NASA Technical Memorandum

NASA TM-86483

IMPROVING THE SPACELAB MASS MEMORY UNIT TAPE LAYOUT WITH A SIMULATION MODEL

(NASA-TM-86483) IMPROVING THE SPACELAB MASS MEMORY UNIT TAPE LAYOUT WITH A SIMULATION MODEL (NASA) 136 p HC A07/MF A01 CSCL 09B

N85-14571

Unclas

G3/60 11465

By Steven R. Noneman
Systems Analysis and Integration Laboratory

December 1984

National Aeronautics and Space Administration

George C. Marshall Space Flight Center

TECHNICAL REPORT STANDARD TITLE PAGE

1. REPORT NO. NASA TM-86483	2. GOVERNMENT ACCESSION NO.	3. RECIPIENT'S CATALOG NO.

4. TITLE AND SUBTITLE Improving the Spacelab Mass Memory Unit Tape Layout with a Simulation Model	5. REPORT DATE December 1984 6. PERFORMING ORGANIZATION CODE
7. AUTHOR(S) Steven R. Noneman	8. PERFORMING ORGANIZATION REPORT #
9. PERFORMING ORGANIZATION NAME AND ADDRESS NASA-Marshall Space Flight Center Marshall Space Flight Center, Alabama 35812	10. WORK UNIT NO. 11. CONTRACT OR GRANT NO. 13. TYPE OF REPORT & PERIOD COVERED
12. SPONSORING AGENCY NAME AND ADDRESS National Aeronautics and Space Administration Washington, D. C. 20546	Technical Memorandum 14. SPONSORING AGENCY CODE

15. SUPPLEMENTARY NOTES

Prepared by Systems Analysis and Integration Laboratory, Science and Engineering Directorate

16. ABSTRACT

A tape drive called the Mass Memory Unit (MMU) stores software used by Spacelab computers. MMU tape motion must be minimized during typical flight operations to avoid a loss of scientific data. A projection of the tape motion is needed for evaluation of candidate tape layouts. A computer simulation of the scheduled and unscheduled MMU tape accesses is developed for this purpose. This simulation permits evaluations of candidate tape layouts by tracking and summarizing tape movements. The factors that affect tape travel are investigated and a heuristic is developed to find a "good" tape layout. An improved tape layout for Spacelab I is selected after the evaluation of fourteen candidates. The simulation model will provide the ability to determine MMU layouts that substantially decrease the tape travel on future Spacelab flights.

17. KEY WORDS Mass Memory Unit Tape motion Computer simulation	18. DISTRIBUTION STATEMENT Unclassified-Unlimited

19. SECURITY CLASSIF. (of this report)	20. SECURITY CLASSIF. (of this page)	21. NO. OF PAGES	22. PRICE
Unclassified	Unclassified	134	NTIS

For sale by National Technical Information Service, Springfield, Virginia 22151

TABLE OF CONTENTS

PRECEDING PAGE BLANK NOT FILMED

LIST OF TABLES

LIST OF TABLES (CONTINUED)

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

ATB	Average travel between operations
BETA	Weighting factor of the measures of performance
c	Number of operations where $k > 1$
CDMS	Command and Data Management Subsystem
DDS	Data Display System
DEP	Dedicated Experiment Processor
d(i)	Data-set i of K data-sets
DV	Ordered list of data-sets
ECIO	Experiment Computer Input/Output
ECOS	Experiment Computer Operating System
f	Tape travel function
g	Data-set positioning function
HRM	High Rate Multiplexer
i	An integer number ($\neq$ j)
I	Ordered set of CDMS operations
ISKIP1	Tape travel between data-sets within an operation
IBM	International Business Machines, Inc.
I/O	Input/Output
j	An integer number ($\neq$ i)
JSK	Total blocks skipped within an operation

k	Number of data-sets accessed within an operation
K	Number of data-sets with variable tape positions
m	Number of scheduled operations
MDM	Multiplexer Demultiplexer
MMU	Mass Memory Unit
MTI	Maximum tape travel to a data-set in an operation
MTL	Master Timeline
n	Number of unscheduled operations
NASA	National Aeronautics and Space Administration
NNI	Number of evaluations not improved
NSK	Blocks skipped between operations
P[d(i)]	Tape position of data-set, d(i)
POCC	Payload Operations Control Center
s	A scheduled operation
SCOS	Subsystem Computer Operating System
SC	Layout comparison score
SL-1	Spacelab Mission I
SSIO	Subsystem Input/Output
STL	Subordinate Timeline
UDS	User Data-set
v(i)	Unscheduled operation i
V	Ordered set of unscheduled operations
w(i)	Scheduled operation i
W	Ordered set of scheduled operations

CHAPTER 1.

INTRODUCTION

1.1 Background

The Space Shuttle will be carrying on many of its missions a facility called Spacelab to support scientific investigations. Spacelab is a set of equipment installed in the Shuttle payload bay (as shown in figure 1) which transforms the Shuttle into an orbiting laboratory for space science. Spacelab will be equipped with computers to support operations of the Spacelab equipment and its scientific payload of experiments. The software that the computers use is stored on a magnetic tape storage device called a Mass Memory Unit or MMU.

The Spacelab design [24] was established in the mid-1970s, and in turn, the flight-qualified computer systems were constrained to computers with only 65,536 sixteen-bit word memories and the magnetic tape mass storage

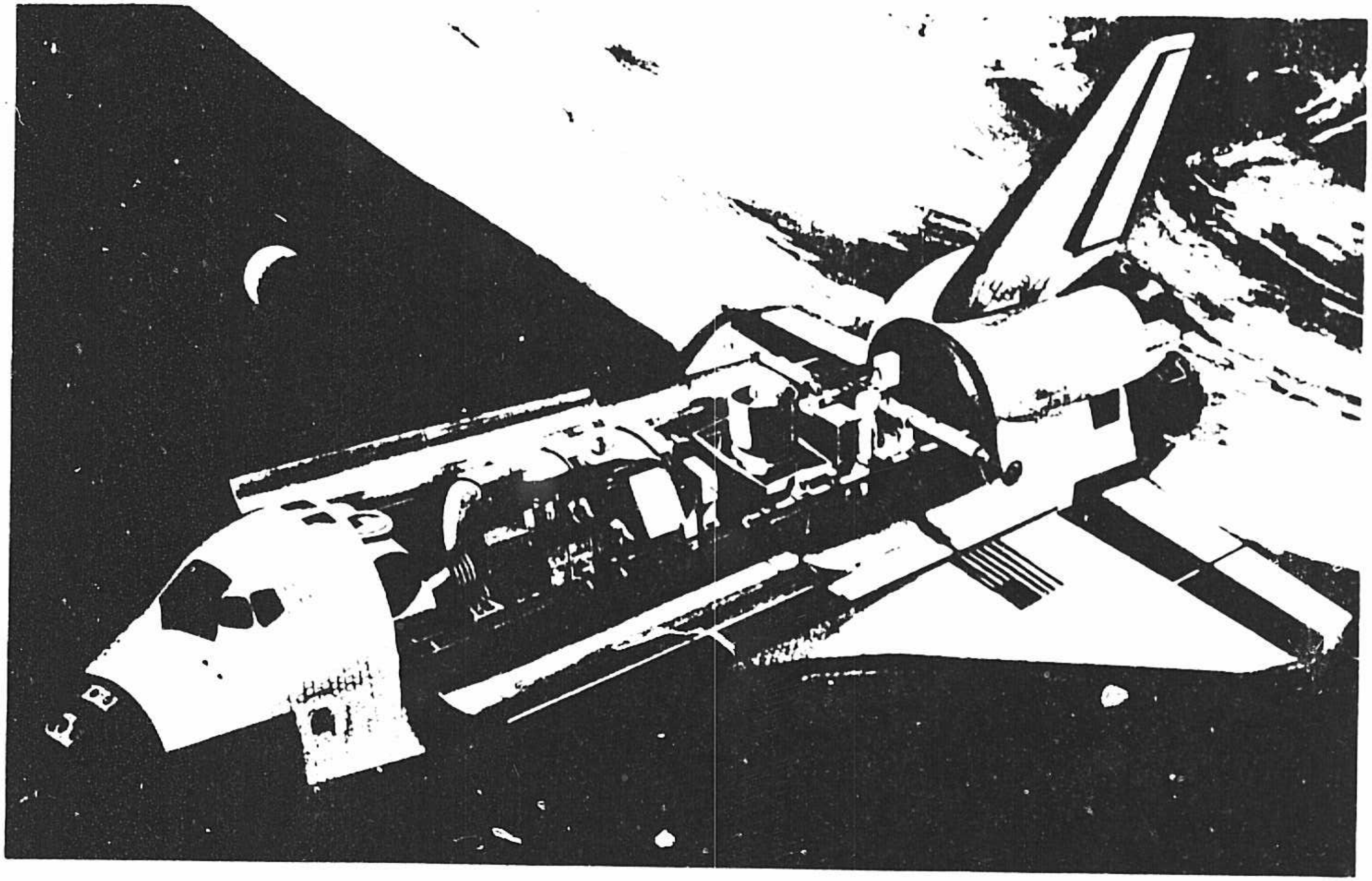

FIGURE 1. SPACELAB IN THE SPACE SHUTTLE

device. Because of the small memory size, and the large number of functions supported by the experiment computer, software must be retrieved frequently from the MMU tape. Since the MMU tape drive is the only space flight qualified mass storage device available, the time required to access software can be large by today's standards. Software access times will vary with the tape travel necessary to reposition from one software module to another. That is, the greater the distance between two software modules on the tape, the longer the time needed to access one and then the other. It is, therefore, desirable to minimize the tape travel, and thus the software access time.

The accesses of MMU software modules, commonly called data-sets, are made to support hundreds of operations during a Spacelab flight. These operations may involve setting up an experiment, conducting an experiment, displaying information to the Spacelab astronauts, and many other functions. Because of the potential for large MMU access times, there are concerns that these software operations might take longer to perform than is expected or necessary. Unexpectedly long access times might even result in operations performed incorrectly or at the wrong time. Spacelab missions may have specific operations which need small access times. It may also be desired that tape travel is minimized for all of the operations performed. Thus, the definition of good tape positions for the data-sets will vary with the needs of the specific mission.

In the past, the tape positions of the data-sets have been selected in the following manner (see Figure 2): A list of all the data-sets and their sizes was made by the flight software developer, IBM. This list was provided to a person at the National Aeronautics and Space Administration (NASA) who was familiar with the Spacelab flight operations and software. Using his best judgement, this person ordered this list so that the data-sets called close together in time during the Spacelab flight would appear close together in the list. The reordered list was then returned to IBM. IBM then used the reordered list as an input to an algorithm which determines the data-sets' tape positions. The algorithm packs the data-sets on the tape as densely as possible when it is loaded before launch. The loaded MMU tape must be integrated into Spacelab and checked to assure it is operational. While the MMU is installed, it is not possible to revise the tape layout. There are no preflight tests of the Spacelab that exercise the MMU as is expected during the flight.

```
 IBM                NASA               IBM

 ----------         ----------         ----------
|List all  |       |Order     |       |Reordered |     MMU
|MMU Data- |-->    |data-set  |-->    |list into |  -->TAPE
|sets      |       |list      |       |algorithm |     LAYOUT
 ----------         ----------         ----------
```

Figure 2. Tape Layout Process

1.2 Problem Definition

The current method of selecting the MMU tape layout does not provide adequate assurance that MMU tape travel will be minimized. This is due to several reasons. First, the data-set utilization has not been systematically examined by the NASA person before he or she orders the data-set list. Next, the NASA person does not have immediate knowledge of the layout algorithm logic. Also, IBM is not knowledgable on Spacelab operations which require MMU accesses. Thus, the list order is an educated guess by NASA.

1.3 Solution Approach

The problem must be approached using the existing layout algorithm so there will be no change to the IBM activities. Thus, improving NASA's method of ordering the data-set list is required. A rigorous method of ordering the input to the layout algorithm will be developed such that the resulting tape layout will minimize tape travel for expected sequences of data-set accesses.

To do this, three questions must be addressed:

1. When are data-sets expected to be used?

2. Will large MMU tape movements occur for a tape layout given the expected data-set utilization? If so, what are the associated data-sets?

3. Can a proposed data-set list be ordered differently to decrease tape travel?

To answer these questions in a rigorous way, the following actions will be performed: First, the utilization of the MMU data-sets will be systematically identified. This utilization will be correlated to Spacelab operations involving data-sets. Next, the tape travel output variable will be modeled using a sequence of data-set accesses as the independent variable and the layout algorithm input list as the decision variable. This model will be used to evaluate the tape travel associated with different orders of the data-set list that is input to IBM's layout algorithm. Measures of performance will be established for comparing tape travel of different tape layouts.

1.4 Summary

Spacelab I information will be used in this study. This first Spacelab flight was a multi-discipline mission involving over forty experiments in solar physics, life sciences, space plasma physics, earth observations, astronomy, and material science studies. To support this large number and variety of experiments, Spacelab I had 182 data-sets of different sizes to be positioned in 8192 possible tape locations. This indicates the large number of potential list orders. The Spacelab I MMU layout was determined using the simulation discussed herein. The effectiveness of the solution methods was demonstrated by the flight of Spacelab I and will be discussed.

The experiment operations on Spacelab I are typically sensitive to MMU access time. By studying this problem for Spacelab I the possibility of the loss of science data was decreased. Future Spacelab missions should be able to use the techniques described herein to decrease their MMU tape travel.

CHAPTER 2.

SYSTEM DEFINITION

This chapter will identify and describe the overall system and environment involved with the MMU access problem. A "black box" overview will be presented showing the input, decision variable, process, and output of the system. Each of these components will be discussed in detail.

2.1 System Overview

Figure 3 is a black box overview of the overall system. This figure shows that before launch a list of MMU data-sets is made. This list is input to an algorithm which selects tape positions for the data-sets. The data-set positions can be controlled indirectly by changing the order of the data-sets list. Thus, the order of this list is the system decision variable. Before the Shuttle launch, the data-sets are loaded onto the MMU tape in the positions allocated by the layout algorithm. The MMU tape then becomes a component in the command and data management subsystem (CDMS) on Spacelab.

```
                                      ______________
                                      | ORDERED    |
                                      | LIST OF    |
                                      | DATA-SETS  |
                                      |____________|
                                            |
                                      ______v_______
                                      |  LAYOUT    |
                                      | ALGORITHM  |
                                      |____________|
          BEFORE                      ______v_______
          LAUNCH                      |   MMU      |
                                      |   TAPE     |
                                      |  LAYOUT    |
                                      |____________|
- - - - - - - - - - - - - - - - - - - - - - - | - - - - - - -
          DURING FLIGHT                       |
                                              |
SCHEDULED ---->|                    __________v___
OPERATIONS     |     CDMS       |          | MMU |      MMU
               +--------------->|   CDMS   |-----|---------->
               | OPERATIONS     |________________|   RESPONSE
UNSCHEDULED -->|
OPERATIONS
```

Figure 3. System Overview

The CDMS consists of many components, including the MMU. The CDMS hardware and software support a variety of mission operations during a Spacelab flight. The CDMS also provides a variety of interfaces for the flight and ground crews through which MMU operations may be initiated.

CDMS operations are the input to the overall system which cannot be controlled. CDMS operations can be divided into two groups. The first group is the set of operations scheduled at specific times during the Spacelab flight. The second group is the set of operations which occur on an unscheduled basis, i.e., at times not known before flight. As both groups of CDMS operations occur during a flight, requests to retrieve software from the MMU will be made.

The system output of interest is the delay incurred while retrieving software from the MMU. If an MMU access delay is larger than expected, the operation supported by the software might not be performed correctly or on time. This might lead to a loss of scientific data. This delay varies with the amount of tape travel between successive data-set accesses.

2.2 Command and Data Management Subsystem

The Command and Data Management Subsystem (CDMS) is the hardware and software that provides command and data capabilities to the astronauts and ground control personnel in the operation of Spacelab and its payload of experiments [24]. The primary elements of the CDMS are depicted in figure 4 and described below. These descriptions are provided to assist the reader's understanding of how MMU data is used in the system.

1. Mass Memory Unit (MMU): The Mass Memory Unit is the mass storage device for the CDMS. It is a magnetic tape drive which uses a tape with one control track and eight data tracks [4,14]. As figure 5 shows, the eight

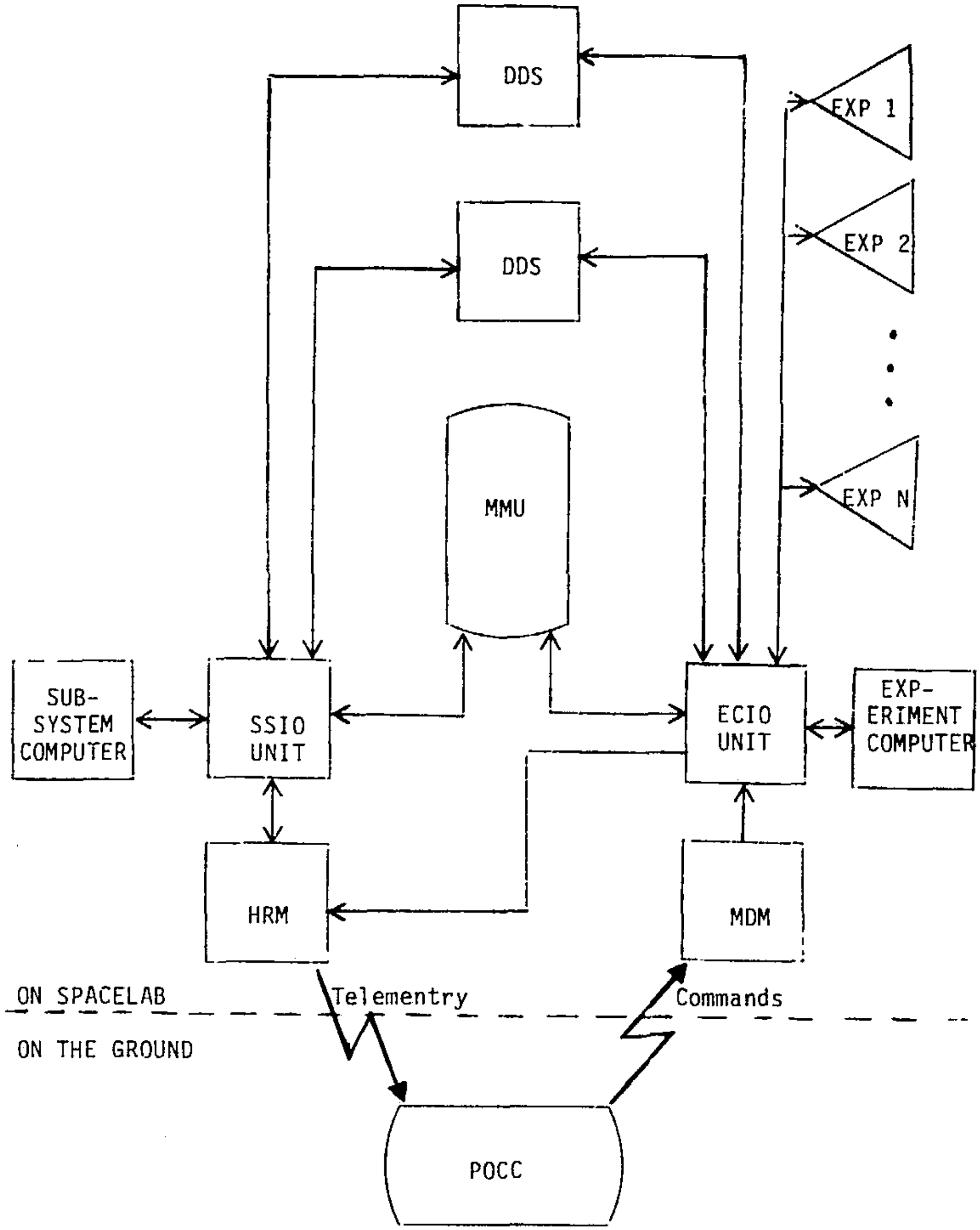

FIGURE 4. CDMS SYSTEM OVERVIEW

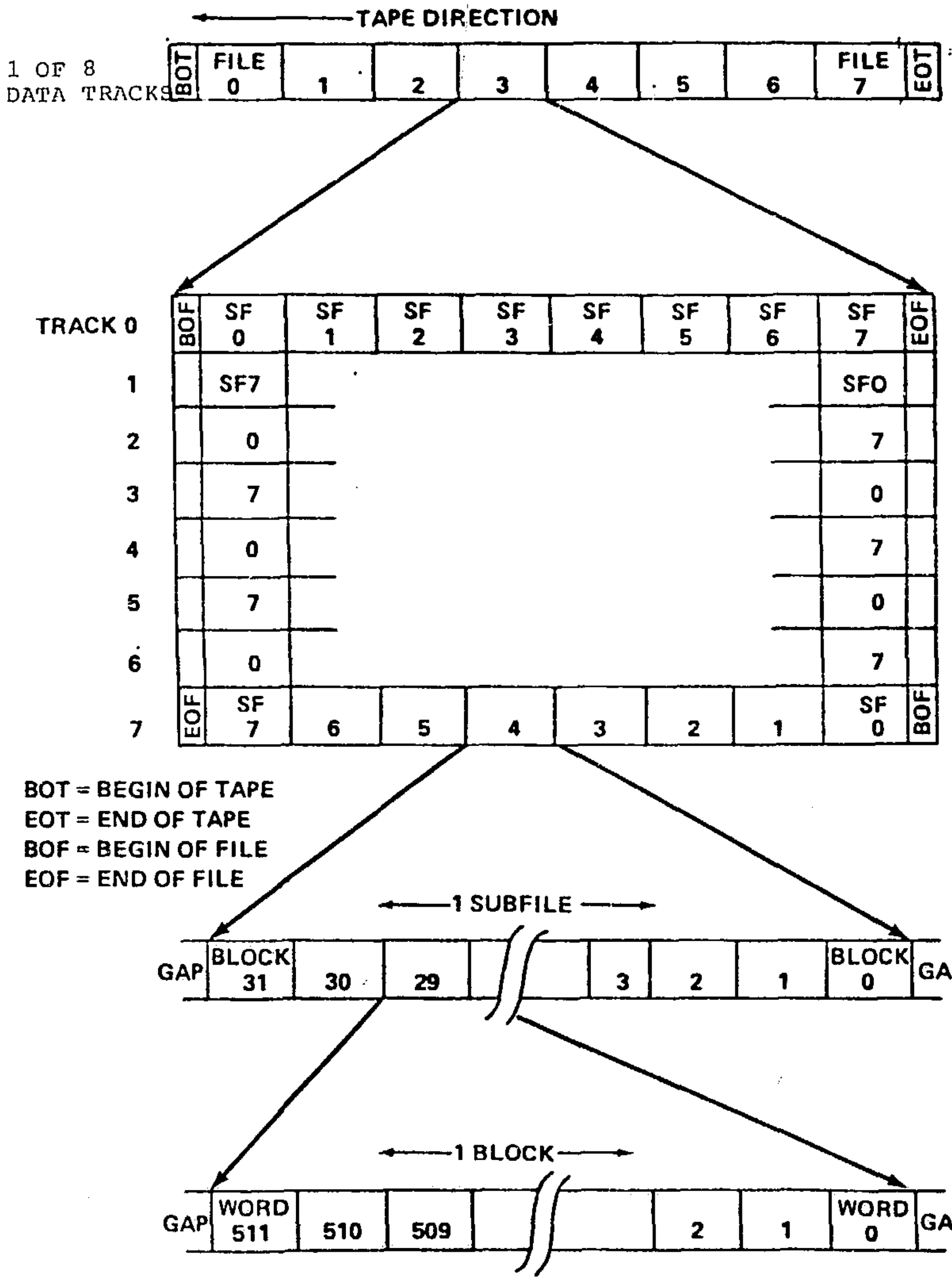

Figure 5. MMU STORAGE AREA CONFIGURATION

parallel data tracks are divided into eight files consisting of eight subfiles each. Each subfile is 32 blocks long and each block contains 512 sixteen bit words. The even tracks (0, 2, 4, and 6) are considered as primary and the odd tracks (1, 3, 5, and 7) as backups. Data on the primary tracks must be duplicated on the backup tracks [4]. Therefore, 8192 different data blocks may be stored on the MMU.

2. Experiment Computer: The experiment computer may be used in the operation of Spacelab experiments. On Spacelab I, the majority of MMU accesses are inititated through this computer. It will recall application programs, crew display formats, data files, files of time-tagged commands, and memory loads for dedicated experiment processors. Also, data may be written to the MMU tape from the experiment computer [4,5].

3. Experiment Computer Input/Output Unit (ECIO Unit): The ECIO unit interfaces the experiment computer to other CDMS components [24].

4. Subsystem Computer: The subsystem computer is used in the operation of Spacelab subsystems. On Spacelab I, the subsystem computer interfaces with the MMU when it is initialized and when commanded from the ground

[20,21]. Initialization of this computer should occur once at the beginning of the mission. The ground will command the subsystem computer to access MMU data approximately 155 times.

5. Subsystem Computer Input/Output Unit (SSIO Unit): The SSIO unit interfaces the subsystem computer to other CDMS components.

6. Data Display System (DDS): The Data Display System is a display and keyboard which provides access to the subsystem and experiment computers by the Spacelab astronauts. As a result of commands through the DDS, MMU data will be accessed [4,5,23].

7. Multiplexer Demultiplexer (MDM): The Multiplexer Demultiplexer provides access to the computers and MMU from the Shuttle orbiter and Payload Operations Control Center (POCC) [4,5,24]. In the POCC, experimenters and engineers can send commands through the MDM when there is a need to change an MMU data-set's contents. On Spacelab I, frequent data-set updates through the MDM are anticipated.

