



System Test

User Guide

Mini-Computer Operations

2722 Michelson Drive
P.O. Box C-19504
Irvine, California 92713

UP-9098



SYSTEM TEST

User Guide

UP-9098

98A 9952 860

APRIL 1980

The statements in this publication are not intended to create any warranty, express or implied. Equipment specifications and performance characteristics stated herein may be changed at any time without notice. Address comments regarding this document to Sperry Univac, Mini-Computer Operations, Publications Department, 2722 Michelson Drive, P.O. Box C-19504, Irvine, California, 92713.

**COPYRIGHT ©1980 by
SPERRY CORPORATION
ALL RIGHTS RESERVED**

Sperry Univac is a division of Sperry Corporation

Printed in U.S.A.

PAGE STATUS SUMMARY

ISSUE: UP-9098

| Part/Section | Page Number | Update Level | Part/Section | Page Number | Update Level | Part/Section | Page Number | Update Level |
|--------------|-------------|--------------|--------------|-------------|--------------|--------------|-------------|--------------|
| Cover | | | 16 | 1 thru 4 | | | | |
| Title Page | | | Appendix A | 1 thru 32 | | | | |
| PSS | 1 | | | | | | | |
| CR | 1 | | | | | | | |
| Contents | 1 thru 9 | | | | | | | |
| 1 | 1 thru 7 | | | | | | | |
| 2 | 1 thru 5 | | | | | | | |
| 3 | 1 thru 8 | | | | | | | |
| 4 | 1 thru 42 | | | | | | | |
| 5 | 1 thru 15 | | | | | | | |
| 6 | 1 thru 66 | | | | | | | |
| 7 | 1 thru 2 | | | | | | | |
| 8 | 1 | | | | | | | |
| 9 | 1 thru 2 | | | | | | | |
| 10 | 1 thru 2 | | | | | | | |
| 11 | 1 thru 5 | | | | | | | |
| 12 | 1 thru 4 | | | | | | | |
| 13 | 1 thru 4 | | | | | | | |
| 14 | 1 thru 4 | | | | | | | |
| 15 | 1 thru 2 | | | | | | | |

**New pages*

All the technical changes are denoted by an arrow (→) in the margin. A downward pointing arrow (↓) next to a line indicates that technical changes begin at this line and continue until an upward pointing arrow (↑) is found. A horizontal arrow (→) pointing to a line indicates a technical change in only that line. A horizontal arrow located between two consecutive lines indicates technical changes in both lines or deletions.

CHANGE RECORD

| Change Designation | Issue Date | Change Description |
|--------------------|------------|--------------------|
| | April 1980 | Original issue. |

Change Procedure:

When changes are made to this manual, updated pages are issued. These updated pages are either added to this manual or used to replace obsolete pages. The specific pages affected by each change are identified on the PAGE STATUS SUMMARY page.

Contents

PAGE STATUS SUMMARY

CHANGE RECORD

CONTENTS

1 INTRODUCTION

| | | |
|-----|--|-----|
| 1.1 | GENERAL. | 1-1 |
| 1.2 | SCOPE. | 1-1 |
| 1.3 | DESCRIPTION OF THE SYSTEM TEST | 1-3 |
| 1.4 | RELATED PUBLICATIONS | 1-4 |
| 1.5 | NOTATIONS USED IN THIS MANUAL. | 1-6 |

2 SYSTEM OVERVIEW

| | | |
|---------|--|-----|
| 2.1 | COMPONENTS OF THE SYSTEM TEST. | 2-1 |
| 2.1.1 | PRELIMINARY TESTS. | 2-1 |
| 2.1.1.1 | Instructions Test. | 2-1 |
| 2.1.1.2 | Memory Test. | 2-1 |
| 2.1.2 | PRELIMINARY LOADER | 2-1 |
| 2.1.3 | SYSTEM TEST GENERATOR. | 2-2 |
| 2.1.4 | EXECUTIVE PROGRAM. | 2-2 |
| 2.1.5 | LIBRARY OF UNIT PROGRAMS | 2-2 |
| 2.2 | DESIGN GOALS AND MAINTENANCE CONCEPTS. | 2-4 |
| 2.3 | OPERATIONAL FEATURES | 2-5 |

3 SYSTEM TEST GENERATION

| | | |
|-----|----------------------------------|-----|
| 3.1 | PURPOSE. | 3-1 |
| 3.2 | HARDWARE CONFIGURATION | 3-1 |
| 3.3 | OVERVIEW OF THE PROCESS. | 3-2 |

| | | |
|---------|---|-----|
| 3.4 | SAMPLE SYSTEM TEST GENERATION. | 3-2 |
| 3.4.1 | PREPARATION. | 3-2 |
| 3.4.2 | LOADING THE PRELIMINARY LOADER | 3-3 |
| 3.4.2.1 | Manual Loading Via the Computer Control Panel (V77-600 Computers Only). | 3-4 |
| 3.4.2.2 | Manual Loading Via the Virtual Console (V77-800 Computers Only) | 3-6 |
| 3.4.3 | ERROR INDICATIONS AND HANDLING | 3-6 |
| 3.4.4 | CONFIGURATION INTERROGATION. | 3-7 |

4 CONFIGURATION DIALOG

| | | |
|---------|--|------|
| 4.1 | GENERAL. | 4-1 |
| 4.2 | SYSTEM TEST CONFIGURATION GUIDE. | 4-1 |
| 4.3 | CONFIGURATION QUESTIONS. | 4-4 |
| 4.3.1 | INTRODUCTORY CONFIGURATION QUESTION. | 4-5 |
| 4.3.1.1 | Description. | 4-5 |
| 4.3.1.2 | Acceptable Replies | 4-5 |
| 4.3.1.3 | Example. | 4-6 |
| 4.3.2 | CONFIGURATION QUESTION NUMBER 1. | 4-7 |
| 4.3.2.1 | Purpose. | 4-7 |
| 4.3.2.2 | Acceptable Replies | 4-7 |
| 4.3.2.3 | Example. | 4-8 |
| 4.3.3 | CONFIGURATION QUESTION NUMBER 2. | 4-9 |
| 4.3.3.1 | Purpose. | 4-9 |
| 4.3.3.2 | Acceptable Replies | 4-9 |
| 4.3.3.3 | Example. | 4-10 |
| 4.3.4 | CONFIGURATION QUESTION NUMBER 3. | 4-11 |
| 4.3.4.1 | Purpose. | 4-11 |
| 4.3.4.2 | Acceptable Replies | 4-11 |
| 4.3.4.3 | Example. | 4-11 |
| 4.3.5 | CONFIGURATION QUESTION NUMBER 4. | 4-12 |
| 4.3.5.1 | Purpose. | 4-12 |
| 4.3.5.2 | Acceptable Replies | 4-12 |
| 4.3.5.3 | Example. | 4-12 |
| 4.3.6 | CONFIGURATION QUESTION NUMBER 5. | 4-13 |
| 4.3.6.1 | Purpose. | 4-13 |
| 4.3.6.2 | Acceptable Replies | 4-13 |
| 4.3.6.3 | Example. | 4-13 |
| 4.3.7 | CONFIGURATION QUESTION NUMBER 6. | 4-14 |
| 4.3.7.1 | Purpose. | 4-14 |
| 4.3.7.2 | Acceptable Replies | 4-14 |
| 4.3.7.3 | Example. | 4-15 |
| 4.3.8 | CONFIGURATION QUESTION NUMBER 7. | 4-16 |
| 4.3.8.1 | Purpose. | 4-16 |

