0DATA19	1200	Hard	Short Con	FSK/8	B	8	1	5	DPSK1+27

* These options do not work when turned on

Table 31 User data (C0) test matrix

C.3 Secondary User Data (C1) Test Matrix

These tests check the secondary user data ports for the ground terminal that supports coded data rates from 75 b/s to 300 b/s (uncoded rates 150 b/s to 600 b/s).

File Name	End-to-end			Uplink					Downlink Mod
	Rate	Coding	Interleave	Mod	Super	Group	Chan	Mode	
C1DATA01	300	Soft	Short Con	FSK/2	A	8	2	1	DPSK2+12
C1DATA02	150	None	None*	FSK/8	A	8	2	5	FSK/1

Table 32 Secondary user data (C1) test matrix

C.4 User Data to/from Payload (TT&C) Test Matrix

These tests check the communications between the ground terminal and the TT&C port on the payload (coded data rate 2400 b/s and uncoded rate 4800 b/s). One of the multi-user tests in the following section includes a TT&C link.

Each of these tests can be used in three different BER test set configurations. There is a configuration to test the uplink only, downlink only, or with the payload TT&C source switch in loopback position (end-to-end, both links).

File Name	End-to-end			Uplink					Downlink Mod
	Rate	Coding	Interleave	Mod	Super	Group	Chan	Mode	
TTC01	2400	Soft	Med Con	FSK/2	A	8	2	1	DPSK4+108
TTC02	2400	Soft	Short Con	FSK/2	A	8	2	1	DPSK2+54
TTC03	4800	*None*	*None*	FSK/2	A	36	23	2	DPSK2+54
TTC04	2400	Soft	Med Con	FSK/4	C	20	12	6	DPSK1+27

(see also MUDATA03)

Table 33 Test matrix for TT&C

C.5 Multi-user Data Test Matrix

These tests check multiple independent accesses using the C0 data ports on the ground terminal and in one case using the TT&C port of the payload. In the duplex test, one ground terminal user port is transmitting to the other ground terminal user port in the same manner as would be used for a duplex connection.

File Name	Access	End-to-end			Uplink					Downlink Mod
		Rate	Coding	Interleave	Mod	Super	Group	Chan	Mode	
MUDATA01	#1→#1	2400	Soft	Med Con	FSK/2	A	8	2	1	DPSK4+108
	#2→#2	2400	Soft	Med Con	FSK/2	A	8	0	2	DPSK4+108
MUDATA02	#3→#3	2400	Soft	Med Con	FSK/2	A	8	2	3	DPSK4+108
	#4→#4	2400	Soft	Med Con	FSK/2	A	8	2	4	DPSK4+108
MUDATA03 (TT&C)	#1→#1	2400	Soft	Med Con	FSK/2	A	8	2	1	DPSK4+108
	TT&C	2400	Soft	Med Con	FSK/2	A	8	2	2	DPSK4+108
MUDATA04 (Duplex)	#1→#2	2400	Soft	Med Con	FSK/2	A	8	2	1	DPSK4+108
	#2→#1	2400	Soft	Med Con	FSK/2	A	8	2	2	DPSK4+108

Table 34 Multi-user data test matrix

C.6 Uplink Access Control (C2) Test Matrix

These tests check the uplink access control (C2) capabilities. C2's are generated in the ground terminal using either random messages or a fixed bit pattern. Note that the method of specifying the bit pattern on the ground terminal is bit-reversed. C2's are received in the payload and may be examined using the VMETRO bus analyzer. Statistics on the C2 CRC check, pass or fail, can be collected and displayed using the payload user interface.

Note that there are several limitations involved in these tests, and several options are unavailable. The cycle column specifies the entry in Table IX on page 33 of the FASSET LDR Data Link Specification [9].

File Name	Uplink									
	Super	Group	Chan	Type	Cycle	Slot	Div	Rate	Interleave	Cover
C2DATA01	A	8	0	Long	3*	1	8*	300	On	Off*
C2DATA02	B	20	3	Short	3*	1	8*	300	Off	Off*
C2DATA03	C	36	23	Long	3*	2	8*	300	Off	Off*

* This is the only valid value for this column

Table 35 Uplink access control (C2) test matrix

C.7 Downlink Access Control (C3) Test Matrix

These tests check the downlink access control (C3) capabilities. C3's are generated in the payload using either random messages, fixed messages or a fixed bit pattern. C3's are received by the ground terminal but cannot be examined. Statistics on the C3 CRC check (including terminal ID validation) can be collected and displayed using the ground terminal interface.