2.3 MMU Data-sets

The MMU tape contains the software data-sets for both the subsystem computer and experiment computer. The subsystem computer software represents only a small portion of the data-sets on the tape for Spacelab I. It includes the subsystem computer executive and support software and the High Rate Multiplexer (HRM) telemetry formats.

Most of the software stored on the Spacelab I tape is used by the experiment computer. It may be divided into two main groups, system and payload. System software consists of the program code and data tables that support the entire Spacelab payload rather than one specific experiment [4,5]. Payload software generally supports one experiment during the time it is scheduled to operate. Five types of software are used to support experiment operations [20,21,23]: application tasks, displays, user data-sets, Experiment Computer Operating System (ECOS) timelines, and dedicated experiment processor loads. Table 1 summarizes the types of experiment computer data-sets used on Spacelab I. The first character of the data-set name identifies its type.

Table 1. Experiment Computer Data-set Types

Type	Description	1st Char.
Task	Executable program code	A or X
Display	Crew display formats	T
UDS	User data files	U
ECOS TL	Time-tagged command sequences	M or S
DEP Load	Dedicated Exp't Processor S/W	D

Application tasks are the first type of software. They are computer programs used to monitor and control experiments. There are two special tasks which support the ECOS timelines described later. These special tasks have names beginning with an "X".

Another type of MMU data-set is a crew display. Display format definitions for the DDS terminals are defined by this type of data-set. Some displays are designed to work within the capabilities of the operating system and are available on the DDS terminal at any time. Other displays depend upon an application task for support. These display data-sets will be accessed when a task is run or an astronaut requests the display while the task is executing.

User data-sets (UDSs) are another type of data-set which is used to store general blocks of data. Through the experiment computer, these data-sets may be read or written by an application task or by ground command. Typically, instrument settings or timing values are stored in UDSs and will need revision based upon the review of experiment telemetry data. Most of these updates will made by using commands sent from the POCC through the MDM.

Another type of data-set stored on the MMU consists of time-tagged command sequences called ECOS timelines. These sequences come in two forms, master timelines (MTLs) and subordinate timelines (STLs). MTLs initiate experiment operations by sending commands directly to the experiment or, more commonly, by initiating a subordinate timeline. STLs typically contain commands dedicated to a specific experiment. The timing of MTL and STL commands is determined by the Spacelab experiment and the operations scheduling. Like UDSs, MTLs and STLs will be revised frequently due to changes in the scheduling of experiments or the need to change experiment settings.

The last type of data-set is called a dedicated experiment processor (DEP) load. Program loads for microprocessors that are provided by the experiment may be stored on the MMU and loaded into those processors via the experiment computer.

Often when one data-set is called, other data-sets must be called to support an operation. For example, when an application task is run, a root segment may call other program segments, a display data-set, user data-set, and/or an ECOS timeline. This sequence of software calls is the same for each performance of an operation.

2.4 Operations Involving MMU Data-sets

The Spacelab CDMS operations are the drivers of MMU data-set accesses. Many operations are scheduled at specific times [22], but unscheduled events may also cause data-set accesses. CDMS operations may be divided into two types, scheduled and unscheduled. Scheduled operations have predefined times of occurrence. Unscheduled operations occur at undefined times and are associated with routine activities or contingencies. The following paragraphs describe the relationships between CDMS operations and the use of MMU data-sets.

The operations of the Spacelab experiments are carefully scheduled to assure that the Spacelab resources will be available to study a scientific phenomenon when it is apparent [22]. Some of these phenomenon are apparent at a precise time and for only a short time. Thus, any delay in experiment performance could preclude the collection of scientific data.

Spacelab operations are scheduled in the following manner. First, the flight time is divided and allocated to different experiment disciplines as shown in Figure 6. The time slices are selected such that the conditions are conducive to support the experiment operations. At these times, the Shuttle and Spacelab will be operated according to the experiment needs. As examples, the Shuttle payload bay will be pointed to the sun for solar physics experiments, to the earth for earth observation experiments, and to various galaxies and stars for astronomy experiments. The solar and earth observation experiments will be scheduled during the sunlit portions of the orbits while the astronomy experiments will typically be scheduled in darkness. During these time slices, experiments sharing an interest in the scientific phenomenon may be operated in parallel and sequentially.

Before a Spacelab flight, individual operations are identified and encoded into a mission planning computer system [22]. In this ground computer, a data-base is defined wherein each scheduled function is given a label and broken into numbered steps. Each step represents an operation in the performance of a functional objective. The mission planning system software creates a file scheduling

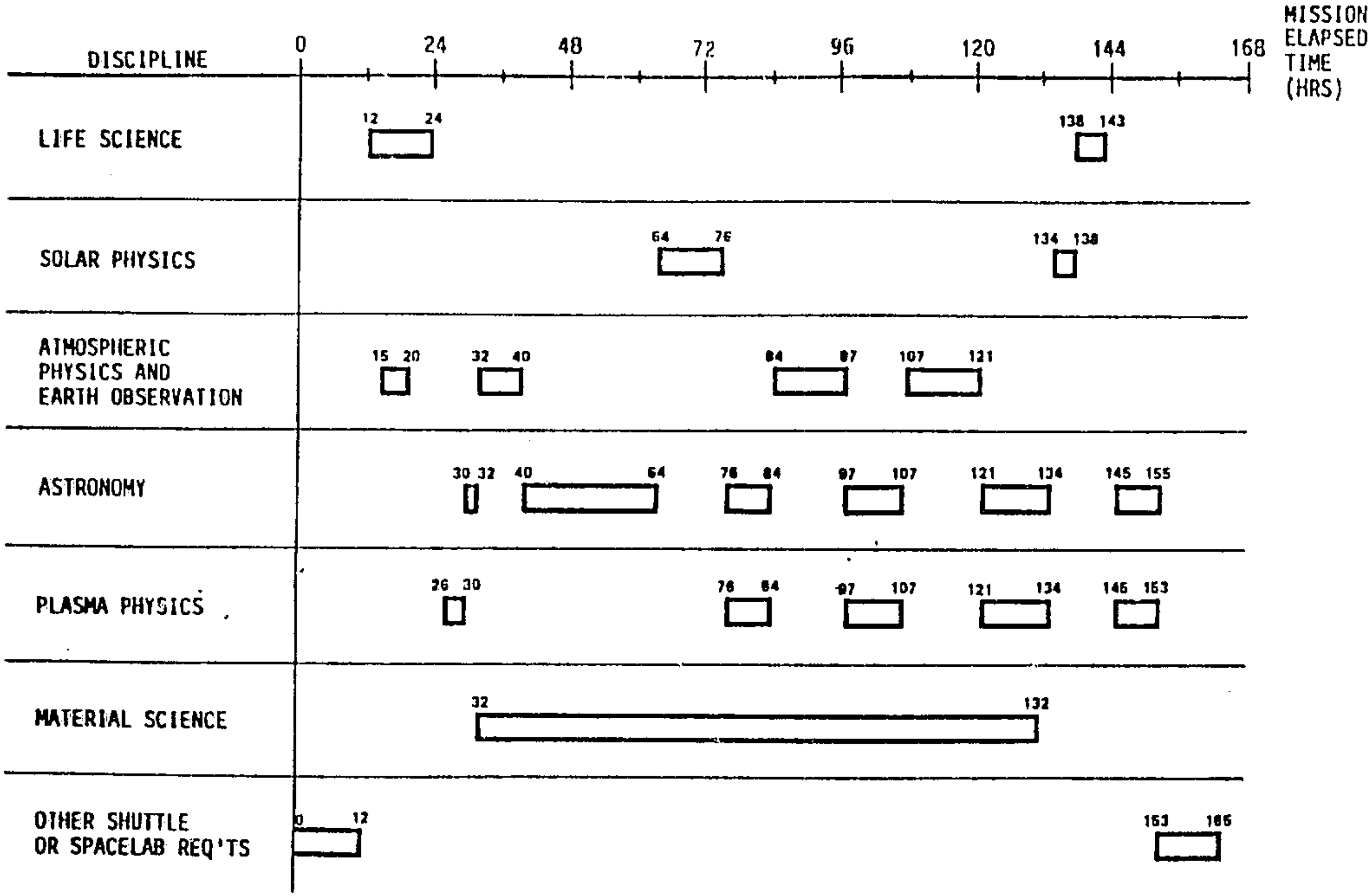

FIGURE 6. SCHEDULING OF EXPERIMENT DISCIPLINES FOR SPACELAB I

these operations at specific times.

Table II shows a segment of the Spacelab I mission schedule stored in the mission planning system. From left to right, the table identifies which astronauts are to support the operation step, the operation label and step number, a description of the step, and the scheduled start and completion times for the operation. A scheduled operation may or may not require that software be accessed on the MMU according to the operation's procedures. This table shows that twenty two operations are scheduled between hours 38 and 39. For this hour, thirteen of these operations will involve data-set accesses.

Some of the software stored on the MMU tape supports operations which do not appear in the mission schedule. Routine unscheduled functions and some contingencies may involve data-set accesses. These accesses will occur during a flight with some estimated frequency distribution. ECOS timeline maintenance, user data-set maintenance, and crew display calls are examples of unscheduled operations.

2.5 Data-set Positioning on the Tape

The position of each data-set on the MMU tape is selected by an MMU tape allocation program normally run by the Spacelab software integration contractor. This program reads down a list of the MMU data-sets and allocates tape

ORIGINAL PAGE IS
OF POOR QUALITY

TABLE 2.

COMPOSITE TABULAR TIMELINE

EXPERIMENTERS	NAME - STEP	STEP DESCRIPTION	(MET) START TIME HRS:MIN:SEC	END TIME HRS:MIN:SEC
	013-F02	4 CALIBRATION (13/F870)	37: 37:20	37: 38:20
	014-F01	1 LOAD AND COUNT STL (14/F200)	37: 37:40	37: 41:40
	013-F02	5 LAUNCH DELAY OPS WINDOW	37: 38:20	37: 41:20
	MIF07B3W	12 DEP CMD ENTRY	37: 41:20	37: 42:20
	014-F01	2 LIMIT CHECK TEMP., SWITCH ON MAIN POWER (14/F200)	37: 41:40	37: 51:40
	MIF07B3W	13 MTL SHUTDOWN (FO-17)	37: 42:20	37: 54:20
	013-F02	6 ABSORPTION (13/F880)	37: 43:20	37: 47:20
PS-1 *MS-1	020-F01A	9 REMOVE HATCH, STOW E3020 (20/3006, 20/F410)	37: 44:00	38: 2:00
	013-F02	7 ABSORPTION (13/F880)	37: 47:20	37: 51:00
	013-F02	8 CALIBRATION (13/F870)	37: 51:00	37: 52:00
	014-F01	3 OPERATE E3014 (14/F200)	37: 51:40	38: 24:00
	013-F02	9 SHUT-DOWN (13/F900)	37: 52:00	37: 53:00
MS-2	SLEEP M2	2 SLEEP PERIOD FOR MS-2	38: 0:00	45: 30:00
OC-2	SLEEP O2	2 SLEEP PERIOD FOR OC-2	38: 0:00	46: 0:00
PS-2	SLEEP P2	2 SLEEP PERIOD FOR PS-2	38: 0:00	45: 30:00
PS-1 *MS-1	022-F1/3	1 INSTALL E3022 (22/F110)	38: 2:00	38: 20:00
	316-F01	5 COOL E3316	38: 20:00	39: 50:00
MS-1	022-F1/3	2 RUN ECAS A22A AND CHECKOUT E3022 (22/3002,22/F120)	38: 20:00	38: 33:00
	017-F01B	1 LOAD AND COUNT STL (17/F200)	38: 20:00	38: 22:00
	017-F01B	2 RUN ECAS A17A AND PASS CYCLE NR. (17/F200)	38: 22:00	38: 27:00
	014-F01	4 SWITCH OFF MAIN POWER (14/F200)	38: 24:00	38: 26:00
	017-F01B	3 OPERATE IN LIMB MODE (17/F200)	38: 27:00	39: 17:20
	TGA	1 WARMUP TGA	38: 30:00	38: 57:00
	033-F03	1 RUN ECAS A33A	38: 33:00	38: 34:00
MS-1	022-F1/3	3 INSTALL HATCH, VENT SAL (22/3003)	38: 33:20	38: 52:00
	033-F03	2 STAND BY	38: 34:00	38: 37:00
	033-F03	3 OPERATE	38: 37:00	38: 49:40
MS-1	330-F01	2 STUDY FDF AND SET UP 330	38: 52:00	39: 2:00
	022-F1/3	4 E3022 IN STANDBY	38: 52:00	57: 35:00
	MIF07B3	1 LAUNCH WINDOW VARIATION	38: 52:00	38: 57:00
	MIF07B3	2 MTL TURNON SEQUENCE (FO-16)	38: 57:00	38: 58:00
	TGA	2 CO ANALYSIS TGA	38: 57:00	39: 39:00
	MIF07B3	3 TURNON SEQUENCE CONT'T (FO-16)	38: 58:00	39: 12:00
	013-F2/3	1 LOAD AND COUNT STL	38: 59:00	39: 1:00
	013-F2/3	2 RUN ECAS A13A (13/F860)	39: 1:00	39: 4:00
MS-1	330-F01	3 OPERATE ANNULAR CYLINDER	39: 2:00	39: 9:00
	013-F2/3	3 SET-UP (13/F860)	39: 4:00	39: 7:00
	013-F2/3	4 CALIBRATION (13/F870)	39: 7:00	39: 8:00
	014-F01	1 LOAD AND COUNT STL (14/F200)	39: 7:00	39: 11:00
	013-F2/3	5 LAUNCH DELAY OPS WINDOW	39: 8:00	39: 13:00
MS-1	330-F01	4 CHANGE PROBE	39: 9:00	39: 11:00
MS-1	330-F01	5 OPERATE RIGHT CIRCULAR CYLINDER	39: 11:00	39: 18:00
	014-F01	2 LIMIT CHECK TEMP., SWITCH ON MAIN POWER (14/F200)	39: 11:00	39: 21:00
	MIF07B3	4 DEP CMD ENTRY	39: 12:00	39: 13:00
	MIF07B3	5 TERMINATOR - LOW ALT. (FO-7B)	39: 13:00	39: 20:40
	013-F2/3	6 ABSORPTION (13/F880)	39: 13:00	39: 17:00
	013-F2/3	7 ABSORPTION (13/F880)	39: 17:00	39: 20:40
MS-1	330-F01	6 CHANGE PROBE AND FILM CASSETTES	39: 18:00	39: 26:00
PS-1	101-F01	5 MS101 DAY 2 OPERATIONS WITH PAYLOAD CREW	39: 18:00	39: 22:00
	013-F2/3	8 CALIBRATION (13/F870)	39: 20:40	39: 21:40
	MIF07B3	6 DEP CMD ENTRY	39: 20:40	39: 21:40

positions for each. Table 3 shows a portion of this list which identifies the data-set name, description, size, initial position (if fixed), and whether its position is fixed or variable. By changing the order of the data-set input list, different MMU tape layouts result.

Table 3. A Partial List of Data-sets

Name	Description	Size (1)	Pos.(2) T	F	S	B	Control
SBOOTP	SCOS BOOTSTRAP PRIME	2	0	0	0	0	FIXED
SBOOTR	SCOS BOOTSTRAP REDUNDANT	2	0	7	7	0	FIXED
SCOSAM	SCOS IPL AMI	126	0	0	0	2	FIXED
SSCDIR	SCOS MMU DIRECTORY	1	0	1	0	0	FIXED
S1S	ECOS	39	0	0	0	0	VAR
S1C	ECOS	138	0	0	0	0	VAR
A13D0	ECOS	3	0	0	0	0	VAR
T13D	ECOS	2	0	0	0	0	VAR
A13E0	ECOS	4	0	0	0	0	VAR
T13E	ECOS	2	0	0	0	0	VAR

(1) Number of 512 word blocks.
(2) T = Track; F = File; S = Subfile; B = Block

In the algorithm, rules on the positioning of data-sets are as follows:

1. Interface agreements and system requirements constrain some software to occupy specific tape positions. These data-sets are said to be fixed on the MMU tape. Tape must be allocated to fixed data-sets first. The remainder of the data-sets will be positioned in any tape blocks not already allocated. The first four

entries in Table 3 are examples of "fixed" data-sets.

2. Odd numbered tracks must contain redundant versions of the even numbered tracks. For example, the data on track 1 is identical to track 0.

3. Data-sets which are larger than a file (256 blocks) must be positioned at the start of a file.

4. Data-sets larger than a subfile but smaller than a file must be positioned at the start of a subfile.

5. Data-sets smaller than a subfile must be contained completely within a subfile. That is, a data-set less than 32 blocks long may not cross subfile boundaries.

6. Experiment computer displays may not be positioned in subfiles 0, 1, 6, or 7 of any file.

7. Data-sets should be positioned such that tape movement across the boundary of files 3 and 4 is minimized.

The allocation algorithm sequentially reads each data-set in the order listed and allocates tape positions for each data-set within the above constraints. The search for unallocated tape is started at file 6, subfile 0, block 0, track 0. As soon as a valid position is found for a data-set, the algorithm permanently allocates that space to it. The program searches the files in the order 6, 5, 7, 4, 3, 2, 1, 0. If the file number is greater than 5, the subfile search order is 0 to 7 and the block order is 0 to

31. Otherwise, the order is 7 to 0 and 31 to 0, respectively. The even tracks are always searched in the order 0, 2, 4, 6.

2.6 MMU Response to Data-set Accesses

The MMU response to data-set accesses is the system output variable of interest. The way that the MMU performs data-set accesses determines the access times and is described below [4,14].

When a data-set is read, the tape moves from its current position to the closest block of the desired data-set and then reads the MMU tape blocks to the end of the data-set. As an example, suppose the data-sets named A34R0 and T14A are positioned on the tape as shown in Table 4.

Table 4. Example Data-set Tape Positions

Data-set	Size	Track	File	Subfile	Blocks
A34R0	12	6	6	1	15-26
T14A	1	2	5	5	30

If data-set A34R0 is read from track 6 in the forward direction, the tape will stop moving positioned at file 6, subfile 1, block 28. To read another data-set, T14A, the tape will skip 124 blocks while moving to file 5, subfile 5, block 30 to read from track 3 in the reverse direction.

When a data-set is accessed, the startup and stop time of the tape movement is about 550 milliseconds. Approximately 26 milliseconds are required to read a block. To skip an entire subfile, 820 milliseconds are required [14].

Only one factor can be controlled that affects data-set access times. It is the tape positions of the data-sets. Thus, to minimize the access time, two data-sets that are often sequentially accessed should be positioned close together on the tape for minimal tape travel.

2.7 Summary of System Characteristics

The overall system consists of the MMU tape layout process, the operations inputs, the CDMS system (including the MMU tape) and the MMU tape movement. The system input is the scheduled and unscheduled operations that require data to be retrieved from the MMU. The CDMS processes these operations. It consists of many interfacing components including the MMU. The tape positions of software data-sets are determined before flight. The data-set positions on the tape can be controlled by changing their input order to the MMU tape layout algorithm. The overall system output of interest is the response time of the MMU. The MMU access time will vary with the tape travel between data-set accesses. By minimizing tape travel, the probability of large access times which adversely impact science data collection will be decreased during Spacelab flights.

CHAPTER 3.

SELECTION OF A SOLUTION APPROACH

In this chapter, the problem will be generalized and classified to identify a solution approach. The relationship between the independent, decision, and output variables will be examined. Potential solution methods will be assessed that will provide an acceptable solution in reasonable time.

3.1 General Problem Statement

To access a data-set, the MMU tape must be moved to the data-set. If the tape should happen to be positioned at that data-set already, then no time delay will occur due to repositioning of the tape. Thus, if two data-sets are accessed together, they should be positioned together on the tape. Often within an operation, several data-sets are always accessed sequentially. Since these accesses always occur together, the distance between them on the tape should be minimized. In a similar way, several operations will always occur together. If these operations have data-set accesses, the data-sets associated with this group of operations should be positioned closely together. The objective will be to find a tape layout that minimizes tape travel for all data-set accesses within and between operations.

Tape travel can be examined in several ways according to the operations the data-set accesses are associated with. As an example, measurements could be made on tape travel for each data-set access within an operation. The tape travel within an operation would indicate the operation's delay times due to data-set accesses. Also, the tape travel between operations could be measured. This travel would be indicative of the wait time to start a new operation.

Any two arbitrarily selected data-sets could be positioned together on the MMU so that there could be no travel between them. But on Spacelab I, there are 166 data-sets to be positioned that will support one or more of over 1000 operations. The objective is to find a "balanced" tape layout that minimizes tape travel for all the operations' data-set accesses expected during a flight. To evaluate whether a layout is "balanced," averages of the maximum travel within an operation and between operations can be calculated. Various tape layouts can be compared using these averages so the best layout can be chosen for use.

For Spacelab I, let us identify tape travel measures MTI and ATB where, over the flight, MTI is the average of the maximum travel for a data-set access within any operation, and ATB is the average travel between operations. These will be the system outputs of interest [10,11,19,27]. Let us represent them collectively by the set O where

$$O = \{MTI, ATB\}.$$

If MTI is minimized, the average time to complete a CDMS operation will be minimized. If ATB is decreased, the travel between operations is decreased which decreases the average delay incurred at the start of an operation.

The independent variable is the time of occurrence of CDMS operations that require data-set accesses. Many of these operations are scheduled for a Spacelab flight and are to be performed at their scheduled times. The times of these operations may be considered deterministic if no deviations from the operations schedule are assumed. Thus, if we were given only these operations, the sequence of data-set accesses would be completely known. Let us represent these scheduled operations by the time ordered set [12]

$$W = \{w(i)\} \quad i = 1 \text{ to } m$$

where

$w(i)$ is the ith of m scheduled operations.

Note that an operation, let it be called s, may be scheduled more than once so

$$s = w(i) = w(j) \text{ is possible for } 1 \leq i,j \leq m.$$

For Spacelab I, m is about 400.

As noted in Chapter 2, a significant portion of CDMS operations can be expected to occur on an unscheduled basis, i.e., at random points in time, interleaved with the scheduled operations. Let the unscheduled operations be represented by the time ordered set

$$V = \{v(i)\} \quad i = 1 \text{ to } n$$

where

$v(i)$ is the ith of n unscheduled operations.

Note that an unscheduled operation, let it be called u, may be occur more than once so

$$u = v(i) = v(j) \text{ is possible for } 1 \leq i,j \leq n.$$

The number of unscheduled operations, n, for Spacelab I is estimated to be over 600.

Now, the total set of CDMS operations can be given by the time ordered set, I, where

$$I = W \cup V$$

which contains m + n total operations. Because the times that unscheduled operations occur are random, the sequence of all operations (i.e, the order of I) is stochastic.

The decision variable is the order of the data-set list input to the layout algorithm. The list can be represented as an ordered set

$$DV = \{d(i)\}$$

where d(i) is data-set i of K data-sets having variable tape positions. The order of the list of data-sets is the decision variable because different list orders determine different tape layouts. Each d(i) has two attributes, a(i), the data-set type; and b(i), the data-set size, that affect where a data-set may be positioned by the tape layout algorithm. Since the algorithm assigns tape positions for the data-sets in the order the data-sets are listed (i.e., the order of DV), the sizes and types of all the data-sets listed before a specific data-set can affect where that data-set will be positioned. So, the position of any one data-set can be given by

$$P[d(i)] = g(DV)$$

where the function, g, is not a mathematical relationship but the set of rules and contraints incorporated into the tape layout computer program.

Now, the tape travel represented by the set of outputs, O, will be a function of the sequence of inputs, I, and the order of the data-set list, DV. That is,

$$O = f(I,DV).$$

The general problem is to find a DV related to I that achieves an objective function of the set of outputs, O.

3.2 Solution Approach Assessment

The general problem statement gives an insight into the problem type and the possible techniques available to solve it. As noted before, the positioning of the data-sets is determined by an algorithm that does not have a typical mathematical representation. Also, the tape travel is related to a discrete ordering of the data-sets and a stochastic, time-related sequence of operations. The problem cannot easily be formulated as a linear programming (e.g., the transportation or assignment problems) or queueing theory problem [10] so that an analytical solution could be attempted. This is because there are a large number of variables and constraints that can readily change and/or are not immediately apparent. Also, the system had not ever been observed in operation before the first Spacelab flight. Thus, the model must be able to predict the system response and allow the system to be investigated under varying conditions.

Methods are needed to (1) predict MMU tape travel for a candidate data-set list, and (2) determine a good order for the list that minimizes tape travel.

There are two ways to reliably predict the tape travel before a Spacelab flight. One would involve using the flight hardware and operating it on the ground as it is expected to be operated during flight and measuring the actual tape motion. There would be many programmatic and logistical problems involved with experimenting with the flight systems. Also, the turnaround time would be excessive for changing tape layouts. The alternative is to simulate the tape motion with a computer model of the tape layout and CDMS operations. Because using the flight hardware is not a viable solution, a simulation will be required.

A technique must be selected to solve problem (2), i.e., to find a good input order, DV. One possible technique would be to use exhaustive enumeration [27] wherein all possible list orders were evaluated. This would not be feasible for Spacelab I which has 166 data-sets that could be listed in 166 factorial different orders. Therefore, a good ordered set, DV, will need to be found in a limited subset of all possible DV.