| | | |
|----------|--|------|
| 4.3.8.2 | Acceptable Replies | 4-16 |
| 4.3.8.3 | Example. | 4-17 |
| 4.3.9 | CONFIGURATION QUESTION NUMBER 8. | 4-18 |
| 4.3.9.1 | Purpose. | 4-18 |
| 4.3.9.2 | Acceptable Replies | 4-18 |
| 4.3.9.3 | Example. | 4-18 |
| 4.3.10 | CONFIGURATION QUESTION NUMBER 9. | 4-19 |
| 4.3.10.1 | Purpose. | 4-19 |
| 4.3.10.2 | Acceptable Replies | 4-19 |
| 4.3.10.3 | Example. | 4-19 |
| 4.3.11 | CONFIGURATION QUESTION NUMBER 10 | 4-20 |
| 4.3.11.1 | Purpose. | 4-20 |
| 4.3.11.2 | Acceptable Replies | 4-20 |
| 4.3.11.3 | Example. | 4-20 |
| 4.3.12 | CONFIGURATION QUESTION NUMBER 11 | 4-21 |
| 4.3.12.1 | Purpose. | 4-21 |
| 4.3.12.2 | Acceptable Replies | 4-22 |
| 4.3.12.3 | Examples | 4-24 |
| 4.3.13 | CONFIGURATION QUESTION NUMBER 12 | 4-27 |
| 4.3.13.1 | Purpose. | 4-27 |
| 4.3.13.2 | Acceptable Replies | 4-28 |
| 4.3.13.3 | Example. | 4-29 |
| 4.3.14 | CONFIGURATION QUESTION NUMBER 13 | 4-30 |
| 4.3.14.1 | Purpose. | 4-30 |
| 4.3.14.2 | Acceptable Replies | 4-30 |
| 4.3.14.3 | Example. | 4-30 |
| 4.3.15 | CONFIGURATION QUESTION NUMBER 14 | 4-31 |
| 4.3.15.1 | Purpose. | 4-31 |
| 4.3.15.2 | Acceptable Replies | 4-31 |
| 4.3.15.3 | Example. | 4-32 |
| 4.3.16 | CONFIGURATION QUESTION NUMBER 15 | 4-33 |
| 4.3.16.1 | Purpose. | 4-33 |
| 4.3.16.2 | Acceptable Replies | 4-33 |
| 4.3.16.3 | Example. | 4-33 |
| 4.3.17 | CONFIGURATION QUESTION NUMBER 16 | 4-34 |
| 4.3.17.1 | Purpose. | 4-34 |
| 4.3.17.2 | Acceptable Replies | 4-34 |
| 4.3.17.3 | Example. | 4-34 |
| 4.3.18 | CONFIGURATION QUESTION NUMBER 17 | 4-35 |
| 4.3.18.1 | Purpose. | 4-35 |
| 4.3.18.2 | Acceptable Replies | 4-35 |
| 4.3.18.3 | Example. | 4-35 |
| 4.3.19 | CONFIGURATION QUESTION NUMBER 18 | 4-36 |
| 4.3.19.1 | Purpose. | 4-36 |
| 4.3.19.2 | Acceptable Replies | 4-36 |
| 4.3.19.3 | Example. | 4-36 |
| 4.3.20 | CONFIGURATION QUESTION NUMBER 19 | 4-38 |
| 4.3.20.1 | Purpose. | 4-38 |
| 4.3.20.2 | Acceptable Replies | 4-38 |

| | | |
|----------|---|------|
| 4.3.20.3 | Example. | 4-38 |
| 4.3.21 | PREPARATION OF THE CONFIGURED SYSTEM TEST | 4-39 |
| 4.4 | ERROR DETECTION DURING SYSTEM TEST | |
| | GENERATION | 4-40 |
| 4.4.1 | PRELIMINARY LOADER | 4-40 |
| 4.4.2 | ERRORS DETECTED BY THE SYSTEM TEST | |
| | GENERATOR. | 4-41 |
| 4.4.2.1 | Map Errors | 4-41 |
| 4.4.2.2 | Configuration Errors | 4-42 |
| 4.4.2.3 | Input and Output Errors. | 4-42 |
| 5 | CONFIGURED SYSTEM TEST: OPERATIONAL THEORY | |
| 5.1 | GENERAL. | 5-1 |
| 5.2 | PURPOSE. | 5-1 |
| 5.3 | COMPONENTS | 5-1 |
| 5.3.1 | PRELIMINARY INSTRUCTIONS TEST. | 5-2 |
| 5.3.2 | PRELIMINARY MEMORY TEST. | 5-4 |
| 5.3.3 | PRELIMINARY LOADER | 5-6 |
| 5.3.4 | EXECUTIVE PROGRAM. | 5-6 |
| 5.3.5 | UNIT PROGRAMS. | 5-7 |
| 5.3.5.1 | Identification | 5-7 |
| 5.3.5.2 | Assembly | 5-9 |
| 5.3.5.3 | Executive Service Calls. | 5-10 |
| 5.3.5.4 | Operational Theory | 5-14 |
| 6 | SYSTEM TEST OPERATION | |
| 6.1 | OVERVIEW | 6-1 |
| 6.2 | PREPARATION. | 6-1 |
| 6.3 | LOADING THE CONFIGURED SYSTEM TEST | 6-2 |
| 6.4 | OPERATOR COMMANDS. | 6-2 |
| 6.4.1 | USING THE OPERATOR COMMANDS. | 6-3 |
| 6.4.2 | HELP COMMAND | 6-7 |
| 6.4.2.1 | Purpose. | 6-7 |
| 6.4.2.2 | Format | 6-7 |
| 6.4.2.3 | Usage. | 6-7 |
| 6.4.2.4 | Example. | 6-8 |
| 6.4.3 | INDEX COMMAND. | 6-9 |
| 6.4.3.1 | Purpose. | 6-9 |