File Name	Downlink							
	ID	Robust	Div	Mod	Doubled	Rate	Interleave	Cover
C3DATA01	LGT2	Least*	Least	DPSK 2+12	Off	2400	On	On
C3DATA02	LGT2	Least*	Most	DPSK 1+6	Off	1200	On	On
C3DATA03	LGT2	Least*	Least	DPSK 2+12	On	4800	On	On
C3DATA04	LGT2	Least*	Most	DPSK 1+6	On	2400	On	On

* This is the only valid value for this column

Table 36 Downlink access control (C3) test matrix

Annex D Custom Software Development Tools

To facilitate modifications of payload software, two programs were developed. ABStoBIN aided in preparing the linked modules for download or for programming into EPROM. The CountFields application was used to determine the maximum number of fields needed for the user interface of the Configuration Computer. This data was then used to intelligently minimize the size of user interface structures and arrays.

D.1 ABStoBIN

The procedure documented in the Payload Technical Manual [11] for regenerating downloadable software or EPROM firmware involves the use of a special EPROM programmer and requires some hand entry of values. It was desired to automate these procedures as much as possible and to allow the use of the SuperPro II EPROM programmer. Software was developed to change the format, fill holes, and insert a checksum value if necessary.

The ABStoBIN program takes the ".ABS" file generated by the build, the ".MAP" file generated during linking and then outputs a ".BIN" which can be downloaded by the payload executable or immediately loaded into the EPROM programmer to burn an EPROM.

The mode (download or ROM), the applicable files, and other details can be found in the ABStoBIN.txt, which is read when the program is invoked. Comments are preceded by a semicolon ";" and can appear anywhere on a line. Blank lines are allowed. An example of this file follows.

```
; abstobin.txt
;
; configuration file for abstobin
;

c:\fasset\roms\symdatap.abs   ; Intel hex format source file
c:\fasset\roms\symdatap.map   ; Link map used to obtain checksum address for ROM
c:\fasset\roms\symdatap.bin   ; Binary output file
0                             ; Fill value for unused locations
0x0                           ; Base address for ROM=0xF0C00000 and for DL=0x0
Download                      ; Download or ROM
```

In this example, symdatap.abs is the source file, and symdatap.map is not used because the operation specified is a download, and symdatap.bin is the output file. Any address not specified in the source file will be filled with 0's. The base address for the download is 0x0. The last line specifies that this is a download.

A ROM case would differ in three ways. The base address would be 0xF0C00000 and the last line would read ROM. The other difference is that the link map file would be used to determine the location of the checksum field and the appropriate checksum would be automatically inserted. This ensures that the ROM passes the power-up checksum test.

To use ABStoBIN, it is first necessary to edit the ABStoBIN.txt file to match the desired configuration. The ABStoBIN application is then invoked to generate the output. For ROMs, this is then loaded in binary format into the programmer. For download, the output is copied onto the unclassified disk. For further information, see Section 6.3 on regenerating LDR Symbol Processor software.

D.2 CountFields

The application CountFields is used during modifications to the user interface of the payload's Configuration Computer. It counts the number of fields in the file "__HMIDEF.FST", and determines the maximum number of fields required for the user interface. This data is then used to intelligently minimize the size of user interface structures and arrays.

The user interface screens for the payload are implemented using a generic engine with a custom action handler linked in. Other than the actions, the exact screen details are read in at run-time from the text file "__HMIDEF.FST". Changing the screens normally requires modification to only this file. If new actions are needed then the screen library must be recompiled after implementing the changes in the action handler "ACTION_H.C" and adding action tokens to "HMIDEFS.H".

The screen details are read in and stored in internal structures and arrays that have a fixed size specified at compile time. When screens or fields are added, these sizes may be exceeded. This will result in errors in execution and should be avoided. Furthermore, when screens or fields are removed, there is excess capacity wasted. The current FASSET executable is very close to the DOS size limit and further changes might result in it being too large.