The number of DV evaluations must be restricted in a reasonable way. If the layout algorithm is examined, the general tape position of a data-set, d(i), in DV can be predicted according to the position of d(i) in DV. For example, data-sets listed near the top of the list will be positioned close to file 6, subfile 0. Also, it is reasonable to assume that any two consecutive data-sets in DV will be assigned positions close together on the tape. But due to the constraints in the algorithm associated with data-set size and type, these intuitive predictions are not very accurate. This can be illustrated by interchanging the list positions of two data-sets in the list. Potentially, the positions of all of the data-sets may change depending on the sizes, types, and list positions of the two data-sets selected.

With this inaccurate but usable predictability, one can reasonably assume that by starting with a DV ordered according to some prior knowledge of data-set utilization, a tape layout will result that is better than one using an arbitrarily ordered DV. To determine how to best order the list of data-sets, the simulation can be used to determine the sensitivity of tape travel to different methods of ordering. By using the sensitivity knowledge, the number of possible orderings can be constrained reasonably. Then, experiments can be performed with different but contrained orderings to determine one that provides good tape travel measures compared with orders selected without the sensitivity information.

Each different data-set list order will need to be compared with other list orders. Also, the potential for further tape travel improvement needs to be examined. This can be done by comparing the outputs, O, for a layout with other layouts evaluated. If after some number of list order evaluations there is no tape travel improvement, then an "optimal" (quotes indicating that this is not a rigorous optimization) layout has been found.

3.3 Solution Method Selection

Simulation techniques are diverse and may be applied to a broad range of problems [1,3,6,11, 16,17,19,27]. Computer simulation is good technique for evaluating MMU tape motion. Shannon [19] cites advantages for using simulation that apply to this problem. As noted previously, use of the flight hardware has several disadvantages, such as disrupting the preparation for flight and excessive operational cost which does not allow much opportunity for experimentation. Also, simulation allows the MMU utilization over a typical seven-day flight to be compressed into minutes. Finally, another advantage of computer simulation is the accessibility of existing computer data and routines. The simulation can be designed to access the mission schedule data-base directly. By using it directly, data translation errors cannot occur.

3.4 Simulation Method Selection

Given the above reasons for using simulation, the next question is one of simulation method. Simulations may be written in general purpose or special purpose languages [19]. General purpose languages like FORTRAN [19,25,26] or PL/I may be used in a wide range of applications besides simulation. Special purpose languages [1,3,11,16,17], such as GPSS, SIMSCRIPT, SIMULA, or SLAM , have been developed specifically for simulation work. There are advantages and disadvantages for each language category [19]. For this problem, FORTRAN was selected because it is the most familiar and accessible language, the problem does not require any complex random variate routines, and many

FORTRAN routines already exist that may be used in the simulation.

3.5 Summary

To find a good tape layout, the following approach will be taken. First, a computer simulation of the MMU tape and its motion in response to CDMS operations will be developed. Then for the Spacelab I data-set list, an investigation will be performed on how tape travel varies with revisions to the order of data-set list. The simulation will then be used to experiment with different list orders, but the variations in the ordering will be constrained using the results of the investigation. This experiment with different list orders will continue until the ability to improve the tape travel is difficult to achieve. The list that yields the best simulated tape travel will be selected for use.

CHAPTER 4.

SIMULATION MODEL

This chapter describes the computer simulation model of the MMU tape and its motion in response to CDMS operations. The simulation creates a tape layout with the layout algorithm using a candidate data-sets list. The simulation also uses inputs that define the sequences of CDMS operations and the data-sets accessed during these operations. MMU tape travel within and between these CDMS operations is measured for analysis. The tape motion is summarized by averaging the tape movements and identifying the ten largest tape movements. Figure 7 shows an overview of the simulation. A complete listing of the FORTRAN 77 [25,26] program is provided in Appendix A.

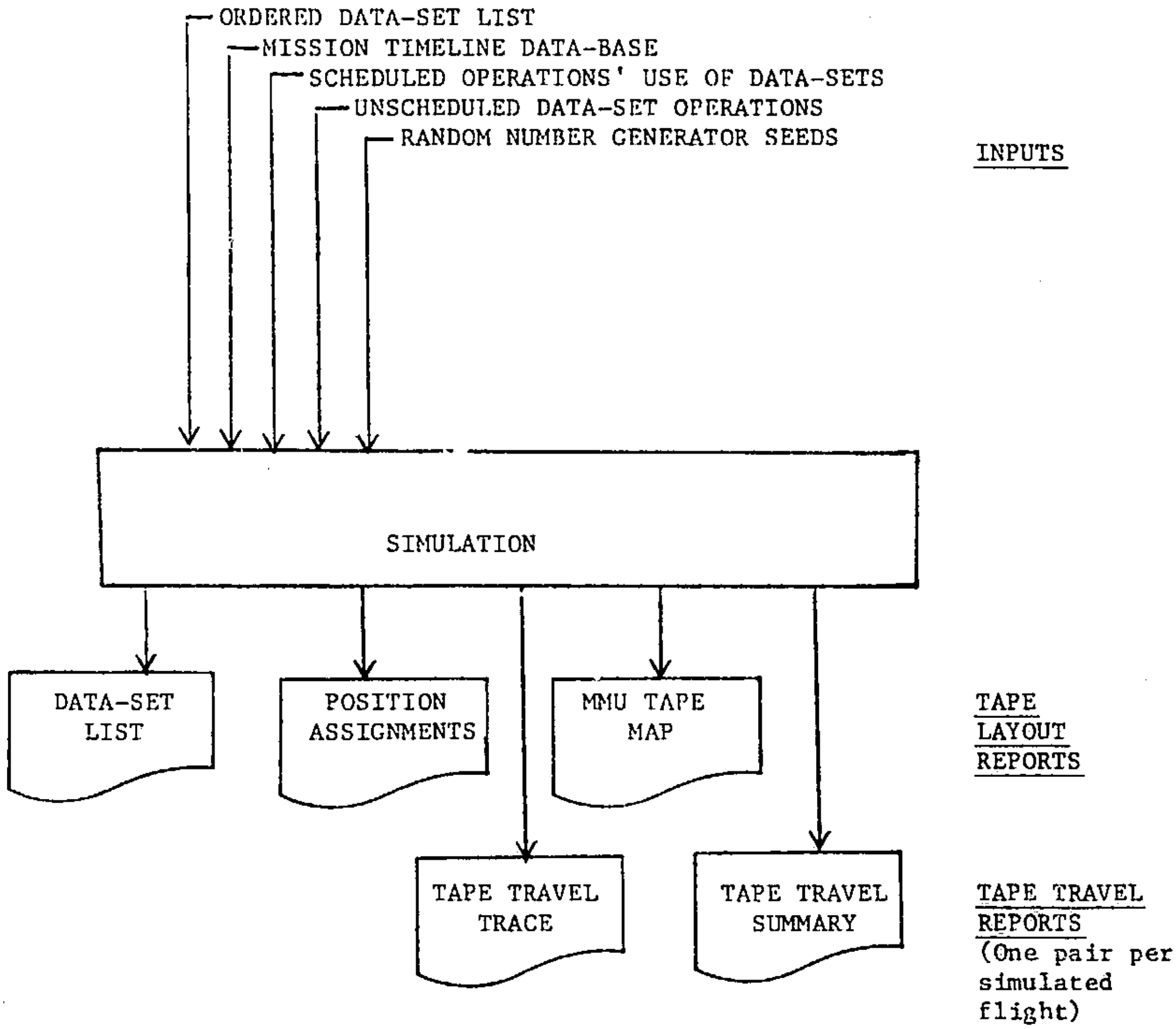

FIGURE 7. SIMULATION OVERVIEW

4.1 Input Data

MMU data-set accesses are associated with scheduled and unscheduled CDMS operations that occur during a Spacelab flight. The tape travel between data-set accesses is a function of the sequencing of the operations, the sequencing of the data-set acccesses within each operation, and the data-set positions on the tape. The following sections describe and discuss the simulation inputs.

4.1.1 Scheduled Operations

The scheduled CDMS operations that require data-set accesses are a subset of the mission timeline data-base of all scheduled operations. This data-base was used to define the sequence of scheduled CDMS operations by identifying the pertinent subset of these operations that have data-set accesses. In addition, the sequence of data-set accesses during these operations had to be identified since this is not a part of the mission timeline data-base.

Persons familiar with the flight operations and software can most readily identify the operations that have data-set accesses and the sequence of data-set accesses. Spacelab I experts were able to efficiently identify the data-sets to be accessed during the flight with a minimal review of the operations procedures and software design.

Someone with little understanding of the flight software operations can be expected to have a more difficult time identifying the data. For the Spacelab I problem, the following procedure was performed to define the data-set accesses associated with scheduled operations:

1. A printout of all the scheduled operations in the mission timeline data-base was received from the mission planners.

2. The procedures for each operation were reviewed with the flight operations experts to determine if any data-set accesses are to occur during the operation.

3. If data-set accesses are expected, the sequence of data-set accesses within the operation was determined from the the operations procedures and the design of the software involved and verified by the operations experts.

Examples of the data defined with these steps are shown in Table 5.

Table 5. Example Data-Set Access Sequences for Scheduled Operations

Scheduled Operation, Step	1st Data-set Accessed	2nd Data-set Accessed	3rd Data-set Accessed	. . . . .	. . . . .
N1F97A8,2	S1S	S1C			
N2C/OF01,3	S02	A02A0	A02A01	T02A	T02G
013-F01,5	S13	A13A0	A13A01	T13A	U13APC

This example shows that during scheduled operation N1F97A8, step 2, data-set S1S will be accessed first and then S1C. The simulation allows up to fifteen data-sets to be correlated to a given step. (Though it is possible for more than fifteen accesses to occur during a given operation, the maximum number for Spacelab I was ten.) The complete set of this data for Spacelab I is given in Appendix B.3.

4.1.2 Unscheduled Operations

Many data-set accesses are associated with operations that are not scheduled in the mission timeline. Again, the Spacelab I unscheduled operations are best determined by Spacelab operations and flight software experts. It would be desirable to determine this input based on the experience of previous Spacelab flights, as well, but this was not possible for Spacelab I since it was the first Spacelab flight. The following procedure was used to define data relevant to these operations:

1. The list of data-sets was reviewed to identify the ones that support unscheduled operations. (Note that a data-set that supports a scheduled operation may also be accessed on an unscheduled basis.)

2. A set of unscheduled operations were defined where each operation was represented by a sequence of data-set accesses. The sequence was determined by the flight software design and procedures for the unscheduled operations.

3. For each unscheduled operation, a frequency and earliest and latest times of occurrence were estimated.

Examples of this data are given in Table 6.

Table 6. Example Input for Unscheduled Operations

Frequency	Start time (hrs)	End time (hrs)	1st Data-set Accessed	2nd Data-set Accessed	3rd Data-set Accessed	. . . .
155	4.0	159.0	TMEM			
44	4.0	159.0	MMA			
7	4.0	159.0	XTLMO	XTLM01	XTLM02	TTLM
14	4.0	159.0	XTMNO	TTMN		
28	4.0	159.0	AOFDO	TOFD	U05PMU	
14	24.0	144.0	S1S			
14	24.0	144.0	S1C			

This example shows seven different unscheduled operations. The first one represents the flight crew's request for the experiment computer memory management display, data-set TMEM. It is expected to be accessed 155 times after 4.0 and before 159.0 hours into the flight. The second unscheduled operation represents the use and revision of the master timeline (MTL), data-set MMA. The simulation allows up to fifteen accesses to define a single unscheduled operation. (Again, it is possible to have more, but the maximum encountered on Spacelab I was four.) The complete set for Spacelab I is given in Appendix B.4.

4.1.3 MMU Data-sets List

To examine MMU tape travel, the data-set tape positions are needed. The data-set positions are determined by the tape layout algorithm as a function of the order of the data-sets list. This ordered list is the decision variable, DV, of Chapter 3. The list identifies all of the data-sets that will be loaded on the MMU tape according to the mission's flight software requirements [20,21]. Each data-set's size and whether it has a fixed or variable position are specified. A complete Spacelab I list of data-sets as it is input to the simulation is shown in Appendix B.1.

4.2 Simulation Process

The simulation processing of the inputs was broken into several phases to be detailed in this section. In the first phase, the MMU tape layout was determined. In the next phase, a set of unscheduled operations was defined in time order. In the third phase, these unscheduled operations and the scheduled operations were processed in time order to determine the data-set accesses. The tape travel was computed for the data-set accesses associated with the operations. Finally, when every operation had been examined, the tape motion was summarized.

4.2.1 Tape Layout Process

The positions of the MMU data-sets are determined by the tape layout algorithm using the input of the data-sets list. The layout algorithm models the MMU tape as a large array in which each element of the array represents an MMU block. The algorithm sequentially reads the data-set list and assigns the data-sets to unused array elements. As the block assignments are made, each data-set's starting position on the tape is stored. Once the assignments are made for all of the data-sets in the list, the stored starting positions can be used to model the MMU tape in the simulation.

A listing of IBM's PL/I program that lays out the MMU was received and translated to FORTRAN 77 for the simulation. The translated program uses the same data-set list and generates the key reports of the PL/I program. (See appendices B.1 and C.1, C.2, and C.3 for examples.) To assure no discrepencies arose in translation, a test using identical inputs assured the reports from both versions were the same.

4.2.2 Determination of Unscheduled Operations

A sequence of unscheduled operations is defined using the unscheduled operations input data. For each one of these operations, a set of times is selected at random between the earliest and latest times. The number of times selected is equal to the expected frequency of occurrence specified. Random times are selected for all of the unscheduled operations and integrated into a single, time-ordered file. This file then defines the sequence of all unscheduled operations and the sequence of data-sets accessed during each operation.

4.2.3 Determination of Scheduled CDMS Operations

To determine the sequence of scheduled CDMS operations, each scheduled operation in the mission timeline data-base is examined and checked against the list of operations that have data-set accesses. If the operation is in the list, then the sequence of data-sets defined in the list can be used to determine the sequence of MMU accesses. This, in turn, defines the positions the tape must travel to at the start of the operation, within the operation, and after the operation is completed.

4.2.4 Tape Travel Computation

After a time-ordered set of scheduled operations and a time-ordered set of unscheduled operations have been defined, the tape travel for data-set accesses was determined as follows:

The simulation starts assuming the tape is positioned at file 6, subfile 0, block 0. (Note that the track number does not affect tape travel since tracks are side-by-side on the tape.) This position was assumed because it will be close to the data-sets (i.e., it is the first tape position assigned). This avoided a large tape movement that could have masked the identification of a large movement during the simulation.

The first simulated CDMS operation to occur in flight, scheduled or unscheduled, is then selected from the defined sequences. As described in sections 4.1.1 and 4.1.2, each operation will have one to fifteen data-set accesses identified in order. The simulation computes the number of blocks skipped to reposition the tape from its current position to the closest block of the first data-set accessed during the operation. (The closest block may be used since data-sets may be accessed in either direction.) Let us call this number NSK(1). Then, the tape is repositioned to the next block past the end of the data-set as it would be after the access is completed.

Figure 8 depicts a portion of the MMU tape to show an example of how the travel is computed. If the first data-set to be accessed is in file 6, subfile 1, blocks 0 through 4, NSK(1) would be 32 since there are 32 blocks in subfile 0 to be skipped to reach the data-set. In the figure this represents a move from position A to position B. The data-set access is simulated by the move over blocks 0 - 4 in subfile 1 to a position at the start of block 5, i.e., from position B to position C.

```
                                   File 6
             Subfile 0                           Subfile 1
Block No.=|0 1 2 3..................31|0 1 2 3 4 5 6......
          |-+-+-+-+-----------------+--|-+-+-+-+-+-+-------
          | | | | |                 |  | | | | | | |
          |-+-+-+-+-----------------+--|-+-+-+-+-+-+-------
           <----32 blocks skipped----> <Access-->
Position  A                           B         C
```

Figure 8. Example Tape Positions

If another data-set is to be accessed within the operation, the simulation computes the number of MMU blocks skipped to get to the next data-set. Let this number be KSKIPI(1). If k data-sets (k > 1) are accessed within the operation, KSKIPI(2), KSKIPI(3), through KSKIPI(k-1) are computed. The simulation will keep track of the largest value of KSKIPI computed within an operation. Let this number be KSK(1) where

KSK(1) = max {KSKIPI(1), KSKIPI(2), ..., KSKIPI(k-1)}.

By summing these values of KSKIPI, the total number of blocks skipped within this operation can be computed. Let this number, JSK(1), be given by the following equation:

JSK(1) = KSKIPI(1) + KSKIPI(2) + ... + KSKIPI(k-1)

If only one data-set is accessed during the operation, JSK is not defined for that operation.

Now, the next operation to occur during the flight is selected from the sequences of operations and NSK(2) and JSK(2) are computed as before. This process continues until all m scheduled operations and all n unscheduled operations have been simulated. Thus, $m + n$ values of NSK and c values of JSK (where $c \leq (m + n)$ and represents the number of operations where $k > 1$) are determined for the simulated flight.

Replications of the mission simulation using the same tape layout and scheduled operations with another set of randomly selected unscheduled operations can be requested. The number of replications is determined by the number of random number seed values stored in another input file. (Appendix B.5 defines the format of this file called SEEDS.DAT.) To stop the simulation replications, a seed value of zero should be specified.

4.3 Reports

The simulation generates reports that permitted the evaluation of the tape layout. Three reports are generated to show the pertinent tape layout information. The first report is a listing of the data-set input file. The next report lists each data-set's name, description, size, assigned track, file, subfile, and block, and whether its position was fixed or variable. The data-sets are listed in the same order as they were input and processed. The third report is an overview of the MMU tape after all the data-sets have been positioned. It indicates the number of blocks used in each subfile where the characters "., 1, 2, ..., U, V, and W" correspond to 0, 1, 2, ..., 30, 31, and 32 blocks used. These reports are shown in Appendix C.1, C.2, and C.3, respectively. They permitted the examination of the data-set positions assigned by the layout algorithm.

The remaining two reports are related to the simulation of tape travel. The fourth report traces chronologically both scheduled and unscheduled operations and their tape travel. Scheduled operations are identified with the mission timeline model name and step number. Unscheduled operations are identified with the name of the first data-set accessed (i.e., there is no comma and number at the end). For each operation, the blocks skipped from the last operation (NSK) and the blocks skipped within the operation (JSK) are shown. Also, the tape position (file, subfile, and block) after the operation is complete is listed, also. (An example page from this report is shown in Appendix C.4.)

The fifth report is produced at the completion of each simulated mission and consists of tables that summarize the data-set accesses. This report is shown in Appendix C.5. The mission run number for the tape layout proposal is listed in the title line followed by the date and time of the simulation. Next, the ten largest values of NSK(i) (the blocks skipped between operations), JSK(i) (the total blocks skipped within an operation), and KSK(i) (the maximum travel for a single access within an operation) are listed. These three tables give the time of the operation, the operation identifier, and the number of blocks skipped. At the bottom of each table, the total number of blocks skipped and number of observations over the mission are listed. Finally, the mean, variance, and coefficient of variation [15] of NSK, JSK, and KSK are given for the simulated mission.

4.4 Summary

The simulation uses the following data:

1. The list of MMU data-sets.

2. Unscheduled operations data.

3. The mission timeline data-base of all scheduled operations.

4. Data-set access sequences for scheduled operations.

5. A random number seed for each simulation replication.

The simulation starts by creating a candidate tape layout based on the list of data-sets. This is achieved via IBM's tape layout algorithm developed to define the data-set positions. Next, a set of unscheduled operations are randomly scheduled within the appropriate time windows and with the frequencies defined by the flight operations experts. These operations represent the predicted use of the MMU that is not scheduled in the mission timeline. The sequence of scheduled operations is derived from the mission timeline data-base of all scheduled operations. A sequence of data-set accesses are identified for each of these operations that need data-set accesses. The simulated tape layout is then used to determine the tape travel between the operations and within the operations. Replications using the same tape layout and different times for the unscheduled operations can be made. The number of replications is specified by the number of random number seeds provided. Reports are generated that will permit the tape layout and the tape travel to be reviewed and analyzed for improvement.

CHAPTER 5.

STRATEGY AND TACTICS OF MODEL OPERATION

This chapter defines how the computer simulation was used to select a "good" order for the list of Spacelab I data-sets. For Spacelab I, there were 166 data-sets to be positioned on the tape which results in 166 factorial possible list orders. Since the simulation typically uses about an hour to simulate five flights for a candidate list, only a selected subset of possible input orders can be examined. This chapter describes how a reduced set of lists were identified for evaluation based upon experience gained using the simulation. The following aspects will be discussed:

First, measures of performance will be established so different tape layouts can be quantitatively compared. A criteria function based upon these measures will be defined to evaluate the tape travel expected for a tape layout.

Second, the factors that affect tape travel will be investigated using the simulation. This is needed because Spacelab MMU tape travel has not been studied before and these factors have not been well understood.

Third, using the knowledge gained from this investigation, the list ordering factors will be categorized as to which ones should be experimentally varied and which ones should be held constant.

Fourth, a procedure will be defined to group data-sets using the prior knowledge of data-set utilization within operations. By fixing the orders within the groups, the order of the complete data-set list will be constrained.

Finally, a heuristic will be defined in which the experimental factors to be varied will be investigated. The experiment results should identify a data-set list that results in less tape travel than other lists evaluated.

5.1 Measures of Performance

The measures of performance for Spacelab I MMU tape travel were established in Chapter 3 as MTI and ATB where, MTI is an average over the flight of the maximum travel within each operation and ATB is an average of the travel between each of the flight operations. For a given tape layout, these measures are found in the simulation summary tables report for each simulation replication. For example, using the summary table report given in Table 7, MTI is 28.6 blocks and ATB is 61.7 blocks.

Since the sequence of CDMS operations is stochastic, replications of the simulated flight should be performed to gather several samples of the measures of MTI and ATB. The narrowing of the confidence intervals with increased samples was balanced against the additional simulation time required per sample. The confidence intervals were computed using the fixed sample size procedure described by Law and Kelton [11] assuming the measures are normally distributed random variables. The general formula for the confidence interval is given by

$$\bar{X}(n) \pm t_{n-1,1-\alpha/2} \frac{s(n)}{\sqrt{n}} \quad .$$

where

	$\bar{X}(n)$	is the sample mean
	$s(n)$	is the sample standard deviation
	n	is the sample size
and	$t_{n-1,1-\alpha/2}$	is the value of a Student-t distribution for n-1 degrees of freedom and a $(1-\alpha/2)$ degree of confidence.

A test run of five mission replications was made which took 55 minutes to complete. The data-set list order was defined in the same way NASA has previously determined the list order, that is, without tape travel evaluations and list order revisions. The results of this test are shown in Table 8.

Table 7. Example Statistical Summary Report

TEN LARGEST "SKIP TO" HRS	MODEL,STEP	BLK
23.98	019-F1/2,2	447
52.42	TMEM	374
38.98	013-F2/3,1	331
45.32	022-F02C,2	320
144.00	U13APS	352
85.57	013-F2/3,8	331
96.47	017-F01B,1	372
116.57	U34AST	384
100.35	N2F09/8C,1	416
10.83	MMA	288

TEN LARGEST "SKIP IN" HRS	MODEL,STEP	BLK
82.43	022-F02K,1	147
32.98	N1F07B34,2	99
40.79	022-F02C,1	147
76.87	022-F02M,1	147
84.16	005-F01A,1	122
64.38	021-F1/2,1	98
10.63	033-F1/4,2	130
61.45	N1F07A6B,2	99
34.56	013-F2/3,2	275
51.18	005-F01C,3	227

TOTAL "SKIP TO":	65078
NO. OF OBS.:	1054
MEAN :	61.7
VARIANCE:	3514.7
CV(%):	96.0

TOTAL "SKIP IN":	13094
NO. OF OBS.:	240
MEAN :	54.6
VARIANCE:	3774.3
CV(%):	112.6

TEN LARGEST MAX TRAVEL IN (MTI)

HRS	MODEL,STEP	BLKS	ACC
84.16	005-F01A,1	63	3
34.56	013-F2/3,2	61	6
64.38	021-F1/2,1	59	4
82.43	022-F02K,1	63	5
32.98	N1F07B34,2	99	2
61.45	N1F07A6B,2	99	2
10.63	033-F1/4,2	124	3
51.18	005-F01C,3	74	8
40.79	022-F02C,1	63	5
76.87	022-F02M,1	63	5

TOTAL MTI :	6855
NO. OF OBS.:	240
MEAN :	28.6
VARIANCE:	472.2
CV(%):	76.1

Table 8. Test Data for Confidence Intervals

Replication	MTI	ATB
1	32.5	56.7
2	32.6	57.5
3	32.1	55.6
4	31.7	57.0
5	32.9	56.3
Mean	32.4	56.6
Std. Dev.	0.47	0.72
Deg. of Confidence	0.98	0.98
Confidence Interval	32.4±0.7	56.6±1.1

A 98 percent degree of confidence for each measure was selected. The confidence levels were chosen such that α, the overall degree of confidence for both measures' confidence intervals, would be less than 0.05. This confidence level is given by [11]

$$100(1 - \alpha) \text{ percent} = 1 - (\alpha_{MTI} + \alpha_{ATB}).$$

Thus,

$$100(1 - \alpha) = 1 - (0.02 + 0.02) = 96 \text{ percent}.$$

To put these intervals in perspective, a range of one block is equivalent to about 0.05 seconds in access time which is imperceptively small. Thus, there is reasonable assurance that the means computed using five simulation replications will be representative of the true means of the measures MTI and ATB. So, for each layout determined by a candidate data-set list, the means of MTI and ATB over five simulation replications will be used to measure the list's performance.