| | | |
|----------|------------------------------|------|
| 6.4.3.2 | Format | 6-9 |
| 6.4.3.3 | Usage. | 6-9 |
| 6.4.3.4 | Example. | 6-10 |
| 6.4.4 | LOG COMMAND. | 6-12 |
| 6.4.4.1 | Purpose. | 6-12 |
| 6.4.4.2 | Format | 6-12 |
| 6.4.4.3 | Usage. | 6-12 |
| 6.4.4.4 | Example. | 6-13 |
| 6.4.5 | RUN COMMAND. | 6-14 |
| 6.4.5.1 | Purpose. | 6-14 |
| 6.4.5.2 | Format | 6-14 |
| 6.4.5.3 | Usage. | 6-14 |
| 6.4.5.4 | Examples | 6-17 |
| 6.4.6 | DISABLE COMMAND. | 6-21 |
| 6.4.6.1 | Purpose. | 6-21 |
| 6.4.6.2 | Format | 6-21 |
| 6.4.6.3 | Usage. | 6-21 |
| 6.4.6.4 | Examples | 6-22 |
| 6.4.7 | ENABLE COMMAND | 6-23 |
| 6.4.7.1 | Purpose. | 6-23 |
| 6.4.7.2 | Format | 6-23 |
| 6.4.7.3 | Usage. | 6-23 |
| 6.4.7.4 | Examples | 6-24 |
| 6.4.8 | CHANGE COMMAND | 6-25 |
| 6.4.8.1 | Purpose. | 6-25 |
| 6.4.8.2 | Format | 6-25 |
| 6.4.8.3 | Usage. | 6-26 |
| 6.4.8.4 | Examples | 6-27 |
| 6.4.9 | CHANGE MAP COMMAND | 6-30 |
| 6.4.9.1 | Purpose. | 6-30 |
| 6.4.9.2 | Format | 6-30 |
| 6.4.9.3 | Usage. | 6-30 |
| 6.4.9.4 | Examples | 6-31 |
| 6.4.10 | EDIT COMMAND | 6-33 |
| 6.4.10.1 | Purpose. | 6-33 |
| 6.4.10.2 | Format | 6-33 |
| 6.4.10.3 | Usage. | 6-33 |
| 6.4.10.4 | Examples | 6-34 |
| 6.4.11 | INSERT COMMAND | 6-36 |
| 6.4.11.1 | Purpose. | 6-36 |
| 6.4.11.2 | Format | 6-36 |
| 6.4.11.3 | Usage. | 6-36 |
| 6.4.11.4 | Examples | 6-37 |
| 6.4.12 | KEY COMMAND. | 6-38 |
| 6.4.12.1 | Purpose. | 6-38 |
| 6.4.12.2 | Format | 6-38 |
| 6.4.12.3 | Usage. | 6-38 |
| 6.4.12.4 | Examples | 6-39 |
| 6.4.13 | REG COMMAND. | 6-40 |
| 6.4.13.1 | Purpose. | 6-40 |

| | | |
|----------|---|------|
| 6.4.13.2 | Format | 6-40 |
| 6.4.13.3 | Usage. | 6-40 |
| 6.4.13.4 | Examples | 6-41 |
| 6.4.14 | SEARCH COMMAND | 6-42 |
| 6.4.14.1 | Purpose. | 6-42 |
| 6.4.14.2 | Format | 6-42 |
| 6.4.14.3 | Usage. | 6-42 |
| 6.4.14.4 | Examples | 6-43 |
| 6.4.15 | TRAP COMMAND | 6-45 |
| 6.4.15.1 | Purpose. | 6-45 |
| 6.4.15.2 | Format | 6-45 |
| 6.4.15.3 | Usage. | 6-45 |
| 6.4.15.4 | Examples | 6-47 |
| 6.4.16 | VIEW COMMAND | 6-49 |
| 6.4.16.1 | Purpose. | 6-49 |
| 6.4.16.2 | Format | 6-49 |
| 6.4.16.3 | Usage. | 6-49 |
| 6.4.16.4 | Examples | 6-50 |
| 6.5 | SUMMARY OF OPERATOR COMMANDS | 6-51 |
| 6.6 | EXECUTIVE SERVICE CALLS. | 6-52 |
| 6.7 | SAMPLE SESSION | 6-53 |
| 6.8 | ERROR DETECTION. | 6-56 |
| 6.8.1 | MAP ERRORS | 6-56 |
| 6.8.2 | COMMAND ERRORS | 6-57 |
| 6.8.3 | EXECUTION ERRORS | 6-58 |
| 6.8.3.1 | Print Request Error. | 6-58 |
| 6.8.3.2 | Invalid BIC Request. | 6-58 |
| 6.8.3.3 | BIC Request Error. | 6-59 |
| 6.8.3.4 | Memory Protect Error | 6-59 |
| 6.8.3.5 | Unexpected Interrupt | 6-63 |
| 6.8.3.6 | I/O Request Error. | 6-64 |
| 6.8.3.7 | Unit Program Errors. | 6-65 |
| 7 | V77-600 WRITABLE CONTROL STORE (F3034) UNIT PROGRAM | |
| 7.1 | PURPOSE AND OPERATION. | 7-1 |
| 7.2 | WCS ERROR MESSAGES | 7-1 |
| 7.3 | EXAMPLES | 7-2 |

| | | |
|------|---|------|
| 8 | V77-600 FLOATING-POINT PROCESSOR (F3027) UNIT PROGRAM | |
| 8.1 | PURPOSE AND OPERATION. | 8-1 |
| 8.2 | FPP ERROR MESSAGES | 8-1 |
| 8.3 | EXAMPLES | 8-1 |
| 9 | V77-800 WRITABLE CONTROL STORE (F3057) AND FLOATING-POINT PROCESSOR (F3053) UNIT PROGRAM | |
| 9.1 | PURPOSE AND OPERATION. | 9-1 |
| 9.2 | WCS AND FPP (V77-800) ERROR MESSAGES . . . | 9-1 |
| 9.3 | EXAMPLES | 9-2 |
| 10 | MEMORY UNIT PROGRAM | |
| 10.1 | PURPOSE AND OPERATION. | 10-1 |
| 10.2 | MEMORY ERROR MESSAGES. | 10-1 |
| 10.3 | EXAMPLES | 10-2 |
| 11 | CARTRIDGE DISK UNIT PROGRAM | |
| 11.1 | PURPOSE AND OPERATION. | 11-1 |
| 11.2 | MEMORY ERROR MESSAGES. | 11-1 |
| 11.3 | EXAMPLES | 11-4 |
| 12 | DISK MEMORY AND CONTROLLER UNIT PROGRAM | |
| 12.1 | PURPOSE AND OPERATION. | 12-1 |
| 12.2 | ERROR MESSAGES | 12-1 |
| 12.3 | EXAMPLES | 12-4 |
| 13 | DIRECT MEMORY ACCESS DISK UNIT PROGRAM | |
| 13.1 | PURPOSE AND OPERATION. | 13-1 |

| | | |
|------|--|------|
| 13.2 | ERROR MESSAGES | 13-1 |
| 13.3 | EXAMPLES | 13-4 |
| 14 | MAGNETIC TAPE UNIT PROGRAM | |
| 14.1 | PURPOSE AND OPERATION. | 14-1 |
| 14.2 | ERROR MESSAGES | 14-1 |
| 14.3 | EXAMPLES | 14-4 |
| 15 | LINE PRINTER UNIT PROGRAM | |
| 15.1 | PURPOSE AND OPERATION. | 15-1 |
| 15.2 | ERROR MESSAGES | 15-1 |
| 15.3 | EXAMPLES | 15-2 |
| 16 | DATA COMMUNICATIONS MULTIPLEXER UNIT PROGRAM | |
| 16.1 | PURPOSE AND OPERATION. | 16-1 |
| 16.2 | ERROR MESSAGES | 16-2 |
| 16.3 | EXAMPLES | 16-2 |
| 16.4 | JUMPER CONFIGURATIONS. | 16-3 |
| A | SYSTEM TEST GENERATION TUTORIAL GUIDE | |
| A.1 | PURPOSE. | A-1 |
| A.2 | SYSTEM TEST CONFIGURATION GUIDE. | A-2 |
| A.3 | SAMPLE SYSTEM TEST GENERATION LISTING. . . | A-18 |
| A.4 | BLANK SYSTEM TEST CONFIGURATION GUIDE. . . | A-21 |