To ensure that the requirements are met but not grossly exceeded, CountFields was developed to parse "__HMIDEF.FST" to determine all of the maximum values necessary. These values are slightly increased to allow for minor programming problems (such as an index going from 0 to N when the array is dimensioned to be size N). The resultant values are then used to set the maximums in "HMIDEFS.H". See Section 6.2.3 for more details on original, required and current maximums.

The path and file name are hard coded into the program. If it desired to change the name or path then the file "CountFields.c" must be edited and then rebuilt. Once invoked, the program parses the file and reports the total number of screens, fields (3 types), and the maximum number of fields used by any one screen.

List of acronyms

AC	Access control (logon protocol)
	Alternating current (power)
ACQ	Acquisition
AGC	Automatic gain control
AROW	Acquisition response order wire
AUI	Attachment unit interface
BBRAM	Battery backed-up random access memory
BER	Bit error rate
BITE	Built-in test equipment
BIUF	Buffer-in-use flag
BNC	Bayonet Neill-Concelman
b/s	Bits per second
C3I	Command, control, communications, and information
Chan	Channel
CON	Convolutional
COTS	Commercial-off-the-shelf
CPU	Central processing unit
CRC	Communications Research Centre (government agency)
	Cyclic redundancy check (error detection code)
CTOD	Calibrated time-of-day
CW	Continuous wave
DC	Direct current
DDS	Direct digital synthesizer
DERA	Defence Evaluation and Research Agency
DIU	Digital interface unit
DL	Downlink
DLB	Downlink buffer
DLP	Downlink processor
DND	Department of National Defence
DOS	Disk operating system
DPSK	Differential phase shift keying
DRDC	Defence Research and Development Canada
DREO	Defence Research Establishment Ottawa
EHF	Extremely high frequency
EPLD	Erasable programmable logic device
EPROM	Erasable programmable read only memory
ESTOD	Emulated system time-of-day
FASSET	Functional Advanced Development Model of an EHF Satellite
	Communications System for Evaluation and Test
FDMA	Frequency division multiple access
FEC	Forward error correction
FIFO	First-in, first-out
FSK	Frequency shift keying
GPIB	General purpose interface bus

GT	Ground terminal
HDD	Hard disk drive
HHR	High hop rate
HP	Hewlett Packard
IACK	Interrupt acknowledge
ID	Identification
IEEE	Institute of Electrical and Electronics Engineers
IF	Intermediate frequency
I/O	Input/Output
IP	Internet protocol
IT	Information technology
KGC	Key generation controller
LAN	Local area network
LDR	Low data rate
LED	Light emitting diode
LGT	Laboratory ground terminal (LGT1 is synonymous with LGT)
LHR	Low hop rate
LNA	Low noise amplifier
LO	Local oscillator
LRU	Line replaceable unit
LSB	Least significant bit
MAC	Media access control (ethernet protocol)
	Monitor and alarm controller (FASSET subsystem)
MCP	Modem control processor
MDR	Medium data rate
MilSatCom	Military satellite communications
MIL-STD	Military standard
MIT	Massachusetts Institute of Technology
Mod	Modulation
MSB	Most significant bit
MU	Multi-user
NVRAM	Non-volatile random access memory
OS	Operating System
PC	Personal computer
PCB	Printed circuit board
PCI	Payload Controller Interface
PLCC	Plastic leaded chip carrier
PLD	Payload
PLL	Phase locked loop
PSU	Power supply unit
RAM	Random access memory
RCVR	Receiver
RF	Radio frequency
RGT	Rugged ground terminal (LGT2 is synonymous with RGT)
RMS	Root mean squared
ROM	Read only memory
RSL	Receive Signal Level
RX	Receive
SATCOM	Satellite communications
SAW	Surface acoustic wave
SBC	Single board computer

SBIRS	Space Based Infrared Systems
SC	System controller
SCSI	Small computer system interface
SHF	Super high frequency
SMA	Sub-miniature version A
Super	Supergroup
TC	Terminal controller
TCP	Transmission control protocol
TCS	Timing Control Subsystem
TDM	Time division multiplexing
TDMA	Time division multiple access
TOD	Time-of-day
TP	Technical panel
TRANSEC	Transmission security
TRANSEM	Transceiver emulator (Ground terminal Receiver/Transmitter Subsystem)
TSNBR	Cryto time slow number
TT&C	Tracking, telemetry & command
TTCP	The Technical Cooperation Program
TX	Transmit
UL	Uplink
ULP	Uplink Processor
UK	United Kingdom
US	United States of America
VME	VERSA module Eurocard
XMTR	Transmitter

DOCUMENT CONTROL DATA

(Security classification of title, body of abstract and indexing annotation must be entered when the overall document is classified)

1. ORIGINATOR (the name and address of the organization preparing the document. Organizations for whom the document was prepared, e.g. Establishment sponsoring a contractor's report, or tasking agency, are entered in section 8.)