5.2 Criteria Function

Using the measures of performance, a criteria function can be established to compare data-set lists. This function was defined with both MTI and ATB but with greater importance given to MTI. This is achieved by using a scoring function [2,6,18] which is defined as follows:

$$SC = \frac{(BETA)\ \overline{MTI}(i)}{\overline{MTI}(0)} + \frac{(1 - BETA)\ \overline{ATB}(i)}{\overline{ATB}(0)}$$

where

$\overline{MTI}(i)$ is the mean of MTI for 5 replications for layout i,

$\overline{MTI}(0)$ is the mean of MTI for 5 replications for layout 0,

$\overline{ATB}(i)$ is the mean of ATB for 5 replications for layout i,

$\overline{ATB}(0)$ is the mean of ATB for 5 replications for layout 0

BETA is a weighting factor for the relative importance of $\overline{MTI}$ versus $\overline{ATB}$.

The weighting factor, BETA, was set to 0.75 for Spacelab I to favor the MTI measure of performance.

This function uses an initial layout, called layout 0, for comparison purposes. The score of the initial layout will be 1.0 by definition. If SC for a layout is less than 1.0, it is better than the initial layout. A perfect layout would have an SC score of zero. The initial layout represents the best possible guess at the list order NASA would have made without the benefit of the simulation. So any improvement from this candidate represents an improvement over the way the layout was previously determined by NASA. The method of selecting the initial order will be discussed later.

5.3 Investigation of the Factors Affecting Tape Travel

MMU tape travel will vary with many factors -- some that can be controlled and others that cannot. To investigate the sensitivity of tape travel to these factors, several exploratory simulation runs were made. These runs indicated that the factors listed in Table 9 affected the tape travel within an operation and between operations. Each of these factors were categorized as suggested by Bartee [2]:

A - A boundary condition (to be held constant in the experiment).

B - An unmeasured experimental factor contributing to experimental error. (In this case, the experimental error is the variation in MTI and ATB between flight replications.)

C - A measured factor in the experiment.

Table 9. Factors Affecting Data-set Positioning and Tape Travel

	FACTOR	CATEGORY
1.	Total number of data-set blocks	A
2.	Data-set size	A
3.	Data-set type	A
4.	Sequence of unscheduled operations	B
5.	Sequence of scheduled operations	A
6.	The relative list positions of data-sets accessed during an operation	A
7.	The relative list positions of data-set groups associated with an experiment	A
8.	The relative list positions of data-sets groups associated with experiment disciplines	C
9.	The list position of data-sets shared by several experiments	C
10.	The list order of the discipline groups of data-sets	A
11.	The direction of tape access	B
12.	The number of data-sets listed before the first "T" type data-set	A

Factors 1, 2, and 3 are determined by the flight software requirements. Factor 4 is determined by the schedule of operations and can be assumed to be a boundary condition if one assumes the performance of the operations is per the schedule. Factors 5 and 11 fluctuate with the random occurrence of unscheduled operations. Factors 6 through 10 and 12 are all associated with the order of the data-sets. It was found that by constraining the order of the data-sets according to the sequence of accesses expected during CDMS operations, the tape travel could be decreased. The ordering factors 6 and 7 will be controlled by fixing the order of groups of data-sets according to their sequence of access during an operation. Factor 10 can be fixed based upon the time criticality of the disciplines' operations.

Factor 12 will be held constant by listing enough data-sets before the first display data-set (type "T") to fill file 6, subfiles 0 and 1. This leaves factors 8 and 9 as the ones to be varied during the experiment.

Hypothesis Statement

Assuming the data-sets are grouped and sufficient data-sets are listed before the first display data-set is listed, the tape travel as measured by SC is dependent upon the list positions of data-sets shared by several operations and the order of the experiment data-set groups within each discipline group.

5.4 Grouping of Data-sets

To group the data-sets by experiment and discipline, each one was assigned a three character code where the first character identified the discipline that the data-set supports and the second and third characters identified the experiment or function it supports. Table 10 summarizes the codes. The discipline identified as "common" represents those data-sets that may be called at any time. The "remainder" discipline represents data-sets that are not expected to be accessed during experiment operations. These assignments were then included in the description field of the data-set list. (See Appendix B.1. as an example.)

Table 10. Data-set Grouping Codes

Discipline	Codes			
Solar Physics	S16	S21		
Astronomy	A05	A22		
Plasma Physics	P02	P03	P19	P20
Earth Observation	E13	E17	E33	E34
Common	COF	C99	CTL	CTM
Remainder	REM			

With the codes assigned, data-sets having the same code were grouped together in the order they are accessed during the operations. Then, the experiment groups were collected together by discipline. The common and remainder disciplines groups need not be collected together as they highly independent of one another. The following procedure was used to define the initial ordering:

1. Each scheduled operation's sequence of data-set accesses was reviewed and its data-sets were assigned a grouping code.

2. If more than one data-set is accessed during the operation, those data-sets were grouped together.

3. Steps 1 and 2 were repeated for the unscheduled operations.

4. The remaining data-sets were assigned to the "remainder" discipline.

5. With the operations groups of data-sets now defined, any groups or individually accessed data-sets that support different operations of the same experiment were merged.

6. The experiment groups were then combined into discipline groups.

7. The total number of blocks for the data-sets listed before the first display data-set was determined. When this total is less than 256 (4 tracks x 2 subfiles x 32 blocks), enough data-sets should be placed before this display data-set to assure the first two subfiles are filled. Data-sets identified as "common" should be used, if possible.

8. The remaining data-sets that have not been grouped up till now were added to the bottom of the list.

Let us demonstrate this procedure with an example using Tables 11 and 12. Table 11 shows a portion of the sequence of data-set accesses that are expected during scheduled operations. For the operations identified as 034-FC3, steps 1 through 7, the data-sets were grouped as shown in Table 12. Note that this same group will support several operations, i.e., 034-FC3, 034-FC4, 034-FC6, and 034-FC7. The complete list of data-sets ordered using this procedure is given in Appendix B.1.

Table 11. Example Scheduled Data-set Access Sequences

Operation	Data-sets		
034-FC3,1	S34	A34RO	U34ALM
034-FC3,3	U34AST		
034-FC3,5	U34AST		
034-FC3,7	U34AST		
034-FC4,1	S34	A34RO	U34ALM
034-FC4,3	U34AST		
034-FC4,5	U34AST		
034-FC4,7	U34AST		
034-FC6,1	S34	A34RO	U34ALM
034-FC6,3	U34AST		
034-FC6,5	U34AST		
034-FC7,1	S34	A34RO	U34ALM
034-FC7,3	U34AST		
034-FC7,5	U34AST		

Table 12. Example Group of Data-sets

S34	ECOS	E34	15	0	0	0	0	VAR
A34RO	ECOS	E34	12	0	0	0	0	VAR
U34ALM	ECOS	E34	1	0	0	0	0	VAR
U34AST	ECOS	E34	100	0	0	0	0	VAR

To show that selecting the initial order as described above will yield a good tape layout, the following test was performed. The data-sets were listed as prescribed above and the means of MTI and ATB were computed. This layout will be assumed to be the initial layout, so SC = 1.00. Two alternative lists were made by ordering them in other ways to compare the values of SC for each. The first alternative list was ordered alphabetically starting with the first character of the data-set name. The second alternative list was ordered alphabetically starting with the second character of the data-set name. A comparison of the results is tabulated in Table 13. This table shows that significantly lower tape travel was experienced when using the grouping procedure.

Table 13. Comparison of Initial Ordering Methods

Method:	Grouping Procedure	Alphabetic by 1st Character	Alphabetic by 2nd Character
Value of SC:	1.00	6.43	2.84

5.5 Experiment Design

A heuristic [19] will be used to decrease the number of layout evaluations. In this method, a candidate layout will be compared with previously evaluated layout candidates using the heuristic rules. The heuristic is defined as follows:

1. Define an initial input order for the list of data-sets per the grouping procedure given above.

2. Run the simulation and compute the means of MTI and ATB for the five flight replications.

3. After reviewing the simulation reports, revise the order of the data-set groups that have the greatest tape travel. Use the revision strategy as given in the next section.

4. With the revised list, run the simulation and compute SC for the new tape layout.

5. Compare SC for the revised order with the minimum value of SC computed for all of the previous candidate orders: Let

   SC(n) be SC for candidate list n,

   and SC(min) be the minimum value of SC for all previous candidates.

   If SC(n) < SC(min), then go to step 3. If SC(n) ≥ SC(min), then increment a value NNI by 1 where NNI represents the number of list order revisions that have not improved since SC(min) was found.

6. If NNI < 5 then go to step 3.

7. STOP. Select the list with the minimum value of SC.

5.6 Strategy of Revising the List Order

After the simulation is run, the simulation reports were reviewed to determine how the input order might be changed to decrease the blocks skipped between data-set accesses. The statistical summary was reviewed to determine which operations skip the largest number of blocks within the operation. Then, the tape positions of the data-sets accessed during these operations was reviewed to determine why the number of blocks skipped was comparatively larger.

The position of data-sets shared across disciplines should be tested first and most thoroughly. Varying the order of experiment groups within a discipline group should be tested next. Several groups may be reordered in a single revision to minimize the number of list evaluations.

5.7 Example of a List Order Revision

To demonstrate how the list order was revised, examine the statistical summary in Table 7 shown earlier. Note the seventh entry of the ten largest values in the table for "MAX TRAVEL IN (MTI)" is associated with the data-set accesses of operation 033-F1/4,2. The operation/data-set correlation input data in Appendix B.3 shows these operations involve the data-sets S33, A33A0, T33A, and U33ACC. Table 14 shows that U33ACC is positioned in file 5, subfile 0 and the others are in file 6, subfiles 3 and 4. It would be better if these data-sets were positioned more closely to file 5. This was accomplished by revising the input order as shown in the optimized input file in Appendix B.2. By moving these data-sets up in the list, they were positioned in subfiles 1 and 2 of file 6 as shown in Table 15.

Table 14. Initial Earth Observation Data-set Positions

Name	Description	Size (1)	Pos.(2) T	F	S	B	Con-trol
S13	ECOS E13	18	6	6	2	11	VAR
A13A0	ECOS E13	12	0	6	3	0	VAR
A13A01	ECOS E13	4	0	6	3	12	VAR
A13A02	ECOS E13	5	0	6	3	16	VAR
A13A03	ECOS E13	5	0	6	3	21	VAR
A13A04	ECOS E13	3	4	6	1	29	VAR
A13A06	ECOS E13	4	[illegible]	6	3	26	VAR
T13A	ECOS E13	2	6	6	2	29	VAR
U13APC	ECOS E13	1	0	6	1	31	VAR
U13APS	ECOS E13	9	4	6	3	0	VAR
A13G0	ECOS E13	4	4	6	3	9	VAR
T13G	ECOS E13	3	4	6	3	13	VAR
A13A05	ECOS E13	4	4	6	3	16	VAR
S17	ECOS E17	11	4	6	3	20	VAR
A17A0	ECOS E17	6	6	6	3	0	VAR
T17A	ECOS E17	2	0	6	3	30	VAR
U17A01	ECOS E17	8	6	6	3	6	VAR
S33	ECOS E33	44	0	6	4	0	VAR
A33A0	ECOS E33	6	6	6	3	14	VAR
T33A	ECOS E33	2	6	6	3	20	VAR
U33ACC	ECOS E33	250	0	5	0	0	VAR
S34	ECOS E34	15	2	6	4	10	VAR
A34R0	ECOS E34	12	4	6	4	0	VAR
U34ALM	ECOS E34	1	0	6	2	31	VAR
U34AST	ECOS E34	100	6	6	4	0	VAR

(1) Number of 512 word blocks.
(2) T = Track; F = File; S = Subfile; B = Block

Table 15. Revised Earth Observation Data-set Positions

Name	Description	Size (1)	Pos.(2) T	F	S	B	Con-trol
S33	ECOS E33	44	4	6	1	0	VAR
A33A0	ECOS E33	6	6	6	1	0	VAR
T33A	ECOS E33	2	0	6	2	4	VAR
U33ACC	ECOS E33	250	0	5	0	0	VAR
S17	ECOS E17	11	6	6	1	6	VAR
A17A0	ECOS E17	6	6	6	1	17	VAR
T17A	ECOS E17	2	0	6	2	6	VAR
U17A01	ECOS E17	8	6	6	1	23	VAR
S34	ECOS E34	15	0	6	2	8	VAR
A34R0	ECOS E34	12	4	6	2	12	VAR
U34ALM	ECOS E34	1	6	6	1	31	VAR
U34AST	ECOS E34	100	6	6	2	0	VAR
S13	ECOS E13	18	0	6	3	0	VAR
A13A0	ECOS E13	12	0	6	3	18	VAR
A13A01	ECOS E13	4	0	6	2	23	VAR
A13A02	ECOS E13	5	0	6	2	27	VAR
A13A03	ECOS E13	5	4	6	2	24	VAR
A13A04	ECOS E13	3	4	6	2	29	VAR
A13A06	ECOS E13	4	4	6	3	0	VAR
T13A	ECOS E13	2	0	6	3	30	VAR
U13APC	ECOS E13	1	4	6	3	4	VAR
U13APS	ECOS E13	9	4	6	3	5	VAR
A13G0	ECOS E13	4	4	6	3	14	VAR
T13G	ECOS E13	3	4	6	3	18	VAR
A13A05	ECOS E13	4	4	6	3	21	VAR

(1) Number of 512 word blocks.
(2) T = Track; F = File; S = Subfile; B = Block

CHAPTER 6.

EXPERIMENT RESULTS, CONCLUSIONS, AND RECOMMENDATIONS

In this chapter, the results of the simulation experiment will be summarized. Tape travel data will be presented for the data-set input orders evaluated, the best input order will be discussed, and conclusions will be made concerning the computer simulation and the data-set list ordering for Spacelab I. Finally, recommendations will be made for improving the simulation and using it to order data-set lists for other Spacelab flights.

6.1 Experiment Data

After the exploratory runs to determine the sensitivity of the tape travel to the ordering of the data-sets, fourteen different list orders were evaluated in the experiment. The initial order was determined using the grouping procedure derived after the investigation of ordering variations and is given in Appendix B.1. Table 16 tabulates the measures of performance for the tape layouts evaluated. The first two revisions of the list order resulted in greater tape travel than the initial order. In these orders an experiment group became split between files 6 and 5. This indicates that the tape travel may increase greatly if groups are split between different files. These two list orders were discarded as unreasonable. The order of the data-sets list was then revised eleven more times per the experiment procedure. Revisions 6 and 7 had the same value of SC so NNI was determined by selecting the order with the lower MTI mean. The eighth revision resulted in the minimum value of SC. This list order was selected as the best one evaluated and is shown in Appendix B.2.

TABLE 16. Measures of Performance Values

ORDER	$\overline{MTI}$	$\overline{ATB}$	SC	NNI
0	28.7	62.8	1.00	0
1	92.9	56.3	2.65	(Note 1)
2	93.2	102.9	2.85	(Note 1)
3	22.0	55.6	0.80	0
4	27.3	56.1	0.94	1
5	26.6	55.1	0.91	0
6	18.8	50.6	0.69	0
7	17.9	56.4	0.69	1 (Note 2)
8 *	11.9	55.4	0.53	0
9	13.4	56.5	0.58	1
10	15.4	55.[illegible]	0.62	2
11	18.4	52.3	0.69	3
12	12.1	55.8	0.54	4
13	18.4	52.3	0.69	5

Note 1: Discarded as an unreasonable order.
Note 2: Tie broken by selecting minimum mean of MTI.

6.2 Discussion of the Best Order

The tape travel data for the eighth list order is detailed in Table 17.

Table 17. "Best" Order Tape Travel Statistics

Replication	MTI	ATB
1	12.0	53.8
2	11.9	54.6
3	11.9	56.0
4	10.6	54.7
5	13.0	58.1
Mean	11.9	55.4
Std. Deviation	0.85	1.68
Coefficient of Variation	7.2 %	3.0 %
98% Confidence Interval	11.9±1.3	55.4±2.6

Some important attributes can be identified in order 8, the best order, that help minimize tape travel. First, the large, "common" group data-sets, excluding the "T" type are listed closer to the top of the list to fill subfiles 0 and 1 in file 6. With these subfiles filled, experiment groups will not be separated when they include "T" type data-sets. Second, within the same discipline, the experiment groups with large data-sets are listed before groups with smaller data-sets. In this case, more contiguous tape space is available to position the large data-sets and the smaller data-sets then fill in the smaller, unallocated gaps remaining. Third, the data-set U33ACC, which is 250 blocks large, was listed so it was positioned in file 5. This permitted the remaining experiment and "common" data-set groups to be positioned in file 6. Fourth, the more frequently accessed data-sets are positioned toward the center of the list. This tends to distribute as many data-sets to the left as to the right of them on the MMU tape. Finally, the data-set U05PMU was listed with the S21 group. In this position it was centered on the tape relative to the different groups that include its access.

6.3 Sensitivity Analysis

To investigate the sensitivity of changes in the assumptions made about MMU use, several test cases were defined. In the test cases described below, the "best" order's simulation input data was changed in ways that might actually occur. Table 18 summarizes the statistical data for the best order and each test case.

Case A: Unexpected MMU Accesses

For the best order, an unscheduled set of operations that seemed likely to occur and cause MMU accesses was simulated. In this test case, the sensitivity of tape travel to additional unscheduled operations was examined. (This situation occurred during Spacelab I and will be noted later.) This case was defined by adding 28 unscheduled accesses of the data-sets TVTR and TVID. As Table 18 below shows, the simulation predicts an increase in travel between operations (ATB) with little change in travel within the operations (MTI).

Case B: Fewer Data-sets Listed Before the First Display Data-set

In the best order, enough data-sets were listed before the first display data-set (type "T") to keep the display data-set positioned close to the other members of its experiment group. In this test case some of the data-sets listed before the first display data-set were moved down in the list. The simulation predicts, in this case, that the average travel within an operation increased while the travel between operations remained nearly the same.

Case C: Change List Positions of the Common Discipline Group

Another assumption made to reduce the number of list evaluations was that the order of the discipline data-set groups could be fixed. Consistent with this, the data-sets identified as "common" i.e., not belonging to one discipline, were listed centrally between the other disciplines. In this test case, these "common" discipline data-set groups were moved ahead of the astronomy discipline data-sets. The tape travel within the operations increased while the travel between operations decreased slightly.

Case D: Data-set Size Changes

The sizes of the data-sets were assumed to be given and unable to be changed. Actually, data-set sizes can be changed early in the flight software development period. To investigate the effect of a size change, the data-set U33ACC was decreased from 250 to 25 blocks. When this was done, U33ACC was positioned in file 6 rather than file 5. The average tape travel within an operation within an operation increased significantly while the travel between operations decreased.

Case E: Deletion of Operations

Contingencies can occur immediately before or in flight that could cause some scheduled operations to be canceled. This is a deviation from the assumption that the operations would be performed per the schedule. This situation was tested by deleting the data-set accesses associated with an experiment. The simulation indicated no significant change in tape travel in this case.

Case F: Changes to the Order of Data-set Accesses

The sequence of data-set accesses associated with an operation could possibly change if the operation's procedures change. To test the sensitivity of tape travel to changes in the assumed sequences, the data-sets associated with an experiment were re-ordered in the data-set list. The order of the data-sets in the list was then different from the operation's defined order of accesses. The simulation predicts an increase in average travel within operations with no change in travel between operations.

Table 18. Sensitivity Analysis Summary

	98% Confidence Intervals MTI	ATB
"Best" Order	11.9±1.3	55.4±2.6
Case A	10.9±1.5	64.5±1.1
Case B	18.2±1.1	53.3±2.5
Case C	19.5±1.2	51.4±2.1
Case D	21.3±0.5	51.0±1.9
Case E	11.8±1.3	55.4±2.6
Case F	14.1±1.3	55.9±2.6

6.4 Weighting of the Measures of Performance

A change in the weighting factor, BETA, could affect which list order was selected. The variation of SC with changes in BETA was investigated using the fourteen candidates' measures of performance. Table 19 shows how SC varied for BETA = 0.25, 0.50, 0.65, 0.75 and 0.85. The data indicates that the same data-set list order would be selected for all values of BETA tested except when BETA = 0.25.

Table 19. SC Computed for Various Values of BETA

ORDER	BETA = 0.25	0.50	0.65	0.75	0.85
0	1.00	1.00	1.00	1.00	1.00
1	1.48	2.07	2.42	2.65	2.89
2	2.04	2.44	2.68	2.78	3.01
3	0.86	0.83	0.81	0.80	0.78
4	0.91	0.92	0.93	0.93	0.94
5	0.89	0.90	0.91	0.91	0.92
6	0.77	0.73	0.71	0.69	0.68
7	0.83	0.76	0.72	0.69	0.66
8 *	0.77	0.65	0.58	0.53	0.48
9	0.79	0.68	0.62	0.58	0.53
10	0.79	0.71	0.66	0.62	0.59
11	0.78	0.74	0.71	0.69	0.67
12	0.77	0.66	0.59	0.54	0.49
13	0.78	0.74	0.71	0.69	0.67

6.5 MMU Operations During Spacelab I

The Spacelab I flight took place November 28 through December 8, 1983. The layout for the flight MMU tape was determined using a preliminary version of the computer simulation discussed in this thesis. Tape layout studies similar to the ones discussed herein were done but a different schedule of experiment operations was used. After the tape layout was determined based upon a schedule for a September 30, 1983 launch date, the flight was delayed to the November 28, 1983 and a revised schedule of experiment operations had to be produced by the mission planners. The MMU layout was unable to be revised using the new schedule because the MMU had already been integrated into the Spacelab and thus could not be reformatted.

Even though a different schedule was used in the simulation to determine the flight MMU layout, the MMU access times during the flight were satisfactory except for two cases. In one case, an experiment operation was not performed as planned because of an unexpected access of a data-set positioned far from the experiment data-sets resulted in the next access taking significantly longer than expected. This incident demonstrated that timely MMU accesses can be important. The second case was a complaint by the astronauts during post-flight debriefings that the crew displays defined by data-sets TVID and TVTR took exceptionally long to become available when requested. This was because the astronauts used these displays more often than expected. The simulation input data did not reflect that these displays would ever be used and thus they were positioned some distance from the other data-sets. This demonstrated that the frequency of use of each data-set must be carefully estimated.

6.6 Conclusions

As a result of this study, the following conclusions can be made concerning the simulation and the list ordering of the Spacelab I data-sets for decreased MMU tape travel. The computer program provided an independent means for NASA to determine a tape layout by ordering the list of data-sets for the flight software integration contractor. As a part of the simulation, the tape travel that would be expected for the resulting tape layout is predicted using the schedule of Spacelab operations and simulated unscheduled operations. The integration of the MMU layout algorithm, the data-base of scheduled operations, and the simulated unscheduled operations into a single computer program provided a rapid way of assessing various data-set lists. With the capability to quickly assess an ordered list of data-sets, various data-set list orders could be investigated to determine ordering strategies. Grouping the data-sets by experiment and science discipline reduced travel and restricted the possible list orderings. Simulation experiments were performed based on these strategies and a better list order was determined to define the MMU layout for flight.

The simulation was used to determine the MMU tape layout for Spacelab I. The data-set access times were satisfactory during the flight with the exception of the two display data-sets discussed earlier. The simulation would have indicated this flight problem had the frequency of these display accesses been predicted correctly.

6.7 Recommendations

The following recommendations are made regarding the simulation and data-set list ordering for future Spacelabs. First, the simulation's trace report should be made optional to decrease the printout produced for simulation runs that do not require the detailed data. Next, understanding the factors that will affect tape travel is particularly important to determine good ordering strategies. Thus, a generous number of simulation runs should be allocated to studying these factors. It is also recommended that the data which correlates data-sets to operations be thoroughly reviewed and defined with operations personnel familiar with various aspects of the Spacelab flight. This data will be critical to the validity of the the simulation results. Finally, the simulation should routinely be used to perform data-set ordering analyses for Spacelab flights with a significant volume of MMU accesses (especially if astronaut initiated) or with accesses that must be timely.

LIST OF REFERENCES

1. Adam, Nabil R. and Dogramici, Ali, ed., Current Issues in Computer Simulation, Academic Press, New York, 1979.

2. Bartee, Edwin M., Engineering Experimental Design Fundamentals, Prentice-Hall, Inc., Englewood Cliffs, N. J., 1968.

3. Bobillier, P. A., et al, Simulation with GPSS and GPSS V, Prentice-Hall, Inc., Englecliffs, New Jersey, 1976.

4. Experiment Computer Operating System Design Specification, Marshall Space Flight Center, Huntsville, Alabama, February 1983.

5. Experiment Computer Operating System Requirements Specification, Marshall Space Flight Center, Huntsville, Alabama, March 1981.

6. Francis, Richard L., Facility Layout and Location, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1974.

7. Gilmore, P. C. and Gomory, R. E., "The Theory and Computation of Knapsack Functions, Operations Research, 14(6), 1966.

8. Greenberg, Harold, Integer Programming, Academic Press, Inc., New York, 1971.

9. Hicks, Charles R., Fundamental Concepts in the Design of Experiments, Holt, Rinehart and Winston, New York, 1964.

10. Hillier, Frederick S. and Lieberman, Gerald J., Introduction to Operations Research, Holden-Day, Inc., San Francisco, 1980.

11. Law, Averill M. and Kelton, W. David, Simulation Modeling and Analysis, McGraw-Hill Book Company, New York, 1981.

12. Lipschutz, Seymour, Theory and Problems of Set Theory and Related Topics, McGraw-Hill Book Company, New York, 1964.

13. Martin, James, Computer Data-Base Organization, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1974.

14. Mass Memory Unit Specification, S. A. MATRA, Velizy, France, August 16, 1978.

15. Miller, Irwin, and Freund, John E., Probability and Statistics for Engineers, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1977.