FIGURES

| | | |
|-----|---|------|
| 1-1 | System Test: Operational Flow | 1-5 |
| A-1 | Excerpts from a System Memo. | A-15 |

TABLES

| | | |
|------|---|------|
| 3-1 | Magnetic Tape Bootstrap Routine with Symbolic Coding. | 3-5 |
| 4-1 | Model 4 (Type DF) Disks and Corresponding Platter Numbers. | 4-23 |
| 5-1 | Preliminary Instructions Test Summary. | 5-3 |
| 5-2 | Preliminary Instruction Test Error Codes | 5-5 |
| 5-3 | Unit Program IDs | 5-8 |
| 5-4 | Contents of Location 01000 to 01007. | 5-9 |
| 5-5 | Contents of Locations 01010-01027. | 5-10 |
| 5-6 | Executive Service Calls. | 5-11 |
| 5-7 | Data Passed By Executive Service Calls | 5-12 |
| 11-1 | Cartridge Disk Status Word Significance. | 11-3 |
| 12-1 | Disk Memory Status Word Significance | 12-3 |
| 13-1 | Direct Memory Access Disk Status Word Significance | 13-2 |
| 13-2 | Direct Memory Access Disk Secondary Status Word Significance | 13-3 |
| 14-1 | Magnetic Tape Status Word Significance | 14-3 |
| 16-1 | LAD Type and Feature Number Cross-Reference. | 16-1 |

Section 1 Introduction

1.1 GENERAL

This manual describes the SPERRY UNIVAC System Test. The System Test is a real-time, multiprogrammed, offline, standalone system test for SPERRY UNIVAC V77-600 and V77-800 computers. The System Test can be ordered as Type Number 6334-xx. The System Test supplements the MAINTAIN III Test Programs. It is neither a replacement nor a substitute for the MAINTAIN III Test Programs.

1.2 SCOPE

This manual contains descriptive data, operating instructions, and reference material. The material is directed toward the person who is responsible for diagnosing malfunctions and maintaining Sperry Univac minicomputers. This person is referred to, in this manual, as the "operator of the test".

Detailed procedures for generating a System Test configured for a specific system configuration, and running it on this system, are provided. Errors detected by the System Test result in the generation of error messages. All these error messages are listed and explained in this manual. The components of the System Test and their interrelationships are described. Information about the theory of operation is provided to allow the reader to develop an understanding of the design structure of the System Test.

This manual is intended to be used in conjunction with servicing manuals for the system's central complex and peripheral equipment. Familiarity with the MAINTAIN III Test Program System is necessary, since this test is usually used in conjunction with the MAINTAIN III Test Programs.

The information in this manual is organized as follows:

| | | |
|-----------|-----------------|---|
| Section 1 | Introduction | Overview of the manual and the System Test. |
| Section 2 | System Overview | Descriptive data about System Test components, concepts and features. |

| | | |
|------------|--|--|
| Section 3 | System Test Generation | Explanation of the process and step-by-step procedures for the generation of a configured System Test. |
| Section 4 | Configuration Dialog | Description of the dialog; discussion of each of the configuration questions; examples; preparation of the configured System Test, and error detection and handling during system test generation. |
| Section 5 | Configured System Test: Operational Theory | Functional description of the configured System Test components. |
| Section 6 | System Test Operation | Procedures for initial preparation and loading of the configured System Test; explanation of how the operator can interact with the test. |
| Section 7 | V77-600 Writable Control Store (F3034) Unit Program | Description of purpose and operation of the unit program; list of error messages detected by the program; examples. |
| Section 8 | V77-600 Floating-Point Processor (F3027) Unit Program | Description of purpose and operation of the unit program; list of error messages detected by the program; examples. |
| Section 9 | V77-800 Writable Control Store (F3057) and Floating-Point Processor (F3053) Unit Program | Description of the purpose and operation of the unit program; list of error messages; examples. |
| Section 10 | Memory Unit Program | Description of the purpose and operation of the unit program; list of error messages; examples. |

| | | |
|------------|--|---|
| Section 11 | Cartridge Disk Unit Program | Description of the purpose and operation of the unit program; list of error messages; examples. |
| Section 12 | Disk Memory and Controller Unit Program | Description of the purpose and operation of the unit program; list of error messages; examples. |
| Section 13 | Direct Memory Access Disk Unit Program | Description of the purpose and operation of the unit program; list of error messages; examples. |
| Section 14 | Magnetic Tape Unit Program | Description of the purpose and operation of the unit program; list of error messages; examples. |
| Section 15 | Line Printer | Description of the purpose and operation of the unit program; list of error messages; examples. |
| Section 16 | Data Communications Multiplexer Unit Program | Description of the purpose and operation of the unit program; list of error messages; examples. |
| Appendix A | System Test Generation Tutorial Guide | Instructions on how to prepare system configuration dialog responses. |

1.3 DESCRIPTION OF THE SYSTEM TEST

The System Test is an offline, standalone, diagnostic software package for V77-600 and V77-800 Sperry Univac minicomputers. It allows system components to be tested simultaneously. The System Test allows the user to diagnose malfunctions and maintain Sperry Univac central complex components and peripherals.

The System Test has the following features:

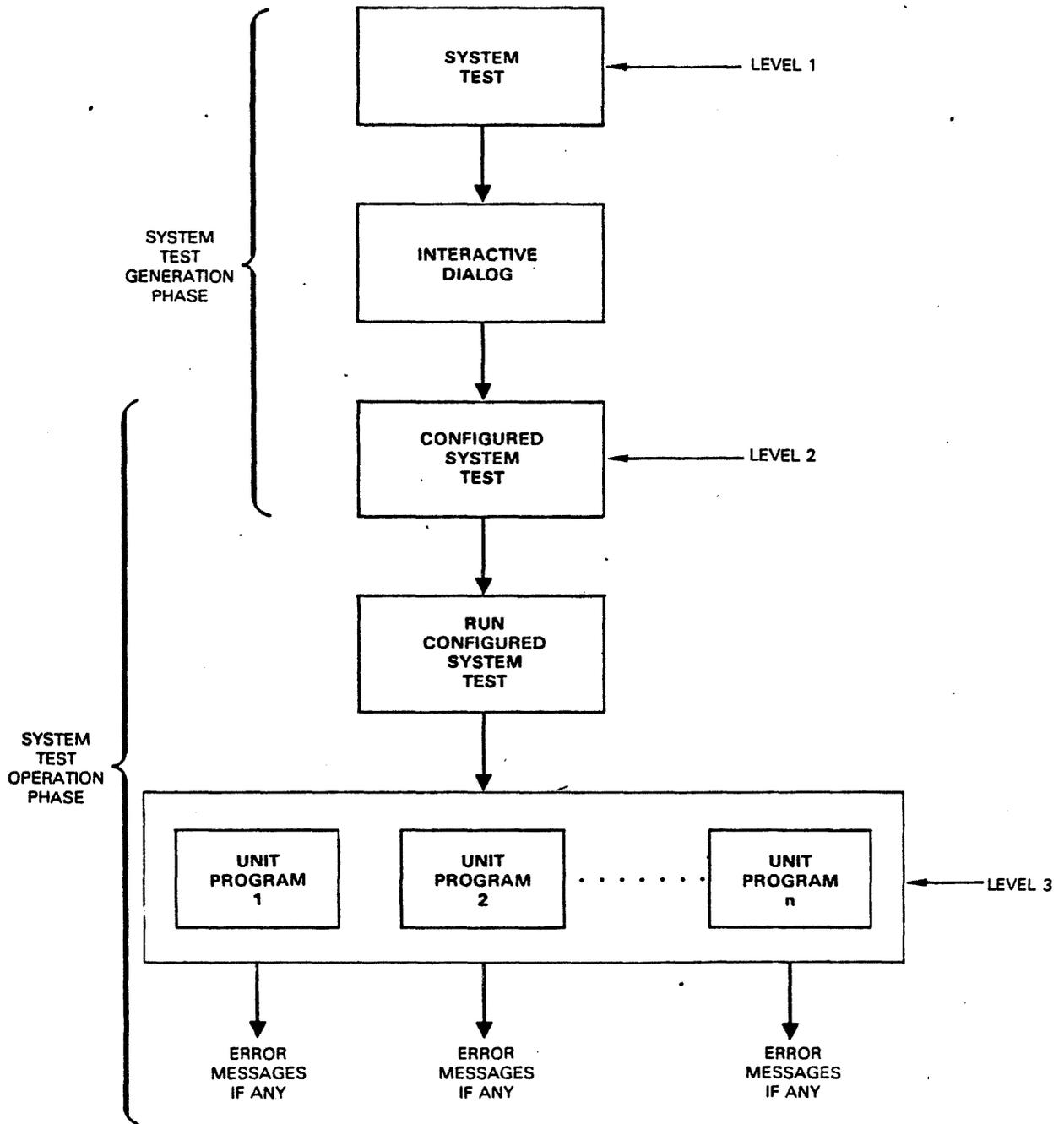
- It is an effective diagnostic tool which allows quick validation of system hardware.