Defence R&D Canada – Ottawa
3701 Carling Avenue, Ottawa, ON, Canada, K1A 0Z4

2. SECURITY CLASSIFICATION (overall security classification of the document, including special warning terms if applicable)

UNCLASSIFIED

3. TITLE (the complete document title as indicated on the title page. Its classification should be indicated by the appropriate abbreviation (S,C or U) in parentheses after the title.)

FASSET Refurbishment (U)

4. AUTHORS (Last name, first name, middle initial)

Liao, B.; Addison, R.; Lambert, J. D.

5. DATE OF PUBLICATION (month and year of publication of document)	6a. NO. OF PAGES (total containing information. Include Annexes, Appendices, etc.)	6b. NO. OF REFS (total cited in document)
July 2004	116	22

7. DESCRIPTIVE NOTES (the category of the document, e.g. technical report, technical note or memorandum. If appropriate, enter the type of report, e.g. interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered.)

Technical Memorandum

8. SPONSORING ACTIVITY (the name of the department project office or laboratory sponsoring the research and development. Include the address.)

Defence R&D Canada – Ottawa
3701 Carling Avenue, Ottawa, ON, Canada, K1A 0Z4

9a. PROJECT OR GRANT NO. (if appropriate, the applicable research and development project or grant number under which the document was written. Please specify whether project or grant)	9b. CONTRACT NO. (if appropriate, the applicable number under which the document was written)
	2948253

10a. ORIGINATOR'S DOCUMENT NUMBER (the official document number by which the document is identified by the originating activity. This number must be unique to this document.)	10b. OTHER DOCUMENT NOS. (Any other numbers which may be assigned this document either by the originator or by the sponsor)
DRDC Ottawa TM 2004-134	

11. DOCUMENT AVAILABILITY (any limitations on further dissemination of the document, other than those imposed by security classification)

(X) Unlimited distribution
( ) Distribution limited to defence departments and defence contractors; further distribution only as approved
( ) Distribution limited to defence departments and Canadian defence contractors; further distribution only as approved
( ) Distribution limited to government departments and agencies; further distribution only as approved
( ) Distribution limited to defence departments; further distribution only as approved
( ) Other (please specify):

12. DOCUMENT ANNOUNCEMENT (any limitation to the bibliographic announcement of this document. This will normally correspond to the Document Availability (11). However, where further distribution (beyond the audience specified in 11) is possible, a wider announcement audience may be selected.)

13. ABSTRACT (a brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C), or (U). It is not necessary to include here abstracts in both official languages unless the text is bilingual).

The FASSET system is an extremely high frequency (EHF) satellite communications payload and ground terminal test bed, originally developed under contract for Defence R&D Canada – Ottawa (DRDC Ottawa). In July of 2002, DRDC Ottawa signed a contract with Harris Corporation that resulted in the refurbishment and delivery of FASSET to assist in the development of their EHF modem. This document details the repairs and improvements that were conducted for Harris in support of the contract. The primary goal of the refurbishment was to reconstitute FASSET to the state of functionality that it possessed prior to being stored for many years.

This report covers the significant refurbishment events including items repaired, modifications and improvements, and unresolved problems. It provides a technical reference of knowledge collected as well as a record and explanation of refurbishment events. Other topics covered include test equipment, maintenance procedures, lessons learned, and future work.

14. KEYWORDS, DESCRIPTORS or IDENTIFIERS (technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifiers such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible keywords should be selected from a published thesaurus. e.g. Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus-identified. If it is not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title.)

FASSET
refurbishment
Harris Corporation
military satellite communications
payload
ground terminal

Defence R&D Canada

Canada's leader in defence
and national security R&D

R & D pour la défense Canada

Chef de file au Canada en R & D
pour la défense et la sécurité nationale

www.drdc-rddc.gc.ca