16. Payne, James A., Introduction to Simulation: Programming Techniques and Methods of Analysis, McGraw-Hill Book Company, New York, 1982.

17. Pritsker, A. Alan B., and Pegden, Claude Dennis, Introduction to Simulation and SLAM, John Wiley and Sons, Inc., New York, 1979.

18. Shannon, Robert E., Engineering Management, John Wiley and Sons, Inc., New York, 1980.

19. Shannon, Robert E., Systems Simulation, The Art Science, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1975.

20. Spacelab Mission 1 Flight Software Design Specification, ECO 8951C, Marshall Space Flight Center, Huntsville, Alabama, May 1983.

21. Spacelab Mission 1 Flight Software Requirements Specification, MDC G8315C, Marshall Space Flight Center, Huntsville, Alabama, December 1982.

22. Spacelab One, Flight Definition Document, Systems Analysis and Integration Laboratory, Marshall Space Flight Center, Huntsville, Alabama, March 1981.

23. Spacelab One, Software Requirements Document, JA-026C, Marshall Space Flight Center, Huntsville, Alabama, 1981.

24. Spacelab Payload Accomodation Handbook, Marshall Space Flight Center, Huntsville, Alabama, 1978.

25. VAX/VMS FORTRAN Language Reference Manual, Digital Equipment Corporation, Maynard, Massachusetts, 1980.

26. VAX/VMS FORTRAN User's Guide, Digital Equipment Corporation, Maynard, Massachusetts, 1980.

27. Vemuri, V., Modeling of Complex Systems. An Introduction, Academic Press, New York, 1978.

28. Wiederhold, Gio., Database Design, McGraw-Hill, Inc., New York, 1977.

APPENDIX A

PROGRAM LISTINGS

This appendix contains listings of the simulation program. The program consists of seventeen FORTRAN 77 routines. The converted MMU allocation program is represented by the main program and the first five subroutines. The remaining subroutines were developed to support the simulation of data-set accesses.

```
C23456789 123456789 123456789 123456789 123456789 123456789 1234567890 12
C       STMT 2-24
C       MMU ALLOCATION PROGRAM
C
C       MMUALL PROGRAM SPECIFICATION STATEMENTS
        INTEGER NUMBLOCKS(300),TRACK(300),FILE(300),SUBFILE(300)
        INTEGER BLOCK(300),NUMENTRIES,MMUMAP(16384)
        CHARACTER DSNAME(300)*8,DESCRIPTION(300)*30,RELOCATABLE(300)*5
        CHARACTER COMMENTS(300)*10,DUMMEP(300)*9
        COMMON /BLK1/NUMBLOCKS,TRACK,FILE,SUBFILE,BLOCK
        COMMON /BLK2/NUMENTRIES,MMUMAP
        COMMON /BLK3/DSNAME,DESCRIPTION,RELOCATABLE,COMMENTS,DUMMEP
        OPEN(UNIT=5,READONLY,TYPE='OLD')
        OPEN(UNIT=6,TYPE='NEW',
       *CARRIAGECONTROL='FORTRAN')
C  ALL BLOCKS ARE FLAGGED AS AVAILABLE
        CALL LOAD          !LOAD INPUT DIRECTORY
        CALL MAPFIXED      !FLAG FORCED ALLOCATIONS IN MMUMAP
        CALL ASSIGN        !ALLOCATE DATA SETS BY ALGORITHM
        CALL PRINT         !PRINT DIRECTORY IN SAME ORDER AS INPUT
        CALL PRINTMAP      !PRINT MAP OF MMU UTILIZATION
        CLOSE(UNIT=5)
C  THE FOLLOWING ROUTINES PROVIDE FOR THE SIMULATION
        CALL READMODIN
  1     CALL UNSCH(ISTOP)
        IF(ISTOP.EQ.1)GOTO99
        CALL SIM(NSK,NSK2,NTO,INSKIP,INSQ,IN,MSKIP,MSSQ,MN)
        CALL SUMRY(NSK,NSK2,NTO,INSKIP,INSQ,IN,MSKIP,MSSQ,MN)
        GOTO 1
99      CLOSE(9)
        CLOSE(UNIT=6)
        STOP 'END OF MMU LAYOUT PROGRAM'
        END
```

```
C
C--------------------------------------------------------------
C  STMT 25-32
      SUBROUTINE LOAD             !LOAD INITIAL INPUT DIRECTORY
C     MMUALL PROGRAM SPECIFICATION STATEMENTS
      INTEGER NUMBLOCKS(300),TRACK(300),FILE(300),SUBFILE(300)
      INTEGER BLOCK(300),NUMENTRIES,MMUMAP(16384)
      CHARACTER DSNAME(300)*8,DESCRIPTION(300)*30,RELOCATABLE(300)*5
      CHARACTER COMMENTS(300)*10,DUMMEP(300)*9
      COMMON /BLK1/NUMBLOCKS,TRACK,FILE,SUBFILE,BLOCK
      COMMON /BLK2/NUMENTRIES,MMUMAP
      COMMON /BLK3/DSNAME,DESCRIPTION,RELOCATABLE,COMMENTS,DUMMEP
      NUMENTRIES=0
10    NUMENTRIES=NUMENTRIES+1
      INK=NUMENTRIES
      READ(5,100,END=90)DSNAME(INK),DESCRIPTION(INK),
     *NUMBLOCKS(INK),TRACK(INK),FILE(INK),SUBFILE(INK),
     *BLOCK(INK),RELOCATABLE(INK),COMMENTS(INK),DUMMEP(INK)
100   FORMAT(A8,1X,A30,2X,I3,2X,I1,1X,I1,1X,I1,1X,I2,1X,A5,1X,A10,A9)
      WRITE(6,101)DSNAME(INK),DESCRIPTION(INK),
     *NUMBLOCKS(INK),TRACK(INK),FILE(INK),SUBFILE(INK),
     *BLOCK(INK),RELOCATABLE(INK),COMMENTS(INK),DUMMEP(INK)
101   FORMAT(1X,A8,1X,A30,2X,I3,2X,I1,1X,I1,1X,I1,1X,I2,1X,A5,1X,A10,A9)
      IF(DESCRIPTION(INK)(1:3).NE.'END')GOTO 10
      NUMENTRIES=NUMENTRIES-1
90    RETURN
      END
```

```
C
C-----------------------------------------------------------------
C  STMT 33-40
      SUBROUTINE MAPFIXED
C     MMUALL PROGRAM SPECIFICATION STATEMENTS
      INTEGER NUMBLOCKS(300),TRACK(300),FILE(300),SUBFILE(300)
      INTEGER BLOCK(300),NUMENTRIES,MMUMAP(16384)
      CHARACTER DSNAME(300)*8,DESCRIPTION(300)*30,RELOCATABLE(300)*5
      CHARACTER COMMENTS(300)*10,DUMMEP(300)*9
      COMMON /BLK1/NUMBLOCKS,TRACK,FILE,SUBFILE,BLOCK
      COMMON /BLK2/NUMENTRIES,MMUMAP
      COMMON /BLK3/DSNAME,DESCRIPTION,RELOCATABLE,COMMENTS,DUMMEP
      DO I=1,NUMENTRIES
        IF(RELOCATABLE(I).EQ.'FIXED')THEN
                CALL AVAIL(I,IOUT)
                IF(IOUT.EQ.0)THEN
                        WRITE(6,100)DSNAME(I)
100                     FORMAT(1X,'@@@@MMU ADDRESSING ERROR FOR ',A8)
                ELSE
                        CALL RESERVE(I) !UPDATE MMUMAP FOR FORCED DATA
                END IF
        END IF
      END DO
      RETURN
      END
```

```
C
C-----------------------------------------------------------------
C  STMT 41-45
      SUBROUTINE RESERVE(I)      !SUBR TO UPDATE MMUMAP FOR ASSIGNMENT
C     MMUALL PROGRAM SPECIFICATION STATEMENTS
      INTEGER NUMBLOCKS(300),TRACK(300),FILE(300),SUBFILE(300)
      INTEGER BLOCK(300),NUMENTRIES,MMUMAP(16384)
      CHARACTER DSNAME(300)*8,DESCRIPTION(300)*30,RELOCATABLE(300)*5
      CHARACTER COMMENTS(300)*10,DUMMEP(300)*9
      COMMON /BLK1/NUMBLOCKS,TRACK,FILE,SUBFILE,BLOCK
      COMMON /BLK2/NUMENTRIES,MMUMAP
      COMMON /BLK3/DSNAME,DESCRIPTION,RELOCATABLE,COMMENTS,DUMMEP
      IF(FILE(I).EQ.0.AND.SUBFILE(I).EQ.0.AND.BLOCK(I).EQ.0
     *.AND.TRACK(I).EQ.6)THEN    !LAST MMU ADDRESS MEANS NO SPACE
                WRITE(6,100)DSNAME(I)
100             FORMAT(1X,'$$$$$MMU SPACE NOT AVAILABLE FOR ',A8)
      ELSE
        DO J=1,NUMBLOCKS(I)
        MMUMAP(TRACK(I)*2048+FILE(I)*256+SUBFILE(I)*32+J+BLOCK(I))=1
        END DO
      END IF
      RETURN
      END
```

```
C
C--------------------------------------------------------------
      SUBROUTINE AVAIL(I,IOUT)  !SUBR TO TEST AVAILABILITY & ADDRESSING
C     MMUALL PROGRAM SPECIFICATION STATEMENTS
      INTEGER NUMBLOCKS(300),TRACK(300),FILE(300),SUBFILE(300)
      INTEGER BLOCK(300),NUMENTRIES,MMUMAP(16384)
      CHARACTER DSNAME(300)*8,DESCRIPTION(300)*30,RELOCATABLE(300)*5
      CHARACTER COMMENTS(300)*10,DUMMEP(300)*9
      COMMON /BLK1/NUMBLOCKS,TRACK,FILE,SUBFILE,BLOCK
      COMMON /BLK2/NUMENTRIES,MMUMAP
      COMMON /BLK3/DSNAME,DESCRIPTION,RELOCATABLE,COMMENTS,DUMMEP
      IF(DESCRIPTION(I)(1:4).EQ.'ECOS'.AND.
     *   DSNAME(I)(1:1).EQ.'T'.AND.
     *   SUBFILE(I).NE.2.AND.
     *   SUBFILE(I).NE.3.AND.
     *   SUBFILE(I).NE.4.AND.
     *   SUBFILE(I).NE.5)THEN   !ALLOC ECOS DISPLAYS ONLY IN SF 2-5
               IOUT=0
               RETURN
      END IF
      IF(NUMBLOCKS(I).GT.256)THEN        !DATA SET .GT. FILE
C       ASSURE START OF FILE AND ROOM ON TAPE
        IF((SUBFILE(I).EQ.0.AND.BLOCK(I).EQ.0.AND.(FILE(I)*256
     *      +NUMBLOCKS(I)).LE.2048).EQ..FALSE.)THEN
               IOUT=0
               RETURN
        END IF
      ELSE IF(NUMBLOCKS(I).GT.32)THEN            !.LE. FILE
        IF((BLOCK(I).EQ.0.AND.(SUBFILE(I)*32+NUMBLOCKS(I))
     *  .LE.256).EQ..FALSE.)THEN
               IOUT=0
               RETURN
        END IF
        ELSE IF(BLOCK(I)+NUMBLOCKS(I).GT.32)THEN
                     IOUT=0
                     RETURN
      END IF
        DO II=1,NUMBLOCKS(I)
        IF(MMUMAP(TRACK(I)*2048+FILE(I)*256+SUBFILE(I)
     *         *32+BLOCK(I)+II).NE.0)THEN
               IOUT=0
               RETURN
        END IF
        END DO
        IOUT=1
        RETURN
      END
```

```
C
C------------------------------------------------------------
C  STMT 63-84
      SUBROUTINE ASSIGN
C     MMUALL PROGRAM SPECIFICATION STATEMENTS
      INTEGER NUMBLOCKS(300),TRACK(300),FILE(300),SUBFILE(300)
      INTEGER BLOCK(300),NUMENTRIES,MMUMAP(16384)
      CHARACTER DSNAME(300)*8,DESCRIPTION(300)*30,RELOCATABLE(300)*5
      CHARACTER COMMENTS(300)*10,DUMMEP(300)*9
      INTEGER JORDER(8)
      COMMON /BLK1/NUMBLOCKS,TRACK,FILE,SUBFILE,BLOCK
      COMMON /BLK2/NUMENTRIES,MMUMAP
      COMMON /BLK3/DSNAME,DESCRIPTION,RELOCATABLE,COMMENTS,DUMMEP
      DATA JORDER/6,5,7,4,3,2,1,0/       !FILE ALLOCATION ORDER
      DO I=1,NUMENTRIES
        IF(RELOCATABLE(I)(2:4).EQ.'VAR')THEN    !SKIP FIXED ASSIGNMENTS
                DO NN=1,8
                FILE(I)=JORDER(NN)
                DO K=0,7
                        IF(JORDER(NN).GE.6)THEN
                                SUBFILE(I)=K
                        ELSE
                                SUBFILE(I)=7-K
                        END IF
                        DO L=0,6,2
                                TRACK(I)=L
                                DO M=0,31
                                   IF(JORDER(NN).GE.6)THEN
                                      BLOCK(I)=M
                                   ELSE
                                      BLOCK(I)=31-M
                                   END IF
                                   CALL AVAIL(I,IOUT)
                                   IF(IOUT.EQ.1)THEN
                                      CALL RESERVE(I)
                                      GO TO 10
                                   END IF
                                END DO
                        END DO
                END DO
                END DO
        END IF
10    CONTINUE
      END DO
      RETURN
      END
```

```
C
C--------------------------------------------------------------
C     STMT 85-90
          SUBROUTINE PRINT              !PRINT DIRECTORY
C         MMUALL PROGRAM SPECIFICATION STATEMENTS
          INTEGER NUMBLOCKS(300),TRACK(300),FILE(300),SUBFILE(300)
          INTEGER BLOCK(300),NUMENTRIES,MMUMAP(16384)
          CHARACTER DSNAME(300)*8,DESCRIPTION(300)*30,RELOCATABLE(300)*5
          CHARACTER COMMENTS(300)*10,DUMMEP(300)*9
          COMMON /BLK1/NUMBLOCKS,TRACK,FILE,SUBFILE,BLOCK
          COMMON /BLK2/NUMENTRIES,MMUMAP
          COMMON /BLK3/DSNAME,DESCRIPTION,RELOCATABLE,COMMENTS,DUMMEP
          WRITE(6,102)
102       FORMAT('1ALLOCATED DIRECTORY'/)
          DO INK=1,NUMENTRIES+1
          WRITE(6,101)DSNAME(INK),DESCRIPTION(INK),
         *NUMBLOCKS(INK),TRACK(INK),FILE(INK),SUBFILE(INK),
         *BLOCK(INK),RELOCATABLE(INK),COMMENTS(INK),DUMMEP(INK)
101       FORMAT(1X,A8,1X,A30,2X,I3,2X,I1,1X,I1,1X,I1,1X,I2,1X,A5,1X,A10,A9)
          END DO
          RETURN
          END
```

```
C
C--------------------------------------------------------------
C     STMT 125-147
          SUBROUTINE PRINTMAP
          CHARACTER MMUBUSY(8,8)*1/64*' '/
C     MMU MAP WITH ONE CHARACTER PER SUBFILE
          CHARACTER BLOCKCOUNTS(33)*1
          INTEGER SUBFILESIZE,MMUMAP(16384)
          COMMON /BLK2/NUMENTRIES,MMUMAP
          DATA BLOCKCOUNTS/'.','1','2','3','4','5','6','7','8','9'
         *,'A','B','C','D','E','F','G','H','I','J','K','L','M','N'
         *,'O','P','Q','R','S','T','U','V','W'/
          WRITE(6,100)
100       FORMAT('1',11X,'FILE 0   FILE 1   FILE 2   FILE 3   ',
         *'FILE 4   FILE 5   FILE 6   FILE 7')
          DO 10 I=1,8,2        !COMPUTE AND PRINT 8 LINES, 1/TRACK
          DO 20 J=1,8          !COMPUTE 8 FILES OR EACH LINE (TRACK)
          DO 30 K=1,8          !COMPUTE 8 SUBFILES FOR EACH FILE
C
          SUBFILESIZE=0        !INITIALIZE TO ZERO
          DO 40 L=1,32         !CHECK ALL BLOCKS WITHIN SUBFILE
          ISUB=(I-1)*2048+(J-1)*256+(K-1)*32+L
          IF(MMUMAP(ISUB).NE.0) THEN
                    SUBFILESIZE=SUBFILESIZE+1
          END IF
40        CONTINUE
          MMUBUSY(K,J)=BLOCKCOUNTS(SUBFILESIZE+1)
30        CONTINUE
20        CONTINUE
          WRITE(6,101)(I-1),MMUBUSY
101       FORMAT(' TRACK ',I1,3X,8(8A1,1X))
          WRITE(6,101)I,MMUBUSY
          DO 60 II=1,8
          DO 60 JJ=1,8
60        MMUBUSY(II,JJ)=' '
10        CONTINUE
          RETURN
          END
```

```
      SUBROUTINE READMODIN
      CHARACTER MODST*12,DSN(15)*10
      N=0
      OPEN(2,STATUS='SCRATCH',ORGANIZATION='INDEXED',
     *ACCESS='KEYED',RECORDTYPE='VARIABLE',FORM='UNFORMATTED',
     *RECL=41,KEY=(1:12:CHARACTER))
      OPEN(3,NAME='MSDS.DAT',STATUS='OLD',READONLY)
      OPEN(9,NAME='SEEDS.DAT',STATUS='OLD',READONLY)
    1 N=N+1
      READ(3,10,END=90)MODST,DSN
10    FORMAT(A12,15A10)
      WRITE(2,ERR=91)MODST,DSN
      GOTO 1
   91 WRITE(6,20)N,MODST,DSN
   20 FORMAT(/1X,'***ERROR WRITING MODEL FILE LINE:',
     */1X,I3,1X,A12,6A10/17X,6A10/17X,3A10)
      GOTO 1
   90 CLOSE(3)
        RETURN
        END
```

```
C
C-----------------------------------------------------------------
C
      SUBROUTINE UNSCH(ISTOP)
      CHARACTER*10 DSN(15)
      COMMON /BLK5/ISEED
C     WRITE(1,150)
C 150 FORMAT(1X,'INPUT A LARGE ODD INTEGER FOR A SEED(87654321):',$)
      READ(9,151,END=92)ISEED
  151 FORMAT(I8)
      IF(ISEED.EQ.0)THEN
92      ISTOP=1
        RETURN
      END IF
      WRITE(6,152)ISEED
152   FORMAT(//' NEXT SEED VALUE=',I10)
      OPEN(4,STATUS='SCRATCH',ORGANIZATION='INDEXED',
     *ACCESS='KEYED',RECORDTYPE='VARIABLE',RECL=39,KEY=(1:4:INTEGER)
     *,FORM='UNFORMATTED')
      OPEN(7,NAME='UNSCHE.DAT',STATUS='OLD',READONLY)
   20 READ(7,10,END=90)N,START,END,DSN
   10 FORMAT(I4,2F8.3,15A10)
      DO II=1,N
        CALL UNIFRM(START,END,TIME)
        TIME=TIME*3600.
        ISEC=IFIX(TIME)
70      WRITE(4,IOSTAT=IERR)ISEC,DSN
        THRS=TIME/3600.
        IF(IERR.EQ.50)THEN
                WRITE(6,60)THRS
60              FORMAT(' @@@DIAG: DUPLICATE UNSCHED TIME=',F8.2,' HRS')
                ISEC=ISEC+1
                GOTO 70
        ELSE IF(IOSTAT.NE.0)THEN
                 WRITE(6,11)
   11            FORMAT(' ***ERROR IN WRITING UNSCHEDULED ACCESSES FILE')
        END IF
      END DO
      GO TO 20
   90 CLOSE(7)
      RETURN
      END
```

```
C
C-----------------------------------------------------------------
C
      SUBROUTINE UNIFRM(A,B,X)
      COMMON /BLK5/ISEED
      RN=RAN(ISEED)
      X=A+(B-A)*RN
      RETURN
      END
```

```
C
C-----------------------------------------------------------------
C
      SUBROUTINE ENCMODST(DATA,MODST)
      CHARACTER*12 MODST,MODEL*8,STEP*4
      REAL*4 DATA(15)
      CALL TRANR_C(DATA,5,6,MODEL)
      CALL TRANR_C(DATA,7,7,STEP)
      DO II=2,4
      IF(STEP(II:II).NE.' ')GOTO 30
      END DO
30    GOTO (1,2,3)(II-1)
 1    MODST=MODEL//','//STEP(II:4)
      RETURN
 2    MODST=MODEL//','//STEP(II:4)//' '
      RETURN
 3    MODST=MODEL//','//STEP(II:4)//'  '
      RETURN
      END
```

```
C***
      SUBROUTINE TRANR_C(DATA,ISTW,IENDW,STRING)
      CHARACTER STRING*(*)
      REAL*4 DATA(15)
C  ISTW=START WORD OF DATA(15)---IENDW=END WORD.
      NOW=(IENDW-ISTW+1)
      L=NOW*4
      ENCODE(L,101,STRING(1:L)) (DATA(I),I=ISTW,IENDW)
101   FORMAT(<NOW>A4)
C
      RETURN
      END
```

```
0001      C
0002      C-------------------------------------------------------------
0003      C
0004            SUBROUTINE SIM(NSK,NSK2,NTO,INSKIP,INSQ,IN,MSKIP,MSSQ,MN)
0005            CHARACTER DSN(15)*10,MODST*12,FID(3)*4
0006            REAL*4 DATA(15),NSK2,INSQ
0007            COMMON /BLK5/ISEED
0008            DATA FID/'MMUA','LL.P','RT  '/
0009            NSK=0
0010            NSK2=0.
0011            NTO=0
0012            INSKIP=0
0013            INSQ=0.
0014            IN=0
0015            MSKIP=0
0016            MSSQ=0
0017            MN=0
0018            NEXT=1
0019            NREAD=-1
0020            WRITE(6,50)
0021         50 FORMAT('1',10X,'S I M U L A T E D   M I S S I O N   O F',
0022           *'   M M U   U S E     '/1X,74('-')/
0023           *'                       |-------BLOCKS SKIPPED-----------|'/
0024           *'   MET   OPERATION    | BEFORE  |  TOTAL |  MAX     ON  |',
0025           *'  LAST POSITION'/
0026           *'  (HRS) (MODEL,STEP) |ACCESS #1|      IN |   IN  ACCESS#|',
0027           *'  FILE   SF    BLK')
0028            LF=6
0029            LSF=0
0030            LB=0
0031            OPEN(8,NAME='EXPERIMNT.FIN',STATUS='OLD'
0032           *,READONLY,ACCESS='DIRECT',RECORDSIZE=15)
0033         90 CONTINUE
0034            IF(NREAD.EQ.-1)THEN
0035              CALL READOF(8,FID,5,NEXT,NS,DATA,*1199)
0036              READ(4,KEYGE=0,KEYID=0,END=99)ISEC,DSN
%FORT-W-INVENDKEY, Invalid END= keyword, ignored
        [EYID=0,END=99)] in module SIM at line 36

0037            END IF
0038            IF(NREAD.EQ.0)CALL READOF(8,FID,5,NEXT,NS,DATA,*1199)
0039            IF(NREAD.EQ.1)READ(4,END=99)ISEC,DSN
0040            SHRS=DATA(2)
0041            UHRS=ISEC/3600.
0042            IF(SHRS.LE.UHRS)THEN
0043              NREAD=0
0044              CALL ENCMODST(DATA,MODST)
0045              READ(2,KEY=MODST,KEYID=0,ERR=90,END=90)MODST,DSN
%FORT-W-INVENDKEY, Invalid END= keyword, ignored
        [ERR=90,END=90)] in module SIM at line 45

0046              GO TO 20
0047            ELSE
0048              NREAD=1
0049              READ(4,KEY=ISEC,KEYID=0,END=99)ISEC,DSN
%FORT-W-INVENDKEY, Invalid END= keyword, ignored
        [EYID=0,END=99)] in module SIM at line 49
```

```
              MODST=DSN(1)//'     '
            END IF
         20 CALL SKIPPED(LF,LSF,LB,NSKIPT,DSN(1))
            NSK=NSK+NSKIPT
            NSK2=NSK2+(FLOAT(NSKIPT))**2
            NTO=NTO+1
            DO II=2,15
            IF(DSN(II).EQ.'            ')GOTO 10
            CALL SKIPPED(LF,LSF,LB,KSKIPI,DSN(II))
C
            IF(KSKIPI.GT.KSK)THEN
              KSK=KSKIPI
              KAC=II
            END IF
            JSK=JSK+KSKIPI
            END DO
         10 IF(II.EQ.2)THEN
              KAC=0
              JSK=0
              GOTO 30
            END IF
            INSKIP=INSKIP+JSK
            INSQ=INSQ+(FLOAT(JSK))**2
            IN=IN+1
C
            MSKIP=MSKIP+KSK
            MSSQ=MSSQ+(FLOAT(KSK))**2
            MN=MN+1
C
   30       CONTINUE
            IF(NREAD.EQ.0)THEN
              HRS=SHRS
            ELSE
              HRS=UHRS
            END IF
            CALL COMPARE(MODST,NSKIPT,JSK,HRS,KSK,KAC)
            WRITE(6,51)HRS,MODST,NSKIPT,JSK,KSK,KAC,LF,LSF,LB
   51       FORMAT(1X,F6.2,2X,A12,3X,I5,5X,I5,5(4X,I3))
C  D        WRITE(6,52)NTO,NSK,NSK2,IN,INSKIP,INSQ
C  D     52 FORMAT('+',60X,2(I6,1X,I8,1X,E13.8,2X))
            NP=NP+1
            IF(NP.EQ.51)THEN
              WRITE(6,50)
              NP=0
            END IF
            JSK=0
            KAC=0
            KSK=0
            GO TO 90
 1199       NREAD=1
            GOTO 90
         99 CLOSE(8)
            CLOSE(4)
            RETURN
            END
```

```
C
C-----------------------------------------------------------------
C
      SUBROUTINE SKIPPED(LF,LSF,LB,NSK,DSN)
      CHARACTER*10 DSN,ODSN
C     MMUALL PROGRAM SPECIFICATION STATEMENTS
      INTEGER NUMBLOCKS(300),TRACK(300),FILE(300),SUBFILE(300)
      INTEGER BLOCK(300),NUMENTRIES,MMUMAP(16384)
      CHARACTER DSNAME(300)*8,DESCRIPTION(300)*30,RELOCATABLE(300)*5
      CHARACTER COMMENTS(300)*10,DUMMEP(300)*9
      COMMON /BLK1/NUMBLOCKS,TRACK,FILE,SUBFILE,BLOCK
      COMMON /BLK2/NUMENTRIES,MMUMAP
      COMMON /BLK3/DSNAME,DESCRIPTION,RELOCATABLE,COMMENTS,DUMMEP
      IF(ODSN.EQ.DSN)GOTO 10
      DO I=1,300
        IF(DSNAME(I).EQ.DSN(1:8))GO TO 10
      END DO
      WRITE(6,100)DSN
100   FORMAT(' ***DATA SET: ',A10,' NOT IN TABLE!!')
      RETURN
10    IPOS=FILE(I)*256+SUBFILE(I)*32+BLOCK(I)
      LPOS=LF*256+LSF*32+LB
      IF(IPOS.LT.LPOS)THEN                   !READ RT TO LEFT
        NSK=IABS(LPOS-IPOS-NUMBLOCKS(I))
        IEND=IPOS
      ELSE                                   !READ LEFT TO RT
        NSK=IPOS-LPOS
        IEND=IPOS+NUMBLOCKS(I)
      END IF
      ODSN=DSN
      LF=IEND/256
      IEND=IEND-LF*256
      LSF=IEND/32
      LB=IEND-LSF*32
      RETURN
      END
```

```
C
C-----------------------------------------------------------------
C
      SUBROUTINE COMPARE(MODST,NSKIPT,JSK,HRS,KSK,KAC)
      CHARACTER MODST*12,TMODST(10)*12,IMODST(10)*12,KMODST(10)*12
      INTEGER LST(10),LSI(10),LSK(10),LSP(10)
      REAL THRS(10),XHRS(10),KHRS(10)
      COMMON /BLK4/TMODST,IMODST,KMODST
      COMMON /BLK6/LST,LSI,THRS,XHRS,LSK,KHRS,LSP
C       DETERMINE IF ENTRY IS ALREADY IN TABLE
      DO J=1,10
        IF(MODST.EQ.TMODST(J))GOTO 20
      END DO
      JJ=1
      MIN=LST(1)
C       FIND SMALLEST ENTRY IN THE TABLE
      DO K=1,10
        IF(LST(K).LE.MIN)THEN
                JJ=K
                MIN=LST(K)
        END IF
      END DO
      IF(NSKIPT.GT.MIN)THEN
                LST(JJ)=NSKIPT
                TMODST(JJ)=MODST
                THRS(JJ)=HRS
      END IF
      GOTO 100
20    IF(NSKIPT.GT.LST(J))THEN
        LST(J)=NSKIPT
        THRS(J)=HRS
      END IF
100   DO J=1,10
        IF(MODST.EQ.IMODST(J))GOTO 120
      END DO
      JJ=1
      MIN=LSI(1)
C---------------FIND SMALLEST ENTRY IN TABLE
      DO K=1,10
        IF(LSI(K).LE.MIN)THEN
                JJ=K
                MIN=LSI(K)
        END IF
      END DO
      IF(JSK.GT.MIN)THEN
                LSI(JJ)=JSK
                IMODST(JJ)=MODST
                XHRS(JJ)=HRS
      END IF
      GOTO 2100
120   IF(JSK.GT.LSI(J))THEN
        LSI(J)=JSK
        XHRS(J)=HRS
      END IF
C
C       TABLES FOR MAX BLOCKS SKIPPED IN AN OPERATION
2100   DO J=1,10
```