- The operator is not required to have a detailed knowledge of the internal design of the system in order to operate the System Test.
- A simple procedure allows the operator to configure the System Test to meet the test requirements of a particular system. This procedure involves interactive dialog between the operator and the System Test Generator (which is a part of the System Test). The System Test Generator leads the operator step-by-step through the configuration process.
- For a given system configuration, the operator has to generate the configured System Test only once. The configured System Test can be saved on magnetic tape, and this tape can be used whenever System Test operation is required. Also, additional copies of the configured System Test can be easily made upon successful completion of System Test generation.
- Easy-to-use commands allow the operator to direct and interact with the test.
- Real-time, unit-level tests are available. Error messages output by these tests pinpoint failing mainframe or peripheral components in the system.
- Once activated, unit-level tests do not require any operator intervention. They run continuously unless the operator intervenes, and they automatically output any error messages that may be generated.
- The System Test provides the means for quick and concurrent validation of system hardware operation.
- A count of all the errors detected by each unit test is maintained.

Figure 1-1 is a simplified block diagram showing the three levels and two phases of the test. These concepts are discussed in greater depth in later sections of this manual.

1.4 RELATED PUBLICATIONS

The following Sperry Univac manuals are applicable to the use of the System Test.



UI1-4545

Figure 1-1. System Test: Operational Flow

| <u>Title</u> | <u>Document Number</u> |
|--|------------------------|
| VORTEX II Operating System Programmer Reference | UP-8677 |
| VORTEX II System Generation User Guide/ Programmer Reference | UP-9083 |
| VORTEX II Online Test User Guide | UP-9120 |
| MAINTAIN III Test Programs User Manual | UP-8672 |
| V70 Series Architecture Reference | UP-8634 |
| V77-600 System Reference | UP-8651 |
| V77-800 Computer System Description | UP-8701 |
| V77-800 Computer Operations Reference | UP-9028 |
| Peripheral Equipment Manuals* | |
| System Memo | |
| V70-Series System Test Software Performance Specification | SPS# 89A0470A |

1.5 NOTATIONS USED IN THIS MANUAL

The following notations are used throughout this document:

- \emptyset - Indicates a space character.
- Any number with at least one leading zero denotes an octal number.
- All completely capitalized words must be entered exactly as indicated.
- All lowercase words and words enclosed within angle brackets < > represent generic terms to be supplied by the user. They denote variable entries which depend upon the nature of the field.

* A peripheral equipment manual is provided for each peripheral device in the system.

- [] - Square brackets enclose optional elements.
- { } - Braces enclose options of mandatory fields, indicating that one of the options must be selected.
- <cr> - Indicates a carriage-return character. This is indicated in the manual for explanatory purposes only. It does not echo on the terminal.
- **Shaded text** - Represents text entered by the user (i.e., user input).

Section 2 System Overview

2.1 COMPONENTS OF THE SYSTEM TEST

The System Test consists of the following components:

- Preliminary Tests
 - Instructions Test
 - Memory Test
- Preliminary Loader
- System Test Generator
- Executive Program
- Library of Unit Programs

2.1.1 PRELIMINARY TESTS

2.1.1.1 Instructions Test

This test validates basic CPU operations. For a detailed discussion of this test, refer to Section 5 of this manual.

2.1.1.2 Memory Test

The Preliminary Memory Test is a 32K (map 0) memory test. It checks the operational status of the computer memory and detects faults. For more information about the Preliminary Memory Test, refer to Section 5 of this manual.

2.1.2 PRELIMINARY LOADER

The Preliminary Loader is a binary loader. Following successful execution of the Preliminary tests, the Preliminary Loader automatically loads the rest of the System Test. For more details about the Preliminary Loader, refer to Section 5.

2.1.3 SYSTEM TEST GENERATOR

The System Test Generator helps the operator tailor the standard System Test to meet the test requirements of his particular system. An interactive dialog takes place, during which the System Test Generator asks the operator a number of questions regarding the following:

- (a) The system on which the configured System Test is being generated (the SYS GEN CPU), and
- (b) the system on which the configured System Test will be executed (the SYS TEST CPU).

The two systems, namely, the SYS GEN CPU and the SYS TEST CPU, can be identical or different.

At the conclusion of the dialog, a configured System Test is generated and output. This test is configured specifically for the system which the operator intends to test.

2.1.4 EXECUTIVE PROGRAM

The Executive Program is the principal controlling component of the System Test. It has the following functions:

- Controls the execution of the configured System Test.
- Creates the environment in which unit programs operate.
- Schedules and maps unit programs.
- Maintains a count of errors detected by each unit program.
- Provides the user interface. Commands entered by the operator to direct the System Test are executed by the Executive Program.
- Performs all interrupt and I/O functions.

2.1.5 LIBRARY OF UNIT PROGRAMS

All available unit programs are contained in this library. To generate the configured System Test, the System Test Generator selects specific unit programs from this library. Unit programs are currently available to test the following system components:

- Writable Control Store (F3034) for V77-600
- Floating-Point Processor (F3027) for V77-600
- Writable Control Store (F3057) and Floating-Point Processor (F3053) for V77-800
- Main Memory
- Cartridge Disk and Controller
 - Models 70-7600 and 70-7610: (Type DB)*
 - F3094, F3096, F3310: (Type DF)*
- Disk Memory and Controller
 - 2822 (Type DC)*
 - Model 70-7510 (Type DD)*
 - 2823, 2824, F3092: (Type DE)*
- Direct Memory Disk Drive and Controller
 - 2825, 2826, 2842, 2843, 8433: (Type DH)*
- Magnetic Tape and Controller
 - Model 0 (F3088, F3089)
 - Model 1 (F3093, 0870)
- Line Printer and Controller
 - 2819, 2820, and 0786: (Printers)
 - F3361
- Data Communications Multiplexer (DCM)
 - DCM (F3000)
 - Line Adapters (LADs)
 - F3001-01, -02, -03, -04, -05.
 - F3006
 - Modem Controllers
 - F3002
 - F3003
 - F3004
 - F3005
 - F3006

* These are Software Model Codes used in Sysgen directives. Refer to VORTEX II System Generation User Guide, UP-9083.

When the System Test is loaded and executed, each selected unit program is mapped into a separate memory protection key and runs in multitask, interrupt-driven mode. When a unit program is finished, it is relocated into the next available memory protect key and rerun. Each unit program tests only one mainframe or I/O device option. Therefore, any error messages generated by a particular unit program will help the operator isolate the failing module or I/O device. The first line of the error message always identifies the unit program which generates it.

2.2 DESIGN GOALS AND MAINTENANCE CONCEPTS

The System Test helps the operator test and maintain Sperry Univac V77-600, and V77-800 minicomputers. It is designed to provide a quick validation of system hardware prior to running the VORTEX II Operating System. It should be run after the system has successfully executed the MAINTAIN III Test programs. (Refer to MAINTAIN III Test Programs User Manual, UP-8672).

NOTES

1. The System Test supplements the MAINTAIN III Test Programs. It is neither a replacement nor a substitute for the MAINTAIN III Test Programs.
2. The MAINTAIN III Test Programs do not have to be run each time the System Test is rerun.

The System Test has the following design features:

- It provides a real-time, mapped test environment for running unit-level test programs.
- It detects failing mainframe or peripheral components in the system, and automatically outputs error messages. Once activated, the unit programs do not require any operator intervention.
- It provides a means of generating a standalone system test for any VORTEX II system hardware configuration.

The configured System Test, which is generated from the standard System Test, is a real-time, multiprogrammed, mapped, standalone system test.

The Preliminary Instructions Test validates basic CPU operation. The Preliminary Memory Test checks the first 32K (map 0) of memory. The various unit programs verify the correct operation of individual system peripherals and their controllers, and central complex components (Writable Control Store, Floating-Point Processor and main memory).

Together, the MAINTAIN III programs and the System Test make effective fault isolation to the unit level possible.

2.3 OPERATIONAL FEATURES

Operation of the System Test has the following two phases:

1. The standard System Test is processed (by the System Test Generator) to create the configured System Test. This is the System Test Generation phase.
2. The configured System Test is then run on the computer system being tested. This is the System Test Operation phase.

System Test Generation is described in detail in Sections 3 and 4 of this manual. Procedures for System Test Operation are given in Section 6.

Section 3 System Test Generation

3.1 PURPOSE

The purpose of system test generation is to process the standard System Test, and generate from it a configured System Test. The configured System Test meets the specific test requirements of a given system configuration by including only those unit programs which are required. These unit programs are selected by the System Test Generator from the Library of unit programs, which is part of the standard System Test.

3.2 HARDWARE CONFIGURATION

The System Test requires certain hardware elements to execute properly. These required elements comprise the minimum hardware configuration.

The minimum hardware configuration for the System Test is as follows:

- V77-600 or V77-800 processor
- 64K main memory, consecutively addressed from 0
- Memory mapping facility
- Teletypewriter or keyboard-display terminal (CRT)
- Real-time clock
- Input peripheral device (magnetic tape unit)

The System Test supports the following central complex and peripheral device options:

- Central complex options
 - Floating-Point Processor (FPP)
 - Writable Control Store (WCS)
 - Cache (standard with V77-800, and optional with V77-600)

- Peripheral device options
 - Disk
 - Magnetic tape
 - Line printer
 - Teletypewriter
 - Keyboard-display terminal
 - Data Communications Multiplexer

3.3 OVERVIEW OF THE PROCESS

Basically, the System Test generation process consists of the following steps:

1. Loading the Preliminary Loader
2. Error detection
3. Loading the System Test (performed by the Preliminary Loader)
4. Configuration dialog
5. Error detection during configuration dialog
6. Preparation of the configured System Test.

The configured System Test is generated upon successful completion of the System Test generation process.

3.4 SAMPLE SYSTEM TEST GENERATION

A step-by-step procedure for generating a configured System Test follows.

3.4.1 PREPARATION

This procedure is designed for execution of any V77-Series system which is configured for VORTEX II*. However, on any given system, there are certain parameters which are applicable to that particular system. Steps involving such system-dependent information are flagged as such, and directions are given to configure any desired (or required) system.

* Currently implemented only for V77-600 and V77-800 computers.

The standard System Test is available on magnetic tape. A variety of loading devices are available for use with Sperry Univac computers. Before using the following procedures, the operator should determine what equipment has been installed at his installation, and then obtain the system-dependent information and procedures by referring to the appropriate manuals.

Refer to V77-600 System Reference (UP-8651) for V77-600 operation procedures. Refer to V77-800 Computer Operations Reference (UP-9028) for V77-800 operation procedures.

1. Turn on computer power.
2. Ensure that all devices are online.
3. Reset Sense Switches 1, 2, and 3.
4. Initialize the computer control circuits by pressing RESET.
5. Mount the standard System Test magnetic tape on the tape drive unit at proper density. Advance the tape to load point. Bring it online.

CAUTION

The above procedure does not include system-dependent information such as bootstrap loading, operator interaction, etc. Refer to the appropriate manuals to obtain this information, before beginning System Test generation.

3.4.2 LOADING THE PRELIMINARY LOADER

The Preliminary Loader can be loaded into memory using the following magnetic tape bootstrap routine:

| | | | |
|------|--------|---|--|
| 0200 | 1031BE | ← | For BE supply the BIC EVEN ADDRESS |
| 0201 | 005301 | | |
| 0202 | 1031B0 | ← | For B0 supply the BIC ODD ADDRESS |
| 0203 | 1000BE | | |
| 0204 | 1000DA | ← | For DA supply the DEVICE ADDRESS |
| 0205 | 1012DA | | |
| 0206 | 007002 | | |
| 0207 | 005000 | | |
| 0210 | 001000 | | |
| 0211 | 000205 | | |
| 0212 | 1000B0 | | |
| 0213 | 104UDA | ← | For U supply the magnetic tape unit number (1 to 4) |
| 0214 | 005141 | | |
| 0215 | 001000 | | |
| 0216 | 000200 | | |

(The symbolic coding for this bootstrap routine is presented in Table 3-1). Note that column 1 (0200, 0201, etc.) values are memory addresses, and column 2 values are the actual instructions of the routine.

The magnetic tape bootstrap routine has to be manually loaded into memory through the virtual console, or the computer control panel, and then executed to load the Preliminary Loader.

The V77-600 control panel contains all the switches and indicators needed for computer operation. The V77-800 virtual console consists of either a teletypewriter or a CRT terminal that interfaces directly with the V77-800 computer through console interface circuits in the processor.

Use one of the following procedures to manually enter the magnetic tape bootstrap routine.