COMPARE

```
            IF(MODST.EQ.KMODST(J))GOTO 2120
          END DO
          JJ=1
          MIN=LSK(1)
C-----------------FIND SMALLEST ENTRY IN TABLE
          DO K=1,10
            IF(LSK(K).LE.MIN)THEN
                    JJ=K
                    MIN=LSK(K)
            END IF
          END DO
          IF(KSK.GT.MIN)THEN
                    LSK(JJ)=KSK
                    KMODST(JJ)=MODST
                    KHRS(JJ)=HRS
                    LSP(JJ)=KAC
          END IF
          RETURN
 2120      IF(KSK.GT.LSK(J))THEN
            LSK(J)=KSK
            KHRS(J)=HRS
          END IF
          RETURN
          END
```

```
      C
      C-----------------------------------------------------------------
      C
            SUBROUTINE SUMRY(NSK,NSK2,NTO,INSKIP,INSQ,IN,MSKIP,MSSQ,MN)
            CHARACTER RDATE*9,RTIME*8,MODST(10)*12,IMODST(10)*12
            CHARACTER TMODST(10)*12,KMODST(10)*12
            INTEGER LST(10),LSI(10),LSK(10),LSP(10)
            REAL NSK2,INSQ,THRS(10),XHRS(10),KHRS(10)
            COMMON /BLK4/TMODST,IMODST,KMODST
            COMMON /BLK6/LST,LSI,THRS,XHRS,LSK,KHRS,LSP
            IRUN=IRUN+1
            CALL DATE(RDATE)
            CALL TIME(RTIME)
            AVET=FLOAT(NSK)/FLOAT(NTO)
            AVEI=FLOAT(INSKIP)/FLOAT(IN)
            AVEK=FLOAT(MSKIP)/FLOAT(MN)
            VART=(FLOAT(NTO)*NSK2-(FLOAT(NSK))**2)/FLOAT(NTO*(NTO-1))
            VARI=(FLOAT(IN)*INSQ-(FLOAT(INSKIP))**2)/FLOAT(IN*(IN-1))
            VARK=(FLOAT(MN)*MSSQ-(FLOAT(MSKIP))**2)/FLOAT(MN*(MN-1))
            CVT=SQRT(VART)/AVET*100.
            CVI=SQRT(VARI)/AVEI*100.
            CVK=SQRT(VARK)/AVEK*100.
            WRITE(6,200)IRUN,RDATE,RTIME
      200   FORMAT('1  S T A T I S T I C A L',
           *'    S U M M A R Y  RUN',I3,5X,A9,2X,A8/1X,72('-')//4X,
           *'TEN LARGEST "SKIP TO"',9X,'TEN LARGEST "SKIP IN"'/
           *2(7X,'HRS  MODEL,STEP    #BLK'),/
           *4X,26('-'),4X,26('-'))
            DO J =1,10
            WRITE(6,30)THRS(J),TMODST(J),LST(J),XHRS(J),IMODST(J),LSI(J)
        30  FORMAT(1X,F9.2,2X,A12,2X,I4,1X,F9.2,2X,A12,2X,I4)
            END DO
      C
            WRITE(6,40)NSK,INSKIP,NTO,IN,AVET,AVEI,VART,VARI,CVT,CVI
         40 FORMAT(///5X,'TOTAL "SKIP TO": ',I8,5X,'TOTAL "SKIP IN":',1X,I8
           */9X,'NO. OF OBS.: ',I8,9X,'NO. OF OBS.: ',I8,
           */15X,'MEAN :      ',F5.1,15X,'MEAN :      ',F5.1/
           *12X,'VARIANCE: ',F8.1,12X,'VARIANCE: ',F8.1/
           *15X,'CV(%):      ',F5.1,15X,'CV(%):      ',F5.1)
      C
            WRITE(6,50)
      50    FORMAT(//,20X,'TEN LARGEST MAX TRAVEL IN (MTI)'/
           *           20X,'  HRS  MODEL,STEP   #BLKS  ACC#'/20X,32('-'))
            DO J=1,10
            WRITE(6,31)KHRS(J),KMODST(J),LSK(J),LSP(J)
      31    FORMAT(16X,F9.2,2X,A12,2X,I4,2X,I4)
            END DO
            WRITE(6,41)MSKIP,MN,AVEK,VARK,CVK
      41    FORMAT(///21X,'     TOTAL MTI : ',I8
           */24X,'NO. OF OBS.: ',I8
           */30X,'MEAN :      ',F5.1/
           *27X,'VARIANCE: ',F8.1/
           *30X,'CV(%):      ',F5.1)
            DO I=1,10
              TMODST(I)='            '
              IMODST(I)='            '
              KMODST(I)='            '
              LST(I)=0
              LSI(I)=0
              LSK(I)=0
              THRS(I)=0.
              XHRS(I)=0.
              KHRS(I)=0.
            END DO
            RETURN
            END
```

APPENDIX B

INPUT DATA FILES

This appendix contains listings of the input files used by the simulation computer program. The MMU allocation program input files are are given for the initial order and for the improved order. The inputs needed by the mission simulation portion of the program are also included.

B.1 INITIAL DATA-SET LIST INPUT FILE

Data-set Name	Description	Size (Blk)	T	F	S	B	Pos.
SCOSAM	SCOS IPL AMI	126	0	0	0	2	FIXED
ECOSAM	ECOS IPL AMI	6	4	1	7	0	FIXED
UECOS2	ECOS IPL AMI	16	4	2	1	0	FIXED
UECOS5	ECOS IPL AMI	16	4	2	7	0	FIXED
UECOS4	ECOS IPL AMI	16	4	2	5	0	FIXED
UECOS3	ECOS IPL AMI	16	4	2	3	0	FIXED
UECOS6	ECOS IPL AMI	16	4	3	1	0	FIXED
UECOS8	ECOS IPL AMI	14	4	3	5	0	FIXED
UECOS7	ECOS IPL AMI	16	4	3	3	0	FIXED
SSCDIR	SCOS MMU DIRECTORY	1	0	1	0	0	FIXED
EXCDIR	ECOS DIRECTORY 1ST LEVEL	1	4	3	7	0	FIXED
TEST	ECOS TEST DATA SET	1	4	3	7	1	FIXED
EBOOT	ECOS BOOTSTRAP	2	0	0	4	0	FIXED
EBOOTR	ECOS BOOTSTRAP	2	0	7	3	0	FIXED
SBOOTR	SCOS BOOTSTRAP REDUNDANT	2	0	7	7	0	FIXED
SBOOTP	SCOS BOOTSTRAP PRIME	2	0	0	0	0	FIXED
S1S	ECOS C99	39	0	0	0	0	VAR
S1C	ECOS C99	138	0	0	0	0	VAR
D02A01	ECOS REM	47	0	0	0	0	VAR
A24R0	ECOS C99	1	0	0	0	0	VAR
A34A0	ECOS REM	4	0	0	0	0	VAR
A34E0	ECOS REM	4	0	0	0	0	VAR
MMA	ECOS C99	32	0	0	0	0	VAR
SOP	ECOS C99	10	0	0	0	0	VAR
S99	ECOS C99	18	0	0	0	0	VAR
HRMFMT	SCOS C99 HRM FORMATS	3	0	0	0	0	VAR
A03B0	ECOS REM	1	0	0	0	0	VAR
A34B0	ECOS REM	4	0	0	0	0	VAR
A19E0	ECOS REM	3	0	0	0	0	VAR
S05	ECOS REM	1	0	0	0	0	VAR
A16G0	ECOS REM	4	0	0	0	0	VAR
A20E0	ECOS REM	4	0	0	0	0	VAR
A16A0	ECOS S16	4	0	0	0	0	VAR
T16A	ECOS S16	2	0	0	0	0	VAR
S21	ECOS S21	23	0	0	0	0	VAR
A21A0	ECOS S21	6	0	0	0	0	VAR
T21A	ECOS S21	2	0	0	0	0	VAR
S22	ECOS A22	26	0	0	0	0	VAR
A22A0	ECOS A22	6	0	0	0	0	VAR
T22A	ECOS A22	2	0	0	0	0	VAR
A05A0	ECOS A05	5	0	0	0	0	VAR
T05A	ECOS A05	2	0	0	0	0	VAR
U05TGL	ECOS A05	2	0	0	0	0	VAR
U05PMU	ECOS A05 E13 S21	1	0	0	0	0	VAR
TOFD	ECOS COF	2	0	0	0	0	VAR
AOFDO	ECOS COF	4	0	0	0	0	VAR
S13	ECOS E13	18	0	0	0	0	VAR
A13A0	ECOS E13	12	0	0	0	0	VAR
A13A01	ECOS E13	4	0	0	0	0	VAR
A13A02	ECOS E13	5	0	0	0	0	VAR

B.1 INITIAL DATA-SET LIST INPUT FILE (CONTINUED)

Data-set Name	Description	Size (Blk)	T	F	S	B	Pos.
A13A03	ECOS E13	5	0	0	0	0	VAR
A13A04	ECOS E13	3	0	0	0	0	VAR
A13A06	ECOS E13	4	0	0	0	0	VAR
T13A	ECOS E13	2	0	0	0	0	VAR
U13APC	ECOS E13	1	0	0	0	0	VAR
U13APS	ECOS E13	9	0	0	0	0	VAR
A13G0	ECOS E13	4	0	0	0	0	VAR
T13G	ECOS E13	3	0	0	0	0	VAR
A13A05	ECOS E13	4	0	0	0	0	VAR
S17	ECOS E17	11	0	0	0	0	VAR
A17A0	ECOS E17	6	0	0	0	0	VAR
T17A	ECOS E17	2	0	0	0	0	VAR
U17A01	ECOS E17	8	0	0	0	0	VAR
S33	ECOS E33	44	0	0	0	0	VAR
A33A0	ECOS E33	6	0	0	0	0	VAR
T33A	ECOS E33	2	0	0	0	0	VAR
U33ACC	ECOS E33	250	0	0	0	0	VAR
S34	ECOS E34	15	0	0	0	0	VAR
A34R0	ECOS E34	12	0	0	0	0	VAR
U34ALM	ECOS E34	1	0	0	0	0	VAR
U34AST	ECOS E34	100	0	0	0	0	VAR
TMEM	ECOS C99	2	0	0	0	0	VAR
TEJB	ECOS C99	1	0	0	0	0	VAR
TDPM	ECOS C99	2	0	0	0	0	VAR
TVFI	ECOS C99	1	0	0	0	0	VAR
TPTC	ECOS C99	2	0	0	0	0	VAR
TNBD	ECOS C99	1	0	0	0	0	VAR
TPLS	ECOS C99	2	0	0	0	0	VAR
T27A	ECOS C99	1	0	0	0	0	VAR
XTLM0	ECOS CTL	4	0	0	0	0	VAR
XTLM01	ECOS CTL	2	0	0	0	0	VAR
XTLM02	ECOS CTL	2	0	0	0	0	VAR
TTLM	ECOS CTL	2	0	0	0	0	VAR
XTMN0	ECOS CTM	3	0	0	0	0	VAR
TTMN	ECOS CTM	2	0	0	0	0	VAR
A02A0	ECOS P02	10	0	0	0	0	VAR
A02A02	ECOS P02	6	0	0	0	0	VAR
T02A	ECOS P02	2	0	0	0	0	VAR
T02G	ECOS P02	2	0	0	0	0	VAR
A02A01	ECOS P02	1	0	0	0	0	VAR
T02S	ECOS P02	2	0	0	0	0	VAR
S02	ECOS P02	33	0	0	0	0	VAR
A03A0	ECOS P03	5	0	0	0	0	VAR
T03A	ECOS P03	2	0	0	0	0	VAR
THRZ	ECOS C99 P03	1	0	0	0	0	VAR
AMAGO	ECOS C99 P02 P03 P20	2	0	0	0	0	VAR
A19G0	ECOS P19	5	0	0	0	0	VAR
S19	ECOS P19	1	0	0	0	0	VAR
T19G	ECOS P19	2	0	0	0	0	VAR
A19A0	ECOS P19	6	0	0	0	0	VAR

B.1 INITIAL DATA-SET LIST INPUT FILE (CONTINUED)

Data-set Name	Description	Size (Blk)	T	F	S	B	Pos.
U19A01	ECOS P19	1	0	0	0	0	VAR
T19A	ECOS P19	2	0	0	0	0	VAR
S20	ECOS P20	9	0	0	0	0	VAR
T20A	ECOS P20	2	0	0	0	0	VAR
U20A01	ECOS P20	9	0	0	0	0	VAR
A20A0	ECOS P20	5	0	0	0	0	VAR
A3HA0	ECOS C99	3	0	0	0	0	VAR
T3HA	ECOS C99	2	0	0	0	0	VAR
D300A	ECOS C99	58	0	0	0	0	VAR
A13E0	ECOS REM	4	0	0	0	0	VAR
A13D0	ECOS REM	3	0	0	0	0	VAR
A13C0	ECOS REM	4	0	0	0	0	VAR
T20E	ECOS REM	2	0	0	0	0	VAR
T14A	ECOS REM	1	0	0	0	0	VAR
TVID	ECOS REM	2	0	0	0	0	VAR
TVTR	ECOS REM	2	0	0	0	0	VAR
U16G01	ECOS REM	1	0	0	0	0	VAR
T13E	ECOS REM	2	0	0	0	0	VAR
T19E	ECOS REM	2	0	0	0	0	VAR
D03A01	ECOS REM	76	0	0	0	0	VAR
T34E	ECOS REM	2	0	0	0	0	VAR
T16G	ECOS REM	2	0	0	0	0	VAR
T34A	ECOS REM	2	0	0	0	0	VAR
T13D	ECOS REM	2	0	0	0	0	VAR
T13C	ECOS REM	2	0	0	0	0	VAR
T34B	ECOS REM	2	0	0	0	0	VAR
TITM	ECOS REM	1	0	0	0	0	VAR
TVR6	ECOS REM	1	0	0	0	0	VAR
S24	ECOS REM	8	0	0	0	0	VAR
VER101	ECOS REM	5	0	0	0	0	VAR
D300C	ECOS REM	58	0	0	0	0	VAR
F1	ECOS REM	2	0	0	0	0	VAR
TCDT	ECOS REM	1	0	0	0	0	VAR
TVR7	ECOS REM	1	0	0	0	0	VAR
TVR3	ECOS REM	1	0	0	0	0	VAR
VER102	ECOS REM	1	0	0	0	0	VAR
XBUG0	ECOS REM	4	0	0	0	0	VAR
VER103	ECOS REM	1	0	0	0	0	VAR
VER104	ECOS REM	5	0	0	0	0	VAR
TVR2	ECOS REM	2	0	0	0	0	VAR
TVRB	ECOS REM	1	0	0	0	0	VAR
TXB2	ECOS REM	1	0	0	0	0	VAR
TVR9	ECOS REM	1	0	0	0	0	VAR
TVR8	ECOS REM	1	0	0	0	0	VAR
TVR4	ECOS REM	1	0	0	0	0	VAR
VER10	ECOS REM	9	0	0	0	0	VAR
TXB1	ECOS REM	1	0	0	0	0	VAR
TXB1	ECOS REM	1	0	0	0	0	VAR
TVRA	ECOS REM	1	0	0	0	0	VAR
TVR1	ECOS REM	2	0	0	0	0	VAR

B.1 INITIAL DATA-SET LIST INPUT FILE (CONTINUED)

Data-set Name	Description	Size (Blk)	T	F	S	B	Pos.
TAC1	ECOS REM	1	0	0	0	0	VAR
TDEP	ECOS REM	2	0	0	0	0	VAR
TAC2	ECOS REM	1	0	0	0	0	VAR
TGRP	ECOS REM	2	0	0	0	0	VAR
S32	ECOS REM	4	0	0	0	0	VAR
TGMC	ECOS REM	2	0	0	0	0	VAR
S31	ECOS REM	8	0	0	0	0	VAR
QRTN	ECOS REM	5	0	0	0	0	VAR
FP	ECOS REM	1	0	0	0	0	VAR
QRTN	ECOS REM	5	0	0	0	0	VAR
S08	ECOS REM	16	0	0	0	0	VAR
TAPP	ECOS REM	3	0	0	0	0	VAR
EPP10	ECOS REM	3	0	0	0	0	VAR
TFC3	ECOS REM	1	0	0	0	0	VAR
D300B	ECOS REM	58	0	0	0	0	VAR
FED1	ECOS REM	18	0	0	0	0	VAR
TXB2	ECOS REM	1	0	0	0	0	VAR
TVR5	ECOS REM	1	0	0	0	0	VAR
FCDP	ECOS REM	3	0	0	0	0	VAR
MM1	ECOS REM	56	0	0	0	0	VAR
TADO	ECOS REM	2	0	0	0	0	VAR
THPS	ECOS REM	1	0	0	0	0	VAR
TMA	SCOS FC02, DISPLAY	3	0	0	0	0	VAR
MCON02	SCOS FC02, MC02 DIRECTORY	1	0	0	0	0	VAR
TFFD00	SCOS FC02, G-AMI	2	0	0	0	0	VAR
MCON01	SCOS FC01 DIRECTORY	1	0	0	0	0	VAR
FFDSSC	SCOS FC02 DIRECTORY	1	0	0	0	0	VAR
FCDRIV	SCOS FC02, G-AMI	1	0	0	0	0	VAR
SRW	SCOS FC02, DISPLAY	1	0	0	0	0	VAR
MMJINI	SCOS FC01 DIRECTORY	1	0	0	0	0	VAR
AUD	SCOS FC02, DISPLAY	2	0	0	0	0	VAR
MC1AMIS	SCOS FC01, MC01 AMI'S	35	0	0	0	0	VAR
	END						

B.2 OPTIMIZED DATA-SET LIST INPUT FILE

Data-set Name	Description	Size (Blk)	T	F	S	B	Pos.
SCOSAM	SCOS IPL AMI	126	0	0	0	2	FIXED
ECOSAM	ECOS IPL AMI	6	4	1	7	0	FIXED
UECOS2	ECOS IPL AMI	16	4	2	1	0	FIXED
UECOS5	ECOS IPL AMI	16	4	2	7	0	FIXED
UECOS4	ECOS IPL AMI	16	4	2	5	0	FIXED
UECOS3	ECOS IPL AMI	16	4	2	3	0	FIXED
UECOS6	ECOS IPL AMI	16	4	3	1	0	FIXED
UECOS8	ECOS IPL AMI	14	4	3	5	0	FIXED
UECOS7	ECOS IPL AMI	16	4	3	3	0	FIXED
SSCDIR	SCOS MMU DIRECTORY	1	0	1	0	0	FIXED
EXCDIR	ECOS DIRECTORY 1ST LEVEL	1	4	3	7	0	FIXED
TEST	ECOS TEST DATA SET	1	4	3	7	1	FIXED
EBOOT	ECOS BOOTSTRAP	2	0	0	4	0	FIXED
EBOOTR	ECOS BOOTSTRAP	2	0	7	3	0	FIXED
SBOOTR	SCOS BOOTSTRAP REDUNDANT	2	0	7	7	0	FIXED
SBOOTP	SCOS BOOTSTRAP PRIM	2	0	0	0	0	FIXED
S1S	ECOS C99	39	0	0	0	0	VAR
S1C	ECOS C99	138	0	0	0	0	VAR
MMA	ECOS C99	32	0	0	0	0	VAR
SOP	ECOS C99	10	0	0	0	0	VAR
S99	ECOS C99	18	0	0	0	0	VAR
A34B0	ECOS REM	4	0	0	0	0	VAR
A34A0	ECOS REM	4	0	0	0	0	VAR
A34E0	ECOS REM	4	0	0	0	0	VAR
A03B0	ECOS REM	1	0	0	0	0	VAR
A24R0	ECOS C99	1	0	0	0	0	VAR
MCON02	SCOS FC02, MC02 DIRECTORY	1	0	0	0	0	VAR
F1	ECOS REM	2	0	0	0	0	VAR
S05	ECOS REM	1	0	0	0	0	VAR
XTLM0	ECOS CTL	4	0	0	0	0	VAR
XTLM01	ECOS CTL	2	0	0	0	0	VAR
XTLM02	ECOS CTL	2	0	0	0	0	VAR
TTLM	ECOS CTL	2	0	0	0	0	VAR
XTMN0	ECOS CTM	3	0	0	0	0	VAR
TTMN	ECOS CTM	2	0	0	0	0	VAR
S33	ECOS E33	44	0	0	0	0	VAR
A33A0	ECOS E33	6	0	0	0	0	VAR
T33A	ECOS E33	2	0	0	0	0	VAR
U33ACC	ECOS E33	250	0	0	0	0	VAR
S17	ECOS E17	11	0	0	0	0	VAR
A17A0	ECOS E17	6	0	0	0	0	VAR
T17A	ECOS E17	2	0	0	0	0	VAR
U17A01	ECOS E17	8	0	0	0	0	VAR
S34	ECOS E34	15	0	0	0	0	VAR
A34R0	ECOS E34	12	0	0	0	0	VAR
U34ALM	ECOS E34	1	0	0	0	0	VAR
U34AST	ECOS E34	100	0	0	0	0	VAR
S13	ECOS E13	18	0	0	0	0	VAR
A13A0	ECOS E13	12	0	0	0	0	VAR
A13A01	ECOS E13	4	0	0	0	0	VAR