3.4.2.1 Manual Loading Via the Computer Control Panel (V77-600 Computers only)

In the case of the V77-600 computer, the bootstrap routine can be manually entered via the computer control panel. Refer to the V77-600 System Reference, UP-8651, for these procedures.

1. Enter the instructions for the bootstrap routine into memory starting at address 0200.

Table 3-1. Magnetic Tape Bootstrap Routine with Symbolic Coding

```

*****
*****
*
* START AT 0212 WITH X=07000
*
* 0200 1031BE BOOT OAR BIC SET INITIAL ADDRESS
* 0201 005301 DECR 1 A=177777 FOR LAST ADDRESS
* 0202 1031BO OAR BIC+1 SET FINAL ADDRESS
* 0203 1000BE EXC BIC ACTIVATE BIC
* 0204 1000DA EXC DA READ ONE RECORD FROM MT
* 0205 1012DA WAIT SEN 0200+DA,07002 START IF MT READY
* 0206 007002
* 0207 005000 NOP
* 0210 001000 JMP WAIT WAIT LOOP
* 0211 000205
*
* START HERE WITH X=07000
*
* 0212 1000BO START EXC BIC+1 INITIALIZE BIC
* 0213 104UDA EXC2 (U*0100)+DA SELECT MT UNIT
* 0214 005141 INCR 041 SET A=07001 (START ADDR)
* 0215 001000 JMP BOOT START BIC
* 0216 000200
*****
* BE = BIC ADDRESS,EVEN
* BO = BIC ADDRESS,ODD
* U = TAPE UNIT(1-4)
* DA = MT DA
*****
MT EQU 010
BIC EQU 020
U EQU 01
*****

```

2. Enter 000 212 into the P register.
3. Enter 07000 into the X register.
4. Enter zero in the A and B registers.
5. Press the STEP/RUN switch. This causes the STEP indicator to go out and the RUN indicator to blink. The computer is now ready to enter the run mode when the START switch is pressed.

Step 1 of this procedure causes the bootstrap routine to be loaded. The other four steps execute the bootstrap routine which loads the Preliminary Loader.

3.4.2.2 Manual Loading Via the Virtual Console (V77-800 Computers Only)

Refer to V77-800 Computer Operations Reference, UP-9028, for information about virtual console operation.

1. Manually enter the magnetic tape bootstrap routine on the virtual console.
2. Enter 0212 into the P register.
3. Enter 07000 into register R2.
4. Enter zeros in registers R0 and R1.
5. Press key R. This places the processor in the run mode. The execution starting address in memory is specified by the P register. This command causes that instruction to be executed first.

Step 1 of this procedure loads the bootstrap routine. The other four steps execute the routine. The routine loads the Preliminary Loader.

3.4.3 ERROR INDICATIONS AND HANDLING

During the execution of the Preliminary Loader, the Preliminary Tests, namely, the Instructions test and the Memory test, are executed. Any error detected by these tests results in the following:

1. The computer is forced into STEP mode, and

2. An error code (value in the range 01-052) is returned in the instruction register.

For a discussion of the meanings of these error codes, refer to Section 5 and Table 5-2.

Once such an error occurs, nothing further can be done at the system level. Run the MAINTAIN III diagnostics to isolate the fault. (Refer to MAINTAIN III Test Programs User Manual, UP-8672).

If, however, the preliminary tests are successfully executed, the Preliminary Loader proceeds to load the rest of the standard System Test.

Any error detected at this stage results in the following:

1. The message:

LOAD ERROR

is output to the operator's console;

2. The computer is placed in STEP mode, and
3. The error code '077' is returned in the instruction register.

If this happens, no recovery is possible. Reload and start again.

Error indications during System Test Generation are discussed in detail in Section 4.

3.4.4 CONFIGURATION INTERROGATION

After it is loaded, the System Test Generator outputs messages to the operator on the operator's console, and leads the operator step-by-step through the configuration process. An interactive dialog takes place between the System Test Generator and the operator. The aim of the dialog is to define the system hardware configuration of the following:

1. The system on which the configured System Test is being generated (the SYS GEN CPU), and
2. The system on which the configured System Test will be executed (the SYS TEST CPU).

The two systems, namely the SYS GEN CPU and the SYS TEST CPU, can be identical or different.

When the interrogation is completed, a configured System Test, specifically tailored to the system that is to be tested, is generated. This is output to the device specified as the SYS GEN OUTPUT DEVICE during the interrogation.

The configuration dialog is described in detail in Section 4 of this manual.

Section 4 Configuration Dialog

4.1 GENERAL

A series of configuration questions are output by the System Test Generator on the operator's console. This is the process of configuration interrogation. Jointly, these configuration questions and the operator's responses to them form the configuration dialog.

Most of the information that the operator requires for answering the configuration questions can be obtained from the System Memo which is shipped with each system. Some of the information must be obtained from other manuals. These manuals are identified in the appropriate places in this section.

4.2 SYSTEM TEST CONFIGURATION GUIDE

The System Test Configuration Guide is a checklist for use in preparing the correct responses to configuration questions. Complete the checklist before attempting to answer the configuration questions. An explanation of how the System Test Configuration Guide should be used, and where the required information is obtained from, is provided in Appendix A. The operator can use the blank checklist to prepare for the configuration dialog.

The configuration questions have the following features:

- Each configuration question is numbered. (There are 19 configuration questions).
- System questions (as opposed to unit program questions) are marked with asterisks (*). Questions 1 through 7 are system questions.
- Each configuration question has the following two parts:
 1. The body of the question.
 2. Acceptable user replies enclosed within parentheses. Some of the questions also have default values.

These default values are also specified within the parentheses, with a semicolon separating them from the preceding acceptable user replies.

NOTE

The following example is included here merely to explain the format common to all the configuration questions. Each configuration question is discussed in detail later in this section.

Example:

In the configuration question:

CNFG?(H,A,1,2,...;A)
① ② ③

① The user is asked to select configuration questions.

② Any one of the following replies is an acceptable response:

- H
- A
- 1
- 2
- 3
- .
- .
- .
- 19

NOTE

Questions 1 through 7 are always output irrespective of whether the operator does or does not specify these numbers.

- Any combination of these values, each value being separated from the next by a comma.
- <cr> or "." which specifies the default value (A).

③ The value following the semicolon is the default value.

- If the user does not specify a value but merely enters a <cr> or a "." as the response to a question, the default value is automatically used.
- In cases where the default value is not explicitly defined, the default value is assumed to be zero.

User responses should conform to the following:

- Each user response must be terminated with a <cr> or a ".".
- Whenever the user inputs an unacceptable reply, two question marks (??) are output by the system. The user must then enter a reply which is acceptable to the system.
- The leading zero, which indicates an octal number, may be omitted.
- To start again at the beginning of the configuration interrogation, enter a Control-C character as the response to a numbered question.
- If the response to a question is incorrectly entered:
 1. Type a back-arrow (←) to delete the previous character, or
 2. Type a back-slash (\) to delete the entire response.

Reenter the response correctly.

Configuration questions 1 through 7 are system questions, concerning devices and internal computer options that are part of the SYS TEST and SYS GEN systems. Questions 8 through 19 concern unit programs for specific peripheral units or central complex options.