B.2 OPTIMIZED DATA-SET LIST INPUT FILE (CONTINUED)

Data-set Name	Description	Size (Blk)	T	F	S	B	Pos.
A13A02	ECOS E13	5	0	0	0	0	VAR
A13A03	ECOS E13	5	0	0	0	0	VAR
A13A04	ECOS E13	3	0	0	0	0	VAR
A13A06	ECOS E13	4	0	0	0	0	VAR
T13A	ECOS E13	2	0	0	0	0	VAR
U13APC	ECOS E13	1	0	0	0	0	VAR
U13APS	ECOS E13	9	0	0	0	0	VAR
A13G0	ECOS E13	4	0	0	0	0	VAR
T13G	ECOS E13	3	0	0	0	0	VAR
A13A05	ECOS E13	4	0	0	0	0	VAR
TOFD	ECOS COF	2	0	0	0	0	VAR
AOFD0	ECOS COF	4	0	0	0	0	VAR
S22	ECOS A22	26	0	0	0	0	VAR
A22A0	ECOS A22	6	0	0	0	0	VAR
T22A	ECOS A22	2	0	0	0	0	VAR
A05A0	ECOS A05	5	0	0	0	0	VAR
T05A	ECOS A05	2	0	0	0	0	VAR
U05TGL	ECOS A05	2	0	0	0	0	VAR
S21	ECOS S21	23	0	0	0	0	VAR
A21A0	ECOS S21	6	0	0	0	0	VAR
T21A	ECOS S21	2	0	0	0	0	VAR
U05PMU	ECOS A05 E13 S21	1	0	0	0	0	VAR
A16A0	ECOS S16	4	0	0	0	0	VAR
T16A	ECOS S16	2	0	0	0	0	VAR
HRMFMT	SCOS C99 HRM FORMATS	3	0	0	0	0	VAR
TMEM	ECOS C99	2	0	0	0	0	VAR
TEJB	ECOS C99	1	0	0	0	0	VAR
TDPM	ECOS C99	2	0	0	0	0	VAR
TVFI	ECOS C99	1	0	0	0	0	VAR
TPTC	ECOS C99	2	0	0	0	0	VAR
TNBD	ECOS C99	1	0	0	0	0	VAR
TPLS	ECOS C99	2	0	0	0	0	VAR
T27A	ECOS C99	1	0	0	0	0	VAR
A19G0	ECOS P19	5	0	0	0	0	VAR
S19	ECOS P19	1	0	0	0	0	VAR
T19G	ECOS P19	2	0	0	0	0	VAR
A19A0	ECOS P19	6	0	0	0	0	VAR
U19A01	ECOS P19	1	0	0	0	0	VAR
T19A	ECOS P19	2	0	0	0	0	VAR
S20	ECOS P20	9	0	0	0	0	VAR
T20A	ECOS P20	2	0	0	0	0	VAR
U20A01	ECOS P20	9	0	0	0	0	VAR
A20A0	ECOS P20	5	0	0	0	0	VAR
THRZ	ECOS C99 P03	1	0	0	0	0	VAR
AMAG0	ECOS C99 P02 P03 P20	2	0	0	0	0	VAR
A03A0	ECOS P03	5	0	0	0	0	VAR
T03A	ECOS P03	2	0	0	0	0	VAR
S02	ECOS P02	33	0	0	0	0	VAR
A02A0	ECOS P02	10	0	0	0	0	VAR
A02A01	ECOS P02	1	0	0	0	0	VAR

B.2 OPTIMIZED DATA-SET LIST INPUT FILE (CONTINUED)

Data-set Name	Description	Size (Blk)	T	F	S	B	Pos.
A02A02	ECOS P02	6	0	0	0	0	VAR
T02A	ECOS P02	2	0	0	0	0	VAR
T02G	ECOS P02	2	0	0	0	0	VAR
T02S	ECOS P02	2	0	0	0	0	VAR
A3HA0	ECOS C99	3	0	0	0	0	VAR
T3HA	ECOS C99	2	0	0	0	0	VAR
D300A	ECOS C99	58	0	0	0	0	VAR
A19E0	ECOS REM	3	0	0	0	0	VAR
T14A	ECOS REM	1	0	0	0	0	VAR
TVID	ECOS REM	2	0	0	0	0	VAR
TVTR	ECOS REM	2	0	0	0	0	VAR
D02A01	ECOS REM	47	0	0	0	0	VAR
A13E0	ECOS REM	4	0	0	0	0	VAR
A13D0	ECOS REM	3	0	0	0	0	VAR
A13C0	ECOS REM	4	0	0	0	0	VAR
A16G0	ECOS REM	4	0	0	0	0	VAR
A20E0	ECOS REM	4	0	0	0	0	VAR
T20E	ECOS REM	2	0	0	0	0	VAR
U16G01	ECOS REM	1	0	0	0	0	VAR
T13E	ECOS REM	2	0	0	0	0	VAR
T19E	ECOS REM	2	0	0	0	0	VAR
D03A01	ECOS REM	76	0	0	0	0	VAR
T34E	ECOS REM	2	0	0	0	0	VAR
T16G	ECOS REM	2	0	0	0	0	VAR
T34A	ECOS REM	2	0	0	0	0	VAR
T13D	ECOS REM	2	0	0	0	0	VAR
T13C	ECOS REM	2	0	0	0	0	VAR
T34B	ECOS REM	2	0	0	0	0	VAR
TITM	ECOS REM	1	0	0	0	0	VAR
TVR6	ECOS REM	1	0	0	0	0	VAR
S24	ECOS REM	8	0	0	0	0	VAR
VER101	ECOS REM	5	0	0	0	0	VAR
D300C	ECOS REM	58	0	0	0	0	VAR
TCDT	ECOS REM	1	0	0	0	0	VAR
TVR7	ECOS REM	1	0	0	0	0	VAR
TVR3	ECOS REM	1	0	0	0	0	VAR
VER102	ECOS REM	1	0	0	0	0	VAR
XBUG0	ECOS REM	4	0	0	0	0	VAR
VER103	ECOS REM	1	0	0	0	0	VAR
VER104	ECOS REM	5	0	0	0	0	VAR
TVR2	ECOS REM	2	0	0	0	0	VAR
TVRB	ECOS REM	1	0	0	0	0	VAR
TXB2	ECOS REM	1	0	0	0	0	VAR
TVR9	ECOS REM	1	0	0	0	0	VAR
TVR8	ECOS REM	1	0	0	0	0	VAR
TVR4	ECOS REM	1	0	0	0	0	VAR
VER10	ECOS REM	9	0	0	0	0	VAR
TXB1	ECOS REM	1	0	0	0	0	VAR
TXB1	ECOS REM	1	0	0	0	0	VAR
TVRA	ECOS REM	1	0	0	0	0	VAR

B.2 OPTIMIZED DATA-SET LIST INPUT FILE (CONTINUED)

Data-set Name	Description	Size (Blk)	T	F	S	B	Pos.
--------	-------------------------	---	-	-	-	--	-----
TVR1	ECOS REM	2	0	0	0	0	VAR
TAC1	ECOS REM	1	0	0	0	0	VAR
TDEP	ECOS REM	2	0	0	0	0	VAR
TAC2	ECOS REM	1	0	0	0	0	VAR
TGRP	ECOS REM	2	0	0	0	0	VAR
S32	ECOS REM	4	0	0	0	0	VAR
TGMC	ECOS REM	2	0	0	0	0	VAR
S31	ECOS REM	8	0	0	0	0	VAR
QRTN	ECOS REM	5	0	0	0	0	VAR
FP	ECOS REM	1	0	0	0	0	VAR
QRTN	ECOS REM	5	0	0	0	0	VAR
S08	ECOS REM	16	0	0	0	0	VAR
TAPP	ECOS REM	3	0	0	0	0	VAR
EPP10	ECOS REM	3	0	0	0	0	VAR
TFC3	ECOS REM	1	0	0	0	0	VAR
D300B	ECOS REM	58	0	0	0	0	VAR
FED1	ECOS REM	18	0	0	0	0	VAR
TXB2	ECOS REM	1	0	0	0	0	VAR
TVR5	ECOS REM	1	0	0	0	0	VAR
FCDP	ECOS REM	3	0	0	0	0	VAR
MM1	ECOS REM	56	0	0	0	0	VAR
TADO	ECOS REM	2	0	0	0	0	VAR
THPS	ECOS REM	1	0	0	0	0	VAR
TMA	SCOS FC02, DISPLAY	3	0	0	0	0	VAR
TFFD00	SCOS FC02, G-AMI	2	0	0	0	0	VAR
MCON01	SCOS FC01 DIRECTORY	1	0	0	0	0	VAR
FFDSSC	SCOS FC02 DIRECTORY	1	0	0	0	0	VAR
FCDRIV	SCOS FC02, G-AMI	1	0	0	0	0	VAR
SRW	SCOS FC02, DISPLAY	1	0	0	0	0	VAR
MMUINI	SCOS FC01 DIRECTORY	1	0	0	0	0	VAR
AUD	SCOS FC02, DISPLAY	2	0	0	0	0	VAR
MC1AMIS	SCOS FC01, MC01 AMI'S	35	0	0	0	0	VAR
	END						

B.3 DATA-SET TO MISSION TIMELINE CORRELATION DATA

The data file shown in this section defines the sequence of data-set accesses associated with individual scheduled operations. Each scheduled operation that will have data-set accesses should have an input record. Each record must contain the flight operation label and step number delimited by a comma, left justified in characters 1 - 12. The data-set names to be called during the step must be left justified and begin in positions 12, 22, 32, ..., 152. The order of the input records is not important.

3.3 DATA-SET TO MISSION TIMELINE CORRELATION DATA

```
1234567890I 123456789 123456789 123456789 123456789 123456789 123456789 123456789 123456789
ICCM,6      TYPI
M1FO-1,1    S1S       S1C
M1FO-1,4    S1C
M1FO-1,7    S1C
M1FO-1,9    S1C
M1FO-1,11   S1C
M1FO7B2,2   S1S       S1C
M1FO7B2,4   S1C
M1FO7B2,6   S1C
M1FO7B2,8   S1C
M1FO7B2,10  S1C
M1FO7B3,2   S1S       S1C
M1FO7B3,4   S1C
M1FO7B3,6   S1C
M1FO7B3,8   S1C
M1FO7B3,10  S1C
M1FO7B34,2  S1S       S1C
M1FO7B34,4  S1C
M1FO7B34,5  S1C
M1FO7B34,8  S1C
M1FO7B34,10 S1C
M1FO7B34,12 S1C
M1F7A56A,2  S1S       S1C
M1F7A56A,4  S1C
M1F7A56A,6  S1C
M1F7A56A,8  S1C
M1F7A56A,10 S1C
M1F7A56A,12 S1C
M1F7A56A,14 S1C
M1F7A56A,16 S1C
M1F7A56A,18 S1C
M1F7A56A,20 S1C
M1F7A56A,22 S1C
M1F7A56A,24 S1C
M1F7A56A,26 S1C
M1F7A56A,28 S1C
M1F7A56B,2  S1S       S1C
M1F7A56B,4  S1C
M1F7A56B,6  S1C
M1F7A56B,8  S1C
M1F7A56B,10 S1C
M1F7A56B,12 S1C
M1F7A56B,14 S1C
M1F7A56B,16 S1C
M1F7A56B,18 S1C
M1F7A56B,20 S1C
M1F7A56B,22 S1C
M1FO7A6A,2  S1S       S1C
M1FO7A6A,4  S1C
M1FO7A6A,6  S1C
M1FO7A6A,8  S1C
M1FO7A6A,10 S1C
M1FO7A6B,2  S1S       S1C
M1FO7A6B,4  S1C
M1FO7A6B,6  S1C
M1FO7A6B,8  S1C
M1FO7A6B,10 S1C
M1F7A6BA,2  S1S       S1C
M1F7A6BA,4  S1C
M1F7A6BA,6  S1C
```

3.3 DATA-SET TO MISSION TIMELINE CORRELATION DATA

M1F7A68A,8	S1C							
M1F7A68A,10	S1C							
M1F7A68A,12	S1C							
M1F7A68A,14	S1C							
M1F7A68A,16	S1C							
M1F7A68A,18	S1C							
M1F7A68B,2	S1B	S1C						
M1F7A68B,4	S1C							
M1F7A68B,6	S1C							
M1F7A68B,8	S1C							
M1F7A68B,10	S1C							
M1F7A68B,12	S1C							
M1F7A68B,14	S1C							
M1F7A68B,16	S1C							
M1F0-9,1	S1B	S1C						
M1F0-9,3	S1C							
M1F0-9,5	S1C							
M1F0-10,1	S1B	S1C						
M1F0-10,3	S1C							
M1F0-10,5	S1C							
M1F0-13,1	S1B	S1C						
M1F0-13,4	S1C							
M1F0-13,6	S1C							
M1F0-14,1	S1B	S1C						
M1F0-14,4	S1C							
M1F0-14,6	S1C							
M1F0-15,1	S1B	S1C						
M1F0-15,3	S1C							
M1F0-15,5	S1C							
M1F0-15,7	S1B	S1C						
M2T0F01,1	S02	A02A0	A02A01	A02A02	T02A	T02G		
M2T0F01,2	T02B							
M2T0F01,4	S02							
M2T1F02,1	S02	A02A0	A02A01	A02A02	T02A	T02G		
M2T1F02,2	T02B							
M2T1F02,10	S02							
M2T2F03,1	S02	A02A0	A02A01	A02A02	T02A	T02G	AMAG0	
M2T2F03,2	T02B							
M2T2F03,6	S02							
M2T3F04,1	S02	A02A0	A02A01	A02A02	T02A	T02G		
M2T3F04,2	T02B							
M2T3F04,3	S19	A19A0	T19A	U19A01				
M2T3F04,6	S02							
M2F5AB67,1	S02	A02A0	A02A01	A02A02	T02A	T02G		
M2F5AB67,2	T02B							
M2F5AB67,3	S19	A19A0	T19A	U19A01				
M2F5AB67,13	S02							
M2F09/8A,1	S02	A02A0	A02A01	A02A02	T02A	T02G	A03A0	T03A
M2F09/8A,2	T02B							
M2F09/8A,11	S02							
M2F08/9B,1	S02	A02A0	A02A01	A02A02	T02A	T02G	A03A0	T03A
M2F08/9B,2	T02B							
M2F08/9B,10	S02							
M2F09/8C,1	S02	A02A0	A02A01	A02A02	T02A	T02G	A03A0	T03A
M2F09/8C,2	T02B							
M2F09/8C,10	S02							
M2F09A1,1	S02	A02A0	A02A01	A02A02	T02A	T02G	A03A0	T03A
M2F09A1,2	T02B							
M2F09A1,11	S02							
M2F09A2,1	S02	A02A0	A02A01	A02A02	T02A	T02G	A03A0	T03A
M2F09A2,2	T02B							

B.3 DATA-SET TO MISSION TIMELINE CORRELATION DATA

A2F09A2,5	S02									
M2F9B/C1,1	S02	A02A0	A02A01	A02A02	T02A	T02C	A03A0	T03A		
M2F9B/C1,2	T02S									
M2F9B/C1,9	S02									
M2F9B/C2,1	S02	A02A0	A02A01	A02A02	T02A	T02C	A03A0	T03A		
M2F9B/C2,2	T02S									
M2F9B/C2,9	S02									
M2F9B/C3,1	S02	A02AC	A02A01	A02A02	T02A	T02C	A03A0	T03A		
M2F9B/C3,2	T02S									
M2F9B/C3,9	S02									
M2F10/12,1	S02	A02A0	A02A01	A02A02	T02A	T02C	A03A0	T03A		
M2F10/12,2	T02B									
M2F10/12,10	S02									
M2F11/12,1	S02	A02A0	A02A01	A02A02	T02A	T02C	A03A0	T03A		
M2F11/12,2	T02S									
M2F11/12,9	S02									
M2F013A,1	S02	A02A0	A02A01	A02A02	T02A	T02C				
M2F013A,2	T02S									
M2F013A,4	S02									
M2F014,1	S02	A02A0	A02A01	A02A02	T02A	T02C	A03A0	T03A	AMACO	
M2F014,2	S20	A20A0	T20A	U20A01						
M2F014,5	S02									
A2F14/9A,1	S02	A02A0	A02A01	A02A02	T02A	T02C	A03A0	T03A	A03A0	T03A
M2F14/9A,2	T02A									
M2F14/9A,4	S20	A20A0	T20A	U20A01						
M2F14/9A,6	A03A0	T03A								
M2F14/9A,10	S02									
AMAG,1	SOP	AMAGO								
AMAG,2	SOP	AMAGO								
APAG,1	SOP	APAGO								
M3DCFOT1,2	A03A0	T03A								
M3MCFOT2,2	A03A0	T03A	AMACO							
M3DUPOT3,2	A03A0	T03A								
M3FOTS/6,2	A03A0	T03A	THRI							
M3FO-1B,1	A03A0	T03A								
M3FO-4,1	A03A0	T03A								
M3FO-2E,1	A03A0	T03A								
M3FO-2E,4	S20	A20A0	T20A	U20A01						
M3FO-3F,1	A03A0	T03A								
M3FO-3F,4	S20	A20A0	T20A	U20A01						
S05-F01A,1	A05A0	T05A	U05PMU	U05TGL						
S05-F01C,1	S1B	S1C								
S05-F01C,3	A05A0	T05A	U05PMU	U05TGL	A22A0	T22A	S22	S1C		
S05-F01C,7	S1C									
S13-F01,1	A13A0	A13A01	A13A02	A13A03	A13A04	A13A06	T13A	U05PMU	U13APC	U13APB
S13-F01,2	A13C0	T13C								
S13-F01,6	A13A03									
S13-F02,1	S13									
S13-F02,2	A13A0	A13A01	A13A02	A13A03	A13A04	A13A06	T13A	U05PMU	U13APC	U13APB
S13-F02,4	S13									
S13-F02,6	S13									
S13-F02,8	S13									
S13-F2/3,1	S13									
S13-F2/3,2	A13A0	A13A01	A13A02	A13A03	A13A04	A13A06	T13A	U05PMU	U13APC	U13APB
S13-F2/3,4	S13									
S13-F2/3,6	S13									
S13-F2/3,8	S13									
S13-F2/3,12	S13									
S13-F2/3,13	S13	A13A03								
S13-F2/3,14	S13									
S13-F04,1	S13									

B.3 DATA-SET TO MISSION TIMELINE CORRELATION DATA

013-F04,5	S13								
014-F01,1	S99								
014-F01,4	S99								
014-F02,1	S99								
014-F02,4	S99								
016-F01,1	S99								
016-F01,4	S99								
016-F02,1	S99								
016-F01,1	S99								
016-F03,7	A16AD	T16A							
017-F01A,1	S17								
017-F01A,2	A17AO	T17A	U17A01						
017-F01A,3	U17A01								
017-F01B,1	S17								
017-F01B,2	A17AO	T17A	U17A01						
017-F01B,3	U17A01								
017-F02,1	S17								
017-F02,2	A17AO	T17A	U17A01						
017-F02,3	U17A01								
017-F03,1	S17								
017-F03,2	A17AO	T17A	U17A01						
017-F03,3	U17A01								
019-F1/2,2	S19	A19AO	T19A	U19A01					
019-F03A,1	S19	A19AO	T19A	U19A01	A19CO	T19C			
019-F03B,1	S19	A19AO	T19A	U19A01	A19CO	T19C			
019-F04,2	S19	A19AO	T19A	U19A01	A19CO	T19C	A03AO	T03A	
020-F01A,3	A20AO	T20A	U20A01						
020-F01A,5	A20AO	T20A	U20A01						
020-F01B,2	A20AO	T20A	U20A01						
020-F02,1	S20	A20AO	T20A	U20A01	S19	A19AO	T19A	U19A01	AMACO
020-F02A,1	S20	A20AO	T20A	U20A01	AMACO				
020-F04,1	S20	A20AO	T20A	U20A01	S19	A19AO	T19A	U19A01	AMACO
020-F05A,1	S20	A20AO	T20A	U20A01	S19	A19AO	T19A	U19A01	AMACO
020-F05B,1	S20	A20AO	T20A	U20A01	AMACO				
021-F1/2,1	S21	A21AO	T21A	U05PMU					
021-F1/2,2	S21								
022-F1/3,2	A22AO	T22A							
022-F1/3,12	A22AO	T22A							
022-F1/3,16	A22AO	T22A							
022-F02,1	A22AO	T22A	A05AO	T05A	U05PMU	U05TGL			
022-F02,2	S22								
022-F02C,1	A22AO	T22A	A05AO	T05A	U05PMU	U05TGL			
022-F02C,2	S22								
022-F02K,1	A22AO	T22A	A05AO	T05A	U05PMU	U05TGL			
022-F02K,2	S22								
022-F02M,1	A22AO	T22A	A05AO	T05A	U05PMU	U05TGL			
022-F02M,2	S22								
023-F01,2	S99								
023-F01C,2	S99								
023-F02,3	S99								
024-F00,1	S99	A24RO							
027-F01,1	S99								
027-F01,3	S99	T27A							
029-F01,1	S99								
029-F01,5	S99								
033-F1/4,2	A33AO	T33A	U33ACC						
033-F03,1	S33	A33AO	T33A	U33ACC					
033-F03,3	S33								
033-F05,2	A33AO	T33A	U33ACC						
034-FC3,1	S34	A34RO	U34ALW						
034-FC3,3	U34AR1								

B.3 DATA-SET TO MISSION TIMELINE CORRELATION DATA

034-FC3,5	U34AST		
034-FC3,7	U34AST		
034-FC4,1	B34	A34RO	U34ALM
034-FC4,3	U34AST		
034-FC4,5	U34AST		
034-FC4,7	U34AST		
034-FC6,1	B34	A34RO	U34ALM
034-FC6,3	U34AST		
034-FC6,5	U34AST		
034-FC7,1	B34	A34RO	U34ALM
034-FC7,3	U34AST		
034-FC7,5	U34AST		
034-FC8,1	B34	A34RO	U34ALM
034-FC8,3	U34AST		
034-FC8,5	U34AST		
034-FC9,1	B34	A34RO	U34ALM
034-FC9,3	U34AST		
034-FC9,5	U34AST		
034-FS6,4	B34	A34RO	U34ALM
034-FS6,7	U34AST		
034-FS6P,4	B34	A34RO	U34ALM
034-FS6P,7	U34AST		
034-FS6P,9	U34AST		
034-FS6P,11	U34AST		
300-F00,1	A3MAO	T3NA	D300A
BL-ACT,12	TVF1		
BL-DACT,2	TVF1		
ATT-1,13	THRZ		
ATT-1,26	THRZ		
ATT-3,2	THRZ		
ATT-3,7	THRZ		
ATT-3,14	THRZ		

B.4 UNSCHEDULED DATA-SETS UTILIZATION DATA

This input file defines the expected unscheduled operations. For each operation, the number of times it is expected to occur, the time period in which it expected, and the data-sets accessed during the operation are defined. Up to fifteen accesses may be defined to occur in one unscheduled operation.

No. of Perf.	Start (Hrs)	End (Hrs)	Data-sets			
155	4.0	159.0	TMEM			
44	4.0	159.0	MMA			
7	4.0	159.0	TDPM			
7	4.0	159.0	XTLMO	XTLM01	XTLM02	TTLM
14	4.0	159.0	XTMNO	TTMN		
28	4.0	159.0	AOFDO	TOFD	U05PMU	
7	4.0	159.0	TPLS			
7	4.0	159.0	TPTC			
7	4.0	159.0	TEJB			
7	4.0	159.0	TNBD			
15	4.0	159.0	U05PMU			
14	24.0	144.0	S1S			
14	24.0	144.0	S1C			
14	24.0	144.0	S13			
14	24.0	144.0	S33			
14	24.0	144.0	S34			
14	24.0	144.0	S17			
14	24.0	144.0	S20			
14	24.0	144.0	S99			
14	24.0	144.0	U13APS			
14	24.0	144.0	U17A01			
14	24.0	144.0	U20A01			
14	12.0	144.0	U33ACC			
14	12.0	144.0	U34AST			

B.5 RANDOM NUMBER SEEDS FILE

This input file defines the random number seed for each simulation run. The number of simulation runs is also determined by the number of seeds included in the file. The file is named SEEDS.DAT.

The random numbers are generated by a subroutine that provides a uniformly distributed set of values. This subroutine uses the VAX 11/780 computer provided random number generator. This generator uses the multiplicative congruent method for the number generation [25]. Each seed value must be a large odd integer number.

To assure the random number streams are consistant from one layout proposal to another, the same seed values were used on each replication. The seed values below were used for each layout proposal.

Random Number Seeds File

```
                   1         2
COLUMN= 12345678901234567890
        --------------------
SEED 1= 87654321
SEED 2= 99335427
SEED 3= 85736459
SEED 4= 79827411
SEED 5= 39475893
STOP  = 00000000
```

APPENDIX C

SIMULATION REPORTS

This appendix shows examples of the reports generated by the computer program. The first three reports indicate the how the MMU tape has been laid out for the list of data-sets. These reports are in C.1, C.2, and C.3. The remaining reports are associated with the simulation of the data-set accesses using the tape layout. They are in C.4 and C.5.

C.1 LISTING OF THE MMJ DATA-SETS

ECOSAM	SCOS IPL AMI	126	0	0	0	2	FIXED
ECOSAM	ECOS IPL AMI	6	4	1	7	0	FIXED
UECOS2	ECOS IPL AMI	16	4	2	1	0	FIXED
UECOS5	ECOS IPL AMI	15	4	2	7	0	FIXED
UECOS4	ECOS IPL AMI	16	4	2	5	0	FIXED
UECOS3	ECOS IPL AMI	16	4	2	3	0	FIXED
UECOS6	ECOS IPL AMI	16	4	3	1	0	FIXED
UECOS8	ECOS IPL AMI	14	4	3	5	0	FIXED
UECOS7	ECOS IPL AMI	16	4	3	3	0	FIXED
SSCDIR	SCOS MMU DIRECTORY	1	0	1	0	0	FIXED
ESCDIR	ECOS DIRECTORY 1ST LEVEL	1	4	3	7	0	FIXED
TEST	ECOS TEST DATA SET	1	4	3	7	1	FIXED
EBOOT	ECOS BOOTSTRAP	2	0	0	4	0	FIXED
EBOOTP	ECOS BOOTSTRAP	2	0	7	3	0	FIXED
SBOOTR	SCOS BOOTSTRAP REDUNDANT	2	0	7	7	0	FIXED
SBOOTP	SCOS BOOTSTRAP PRIME	2	0	0	0	0	FIXED
S15	ECOS C99	39	0	0	0	0	VAR
S1C	ECOS C99	138	0	0	0	0	VAR
MMA	ECOS C99	32	0	0	0	0	VAR
SOP	ECOS C99	10	0	0	0	0	VAR
S99	ECOS C99	18	0	0	0	0	VAR
A349C	ECOS REM	4	0	0	0	0	VAR
A344C	ECOS REM	4	0	0	0	0	VAR
A34FC	ECOS REM	4	0	0	0	0	VAR
AC3EC	ECOS REM	1	0	0	0	0	VAR
A24FC	ECOS C99	1	0	0	0	0	VAR
MCONO2	SCOS FC02, MC02 DIRECTORY	1	0	0	0	0	VAR
F1	FCOS REM	2	0	0	0	0	VAR
S05	FCOS REM	1	0	0	0	0	VAR
XTLMO	FCOS CTL	4	0	0	0	0	VAR
XTLM01	ECOS CTL	2	0	0	0	0	VAR
XTLM02	ECOS CTL	2	0	0	0	0	VAR
TTLM	ECOS CTL	2	0	0	0	0	VAR
XTMMO	ECOS CTM	3	0	0	0	0	VAR
TTMM	ECOS CTM	2	0	0	0	0	VAR
S33	ECOS E33	44	0	0	0	0	VAR
A33AO	ECOS E33	6	0	0	0	0	VAR
T33A	ECOS E33	2	0	0	0	0	VAR
U33ACC	ECOS E33	250	0	0	0	0	VAR
S17	FCOS E17	11	0	0	0	0	VAR
A17AO	ECOS E17	6	0	0	0	0	VAR
T17A	ECOS E17	2	0	0	0	0	VAR
U17A01	ECOS E17	6	0	0	0	0	VAR
S34	ECOS E34	15	0	0	0	0	VAR
A34PC	ECOS E34	12	0	0	0	0	VAR
U34ALM	FCOS E34	1	0	0	0	0	VAR
U34AST	FCOS E34	100	0	0	0	0	VAR
S13	FCOS E13	18	0	0	0	0	VAR
A13AS	FCOS E13	12	0	0	0	0	VAR
A13A01	ECOS E13	4	0	0	0	0	VAR
A13A02	ECOS E13	5	0	0	0	0	VAR
A13A03	FCOS E13	5	0	0	0	0	VAR
A13A04	FCOS E13	1	0	0	0	0	VAR
A13A06	FCOS E13	4	0	0	0	0	VAR
T13A	FCOS E13	2	0	0	0	0	VAR
U13APC	FCOS E13	1	0	0	0	0	VAR
U13APS	ECOS E13	9	0	0	0	0	VAR
A13G0	FCOS E13	4	0	0	0	0	VAR
T13G	SCOS E13	3	0	0	0	0	VAR
A13A05	FCOS E13	4	0	0	0	0	VAR
TOFO	FCOS CDT	2	0	0	0	0	VAR

C.2 ALLOCATED DIRECTORY

ALLOCATED DIRECTORY

ECOSAM	ECOS IPL AMI	136	0	0	0	2	FIXED
ECOSAP	ECOS IPL AMI	6	4	1	7	0	FIXED
UECOS2	ECOS IPL AMI	16	4	2	1	0	FIXED
UECOS5	ECOS IPL AMI	16	4	3	7	0	FIXED
UECOS4	ECOS IPL AMI	16	4	2	5	0	FIXED
UECOS3	ECOS IPL AMI	16	4	2	3	0	FIXED
UECOS6	ECOS IPL AMI	16	4	3	1	0	FIXED
UECOS8	ECOS IPL AMI	14	4	3	5	0	FIXED
UECOS7	ECOS IPL AMI	16	4	3	3	0	FIXED
SECDIR	SCOS MPU DIRECTORY	1	0	1	0	0	FIXED
EECDIR	FCOS DIRECTORY 1ST LEVEL	1	4	3	7	0	FIXED
TEST	FCOS TEST DATA SET	1	4	3	7	1	FIXED
EBOOT	ECOS BOOTSTRAP	2	0	0	4	0	FIXED
EBOOTR	ECOS BOOTSTRAP	2	0	7	3	0	FIXED
SBOOTR	SCOS BOOTSTRAP REDUNDANT	2	0	7	7	0	FIXED
SBOOTP	SCOS BOOTSTRAP PRIME	2	0	0	0	0	FIXED
E18	ECOS C99	38	0	6	0	0	VAR
B1C	ECOS C99	138	2	6	0	0	VAR
NNA	ECOS C99	32	6	6	0	0	VAR
SOP	ECOS C99	10	6	6	0	0	VAR
S99	ECOS C99	18	6	6	0	10	VAR
A34BO	ECOS REP	4	6	4	0	28	VAR
A34AO	ECOS REM	4	0	6	1	7	VAR
A34EO	ECOS REP	4	0	6	1	11	VAR
A03BO	ECOS REM	1	0	6	1	15	VAR
A24MO	ECOS C99	1	0	6	1	16	VAR
MCOMO2	SCOS FCO2, MCO2 DIRECTORY	1	0	6	1	17	VAR
F1	ECOS REM	2	0	6	1	18	VAR
S05	ECOS REM	1	0	6	1	20	VAR
XTLMO	ECOS CTL	4	0	6	1	21	VAR
XTLMO1	ECOS CTL	2	0	6	1	25	VAR
XTLMO2	ECOS CTL	2	0	6	1	27	VAR
TYLM	ECOS CTL	2	0	6	2	0	VAR
XTMMO	ECOS CTM	3	0	6	1	29	VAR
TTMM	ECOS CTM	2	0	6	2	2	VAR
E33	ECOS E33	44	4	6	1	0	VAR
A33AO	ECOS E33	6	6	6	1	0	VAR
T33A	ECOS E33	2	0	6	2	4	VAR
U33ACC	ECOS E33	250	0	5	0	0	VAR
E17	ECOS E17	11	6	6	1	6	VAR
A17A8	ECOS E17	6	6	6	1	17	VAR
T17A	ECOS E17	2	0	6	2	6	VAR
U17AO1	ECOS E17	8	6	6	1	23	VAR
E34	ECOS E34	15	0	6	2	8	VAR
A34MO	ECOS E34	12	4	6	2	12	VAR
U34ALM	ECOS E34	1	6	6	1	31	VAR
U34AST	ECOS E34	100	6	6	2	0	VAR
E13	ECOS E13	18	0	6	3	0	VAR
A13AO	ECOS E13	13	0	6	3	18	VAR
A13AO1	ECOS E13	4	0	6	2	23	VAR
A13AO2	ECOS E13	5	0	6	2	27	VAR
A13AO3	ECOS E13	5	4	6	2	24	VAR
A13AO4	ECOS E13	3	4	6	2	29	VAR
A13AO6	ECOS E13	4	4	4	3	0	VAR
T13A	ECOS E13	7	0	6	3	30	VAR
U13APC	ECOS E13	1	4	4	3	4	VAR
U13APB	ECOS E13	8	4	4	3	5	VAR
A13GO	ECOS E13	4	4	6	3	14	VAR
T13C	ECOS E13	3	4	6	3	18	VAR

A13A03	ECOS E13		4	4	6	3	21	VAR
TOFD	ECOS COF		2	4	6	3	25	VAR
AOFDO	ECOS COF		4	4	6	3	27	VAR
B22	ECOS A22		26	0	6	4	0	VAR
A22A0	ECOS A22		8	0	6	4	26	VAR
T22A	ECOS A22		2	3	6	4	10	VAR
A05A0	ECOS A05		5	2	6	4	12	VAR
T05A	ECOS A05		2	2	6	4	17	VAR
W05TGL	ECOS A05		3	2	6	4	19	VAR
B21	ECOS B21		23	4	6	4	0	VAR
A21A0	ECOS B21		6	2	6	4	21	VAR
T21A	ECOS B21		2	2	6	4	27	VAR
W05PWO	ECOS A05	E13 B21	1	4	6	3	31	VAR
A16A0	ECOS B16		6	4	6	4	23	VAR
T16A	ECOS B16		2	2	6	4	29	VAR
WWFMT	ECOS C99	WPM FORMATS	3	4	6	4	27	VAR
THER	ECOS C99		2	4	6	4	30	VAR
TEJB	ECOS C99		1	2	6	4	31	VAR
TOPN	ECOS C99		2	0	6	5	0	VAR
TVFI	ECOS C99		1	0	6	5	2	VAR
TPTC	ECOS C99		2	0	6	5	3	VAR
TWBO	ECOS C99		1	0	6	5	5	VAR
TPLS	ECOS C99		2	0	6	5	6	VAR
T37A	ECOS C99		1	0	6	5	8	VAR
A19G0	ECOS P19		5	0	6	5	9	VAR
B19	ECOS P19		1	0	6	5	14	VAR
T19C	ECOS P19		2	0	6	5	15	VAR
A19A0	ECOS P19		6	0	6	5	17	VAR
U19A01	ECOS P19		1	0	6	5	23	VAR
T19A	ECOS P19		2	0	6	5	24	VAR
B20	ECOS P20		9	2	6	5	0	VAR
T20A	ECOS P20		2	0	6	5	26	VAR
U20A01	ECOS P20		9	2	6	5	9	VAR
A20A0	ECOS P20		5	2	6	5	18	VAR
TMRZ	ECOS C99	P03	1	0	6	5	28	VAR
AMAG0	ECOS C99	P02 P03 P20	2	0	6	5	29	VAR
A03A0	ECOS P03		5	2	6	5	22	VAR
T03A	ECOS P03		2	2	6	5	28	VAR
B02	ECOS P02		13	4	6	5	0	VAR
A02A0	ECOS P02		10	6	6	5	4	VAR
A02A01	ECOS P02		1	0	6	5	11	VAR
A02A02	ECOS P02		6	4	6	5	14	VAR
T02A	ECOS P02		2	2	6	5	10	VAR
T02C	ECOS P02		2	6	6	5	20	VAR
T02S	ECOS P02		2	4	6	5	22	VAR
A8MA0	ECOS C99		3	6	6	5	24	VAR
T8MA	ECOS C99		2	6	6	5	27	VAR
O10NA	ECOS C99		58	0	6	6	0	VAR
A19E0	ECOS REM		3	6	6	5	29	VAR
T14A	ECOS REM		1	2	5	5	31	VAR
TVIO	ECOS REM		2	2	5	5	29	VAR
TVTR	ECOS REM		2	2	5	5	27	VAR
D02A01	ECOS REM		47	2	6	6	0	VAR
A13E0	ECOS REM		4	4	6	6	1	VAR
A13D0	ECOS REM		3	4	6	6	5	VAR
T13C0	ECOS REM		4	4	6	6	8	VAR
A16G0	ECOS REM		4	4	6	6	12	VAR
A20E0	ECOS REM		4	4	6	6	16	VAR
T20F	ECOS REM		2	3	5	5	25	VAR
U16G01	ECOS REM		1	4	6	6	20	VAR
T13E	ECOS REM		2	2	5	5	23	VAR

T19E	ECOS REM	2	2	5	5	31	VAR
D03A01	ECOS REM	76	4	5	5	0	VAR
T34E	ECOS REM	3	2	5	5	19	VAR
T14C	ECOS REM	2	2	5	5	17	VAR
T34A	ECOS REM	2	2	5	5	15	VAR
T13D	ECOS REM	2	2	5	5	13	VAR
T13C	ECOS REM	2	2	5	5	11	VAR
T34B	ECOS REM	2	3	5	5	9	VAR
T22H	ECOS REM	1	2	5	5	8	VAR
TVR6	ECOS REM	1	2	5	5	7	VAR
824	ECOS REM	8	4	6	6	21	VAR
TER101	ECOS REM	5	6	6	6	0	VAR
D300C	ECOS REM	58	2	5	6	0	VAR
TCD7	ECOS REM	1	3	5	5	6	VAR
TVR7	ECOS REM	1	3	5	5	5	VAR
TVR3	ECOS REM	1	3	5	5	4	VAR
TER102	ECOS REM	1	4	6	6	29	VAR
EDUC0	ECOS REM	4	6	6	6	5	VAR
TER103	ECOS REM	1	4	6	6	30	VAR
TER104	ECOS REM	5	6	6	6	9	VAR
TVR2	ECOS REM	2	2	5	5	2	VAR
TVR8	ECOS REM	1	2	5	5	1	VAR
TXR2	ECOS REM	1	2	5	5	0	VAR
TVR9	ECOS REM	1	6	5	5	31	VAR
TVR0	ECOS REM	1	6	5	5	30	VAR
TYR4	ECOS REM	1	6	5	5	29	VAR
VER10	ECOS REM	9	6	6	6	14	VAR
TXB1	ECOS REM	1	6	5	5	28	VAR
TXR1	ECOS REM	1	6	5	5	27	VAR
TVRA	ECOS REM	1	6	5	5	26	VAR
TVR1	ECOS REM	2	6	5	5	24	VAR
TAC1	ECOS REM	1	6	5	5	23	VAR
TDEP	ECOS REM	2	6	5	5	21	VAR
TAC2	ECOS REM	1	6	5	5	20	VAR
TGAP	ECOS REM	2	6	5	5	18	VAR
632	ECOS REM	4	6	6	6	23	VAR
TGBC	ECOS REM	2	6	5	5	18	VAR
831	ECOS REM	8	3	6	7	15	VAR
QRTE	ECOS REM	5	6	6	6	27	VAR
FP	ECOS REM	1	4	6	4	31	VAR
QRTM	ECOS REM	5	0	6	7	35	VAR
808	ECOS REM	16	4	6	7	0	VAR
TAPP	ECOS REM	3	4	5	5	13	VAR
EPP10	ECOS REM	3	2	6	7	23	VAR
TFC3	ECOS REM	1	6	5	5	12	VAR
D300B	ECOS REM	58	4	5	6	0	VAR
FFD1	ECOS REM	18	4	6	7	0	VAR
TXB2	ECOS REM	1	6	5	5	11	VAR
TVR5	ECOS REM	1	6	5	5	10	VAR
FCDP	ECOS REM	3	2	6	7	26	VAR
HH1	ECOS REM	56	2	5	3	0	VAR
TADR	ECOS REM	7	6	5	5	8	VAR
THP3	ECOS REM	1	6	5	5	7	VAR
THA	ECOS FC02, DISPLAY	3	2	6	7	29	VAR
TFFD00	ECOS FC02, G-ANI	2	4	6	7	16	VAR
RCOV01	ECOS FC01 DIRECTORY	1	0	6	7	31	VAR
FFDBSC	ECOS FC02 DIRECTORY	1	4	6	7	18	VAR
FCDRIV	ECOS FC02, G-ANI	1	4	6	7	19	VAR
BRN	ECOS FC02, DISPLAY	1	4	6	7	20	VAR
HHUTHI	ECOS FC01 DIRECTORY	1	4	6	7	21	VAR
AUD	ECOS FC02, DISPLAY	2	4	6	7	22	VAR

```
NCSANIS  SCO3 FC01, MC01 LMI'S          35  6 5 4  0  VAR
         END                             0  0 0 0  0
```

C.3 MMU TAPE MAP

ORIGINAL PAGE IS
OF POOR QUALITY

	FILE 0	FILE 1	FILE 2	FILE 3	FILE 4	FILE 5	FILE 6	FILE 7
TRACK 0	2...2...	1.......	....	....	....	WWWWWWWD	WWWWWWWW	...2...2
TRACK 1	2...2...	1.......	....	....	....	WWWWWWWD	WWWWWWWW	...2...2
TRACK 2	....	....	....	....	....	...WDWWD	WWWWWWWW	....
TRACK 3	....	....	....	....	....	...WDWWD	WWWWWWWW	....
TRACK 4	....	6	.G.G.G.G	.G.G.E.1	....	.....WWC	WWWWWWWD	....
TRACK 5	....	6	.G.G.G.G	.G.G.E.1	....	.....WWC	WWWWWWWD	....
TRACK 6	....	....	....	....	....	....WSWD	WWWWWWW1	....
TRACK 7	....	....	....	....	....	....WBWD	WWWWWWW1	....

NEXT SEED VALUE= 87654321
BBBDIAG: DUPLICATE UNSCHED TIME= 44.01 HRS

C.4 TRACE OF MMU UTILIZATION

S I M U L A T E D   M I S S I O N   O F   M M U   U S E

MET (HRS)	OPERATION (MODEL,STEP)	BLOCKS SKIPPED: BEFORE ACCESS #1	TOTAL IN	MAX IN	ON ACCESS#1	LAST POSITION: FILE	SF	BLK
10.35	NPMFMT	0	0	0	0	6	4	30
10.50	020-F01A,5	20	13	10	3	6	5	9
10.63	033-F1/4,7	131	112	76	3	5	0	0
10.79	TMEM	414	0	0	0	6	5	0
11.41	TMEM	0	0	0	0	6	4	30
11.49	NMMFMT	0	0	0	0	6	4	27
11.85	NMMFMT	0	0	0	0	6	4	30
12.54	U34AST	6	0	0	0	6	2	0
13.93	M30CFOT1,2	119	0	0	0	6	5	30
13.95	MMA	158	0	0	0	6	0	0
14.92	TMEM	159	0	0	0	6	5	0
14.95	TMEM	0	0	0	0	6	4	30
14.97	NMMFMT	0	0	0	0	6	4	27
15.05	017-F01A,1	106	0	0	0	6	1	6
15.08	017-F01A,2	11	24	15	2	6	1	23
15.17	017-F01A,3	0	0	0	0	6	1	31
15.18	TPTC	100	0	0	0	6	5	5
15.42	020-F05A,1	4	47	18	2	6	5	31
15.56	MMA	159	0	0	0	6	0	0
15.60	TDPM	160	0	0	0	6	5	2
15.97	017-F01A,1	113	0	0	0	6	1	6
16.00	017-F01A,2	11	24	15	2	6	1	23
16.08	017-F01A,3	0	0	0	0	6	1	31
16.17	NMMFMT	92	0	0	0	6	4	30
16.21	020-F05A,1	2	47	20	5	6	5	31
16.35	TMEM	31	0	0	0	6	4	30
16.88	020-F05A,1	2	38	19	4	6	5	31
17.22	ITMMO	127	5	5	2	6	2	4
17.46	017-F01A,1	19	0	0	0	6	1	6
17.49	017-F01A,2	11	24	15	2	6	1	23
17.54	TMEM	103	0	0	0	6	5	0
17.57	017-F01A,3	97	0	0	0	6	1	23
17.69	020-F05M,1	105	42	20	5	6	5	31
17.70	U33ACC	197	0	0	0	5	0	0
17.96	TMMD	421	0	0	0	6	5	6
18.36	020-F05A,1	3	47	18	2	6	5	31
18.55	034-F56M,4	104	28	24	3	6	1	31
18.90	034-F56M,7	1	0	0	0	6	5	4
19.22	034-F56M,9	0	0	0	0	6	2	0
19.26	TMEM	94	0	0	0	6	5	0
19.27	017-F01A,1	111	0	0	0	6	1	6
19.30	034-F56M,11	26	0	0	0	6	5	4
19.30	017-F01A,2	109	30	21	2	6	1	23
19.38	017-F01A,3	0	0	0	0	6	1	31
19.55	NPMFMT	92	0	0	0	6	4	30
19.69	M1F07A6A,2	117	0	0	0	6	4	10
19.81	NMMFMT	17	0	0	0	6	4	30
19.86	TEJR	1	0	0	0	6	5	0
19.94	M1F07A6A,4	22	0	0	0	6	0	9
20.08	X1F07A6A,6	0	0	0	0	6	4	10
20.12	NMMFMT	17	0	0	0	6	4	30

C.5 STATISTICAL SUMMARY

S T A T I S T I C A L   S U M M A R Y RUN 4   22-MAR-84 08:03:04

TEN LARGEST "SKIP TO"

HRS	MODEL,STEP	#BLK
13.93	N3DCFOT1,2	419
101.33	U2DA01	425
143.22	AOFDO	379
65.00	NBPFMT	411
18.38	020-F05A,1	416
110.18	B34	328
38.84	017-F01B,1	394
74.19	333	388
123.55	TPLB	422
11.32	THEN	414

TOTAL "SKIP TO":	57634
NO. OF OBS.:	1054
MEAN :	54.7
VARIANCE:	4196.0
CV(%):	118.5

TEN LARGEST "SKIP IN"

HRS	MODEL,STEP	#BLK
100.35	N2F09/8C,1	87
127.09	N2F08/9B,1	87
125.63	N2F11/12,1	87
128.63	N2F9B/C3,1	87
122.57	N2F9B/C2,1	87
80.90	N2F5A067,1	84
10.63	033-F1/4,2	112
121.28	N2F09/8A,1	87
83.11	005-F01C,3	95
34.56	013-F2/3,2	100

TOTAL "SKIP IN":	5854
NO. OF OBS.:	240
MEAN :	24.4
VARIANCE:	803.6
CV(%):	116.2

TEN LARGEST MAX TRAVEL IN (MTI)

HRS	MODEL,STEP	#BLKS	ACC#
121.28	N2F09/8A,1	27	3
127.09	N2F08/9B,1	27	3
128.63	N2F9B/C3,1	27	3
100.35	N2F09/8C,1	27	3
125.63	N2F11/12,1	27	3
122.57	N2F9B/C2,1	27	3
10.63	033-F1/4,2	76	3
80.90	N2F5A067,1	27	3
84.38	021-F1/2,1	29	4
34.56	013-F2/3,2	35	2

TOTAL MTI :	2555
NO. OF OBS.:	240
MEAN :	10.6
VARIANCE:	106.5
CV(%):	96.9

NEXT SEED VALUE= 39475893

APPENDIX D

OPERATING THE SIMULATION

This appendix defines how to run the simulation. The simulation model is designed to run on the Digital Equipment Corporation (DEC) VAX 11/780 computer with FORTRAN 77. The data, command, and program files required and their directory locations will be given. Procedures are presented for setting up the simulation and for running it.

Establishing the Files

The model is currently established in MSFC's VAX4 computer in account QS1:[EL121.NONEMAN]. The input files should reside with the program in the same directory or subdirectory. They should be defined as desribed in Table 20.

Table 20. User Defined Files

File Name	Contents	Record Format
MMUALL.DAT	Data-set Definitions	A8,1X,A30,2X,I3,2X,3(I1,1X),I2,1X,I10,I9
MSDS.DAT	Mission Timeline/ Data-set Correlations	A12,15A10
UNSCHE.DAT	Unscheduled Data-sets Utilizations	I4,2F8.3,15A10
SEEDS.DAT	Random Number Seeds	I8

The following procedure should be followed to set up for simulation runs:

1. Copy the data-set definition data from the MMU generation information into MMUALL.DAT.

2. Define the mission timeline/ data-sets correlation data in file MSDS.DAT.

3. Define the unscheduled data-sets utilization data in file UNSCHE.DAT.

4. Define the random number seeds in SEEDS.DAT. One mission simulation will be performed for each seed entered. The last record of this file must contain a seed value set to zero to stop the simulation.

5. Copy the time-ordered mission timeline experiment ON/OFF file defined by the mission planners to QSA1:[EL121.NONEMAN]EXPERIMNT.FIN.

Running the Simulation

Once the files are established, the simulation may be run from the terminal or in a batch mode. All of the reports are directed to disk files which may be reviewed after the run at the terminal or from a line printer listing. Since the run time of the simulation for five missions typically is greater than fifteen minutes interactively, the batch mode is often preferred. To invoke the model interactively, enter the following command:

```
$ @MMUALL
```

A batch run may be submitted at the terminal by entering the command,

```
$ SUBMIT MMUALL
```

By either method, the command file MMUALL.COM, listed below will be executed.

```
$ SET DEF [EL121'.NONEMAN]
$ ASSIGN MMUALL.DAT FOR005:
$ ASSIGN MMUALL.PRT FOR006:
$ ASSIGN TT: FOR001:
$ RUN MMUALL
$ PRINT MMUALL.PRT
$ PRINT MSDS.DAT,UNSCHE.DAT
```

This command file runs the program and prints the reports file. It is assumed that the executable file MMUALL.EXE exists in the same directory containing the input files.

Program Source

Should there be any reason to modify the simulation code, the program sources are defined as follows. The modified MMU allocation source is in MMUALL.FOR. The simulation routines which supplement this code are in MMUSUB.FOR. When the programs are linked the subroutine, READOF, which reads the mission timeline schedule file, must be included by using the object library QS1:[EL121.EST]ESS.OLB. The link command is in MMULNK.COM listed below.

```
$ LINK MMUALL,MMUSUB,QS1:[EL121.EST]ESS/LIB
```