Example:

The purpose of the following question:

8.WCS-UP1,15?(Y;N)

is to determine whether or not the WCS unit program is to be included in the configured System Test. The UP-numbers (UP1,15) are the unit program identification numbers. (Refer to Table 5-3 for a Unit program name-ID-number cross-reference.) Unit programs are identified by name as well as by number in configuration questions 8 through 19.

4.3 CONFIGURATION QUESTIONS

In this section, each configuration question is individually discussed. For each configuration question, the following are explained:

1. Question identification.
2. Purpose of the question.
3. Acceptable replies that the operator can enter.
4. Example.

A listing obtained during an actual System Test Generation is presented in Appendix A. Examples of operator responses shown in this section are keyed to this listing.

For a detailed explanation of how to prepare configuration responses, refer to Appendix A.

4.3.1 INTRODUCTORY CONFIGURATION QUESTION

V70-SERIES SYSTEM TEST GENERATION

SELECT CONFIGURATION QUESTIONS

NOTE:ACCEPTABLE REPLIES ARE IN ()

DEFAULT=0 OR VALUE AFTER ;

END EACH LINE WITH CARRIAGE RETURN OR PERIOD

IF REPLY IS OMITTED, DEFAULT VALUE IS USED

A=ALL,H=HELP,N=NO OR NONE,Y=YES

ENTER SPACE IF OUTPUT PAUSES

* INDICATES PRESELECTED SYSTEM QUESTION

LEADING 0=OCTAL,MAY BE OMITTED

CNFG?(H,A,1,2,...;A)

4.3.1.1 Description

Upon successful loading of the System Test Generator, a message is output to the operator's console, which provides the following information:

- a. Header
- b. Instruction on how to proceed
- c. Self-explanatory tutorial information
- d. A question which asks the operator to select the configuration questions he wants to answer.

4.3.1.2 Acceptable Replies

The question "CNFG?" prompts the user to select the configuration questions he wants to answer.

Acceptable replies to this question are:

H,A,1,2,..19,

where:

H

Is HELP. Type H to obtain a listing of all the configuration questions.

A

Is ALL. Type A to select all the configuration questions. If A is selected, after the operator answers each question satisfactorily, the next question is output automatically. A is the default value. Questions 1 through 7 are always output irrespective of whether the operator does or does not specify them.

1,2,...

All the configuration questions are numbered. The operator can ask for specific questions by number. He can specify one or more questions. Each number must be separated from the next by a comma.

{<cr>
 . }

Specifies "A" the default value.

4.3.1.3 Example

Typing the following:

3,5,9.

results in eight questions (nos. 1 through 7, and 9) being output to the console.

① In the sample listing shown (A.3), the operator response is "H".

② After a list of all the configuration questions is output, the operator responds by entering "A".

4.3.2 CONFIGURATION QUESTION NUMBER 1

The first configuration question is:

*1.SYS TEST CPU,SYS GEN CPU?(H,0-4,0-4)

The asterisk identifies this as a system question.

4.3.2.1 Purpose

The question asks the operator to state the types of computer (V77-800, V77-600, etc.) being used as:

- a. SYS GEN CPU - The computer being used for the system test generation, and
- b. SYS TEST CPU - The computer which is going to be tested.

4.3.2.2 Acceptable Replies

Any one of the following responses is acceptable:

H

Type H to obtain a listing of the available CPU types. The system responds with:

0 = V77-800
1 = V77-600
2 = V77-400**
3 = V76**
4 = V75**

0-4,0-4

The first part of this response (any number in the range 0 to 4) identifies the type of the SYS TEST CPU. The second part (any number in the range 0 to 4) identifies the type of the SYS GEN CPU. The two numbers are separated by a comma. Terminate the response with a <cr> or ".".

** Currently not implemented

{<cr>
.

Specifies 0,0 since no default values are explicitly defined.

4.3.2.3 Example

③ In the sample listing, the operator response to configuration question number 1 is:

1,1.

This indicates that:

- a. The SYS GEN CPU is a type 1 (V77-600) computer.
- b. The SYS TEST CPU is also a type 1 (V77-600) computer.

NOTES

1. The SYS GEN CPU and the SYS TEST CPU do not have to be of the same type.
2. Both CPUs must be individually specified, even if they are of the same type.

CAUTION

Do not specify a type 2 (V77-400), type 3 (V76), or type 4 (V75) computer. The System Test is currently implemented only for type 0 (V77-800) and type 1 (V77-600) computers.

4.3.3 CONFIGURATION QUESTION NUMBER 2

*2.START,END ADR(K) OF EACH MEM BLOCK?(S1,E1 S2,E2 ...;0,256)

4.3.3.1 Purpose

This is a system question. It asks the operator to specify the start and end address of each memory block being used in the SYS TEST CPU.

4.3.3.2 Acceptable Replies

NOTE

The first 64K words of memory must be continuously addressed from 0.

The start and end addresses for each noncontiguous memory block must be specified in the following format:

S1,E1 S2,E2 S3,E3 ... Sn,En.

where:

- S1 Is the start address of the first memory block.
- E1 Is the end address of the first memory block.
- S2 Is the start address of the second memory block.
- E2 Is the end address of the second memory block.
- Sn Is the start address of the nth memory block.
- En Is the end address of the nth memory block.

NOTES

1. The default value of the start address of the memory block is 0. The default value of the end address of the memory block is 256K words. If the operator responds with a <cr> or a ".", without specifying any start or end addresses, these default values will be automatically used.
2. The values specified in the replies are decimal values.

4.3.3.3 Example

④ In the sample listing, the operator responds with a ".", and so the default values are automatically used.

4.3.4 CONFIGURATION QUESTION NUMBER 3

*3.MEMORY PARITY INTERRUPT ADR?(010-077;060)

4.3.4.1 Purpose

This is also a system question. The operator is asked to specify the memory parity interrupt address. The memory parity interrupt address is variable on the V77-600 computer.

4.3.4.2 Acceptable Replies

Any value from 010 to 077 is an acceptable response. The information required to answer this question can be obtained from the System Memo, as explained in Appendix A. The default value is 060.

NOTE

If the system does not have any parity,
use the default value by entering a
".".

4.3.4.3 Example

⑤ In the sample listing, the operator responds with a ".". So the default value of 060 is used as the memory parity interrupt address.

4.3.5 CONFIGURATION QUESTION NUMBER 4

*4.SYSTEM TEST OPERATOR'S CONSOLE
DEVICE ADDRESS?(0-076; 01)
READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)
WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)

4.3.5.1 Purpose

This configuration question is a system question about the operator's console of the system to be tested. After the operator responds to one question, the next question is output.

4.3.5.2 Acceptable Replies

Any value from 0 to 076 can be specified as the device address. The default value is "01". Any even number from 0100 to 0276 can be specified as the interrupt addresses corresponding to READ READY and WRITE READY completion status.

The information required to answer this question can be obtained from the System Memo. (Refer to Appendix A.)

4.3.5.3 Example

- ⑥ In the sample listing, the operator specifies the following:
1. A Device Address of 1 (by default).
 2. A READ READY COMPLETE interrupt address of 154.
 3. A WRITE READY COMPLETE interrupt address of 156.

4.3.6 CONFIGURATION QUESTION NUMBER 5

*5.SYSTEM TEST LIST DEVICE?(0=TTY/CRT,1=LPR,2=BOTH;2)
PRINTER X:
DEVICE ADDRESS?(0-076; 35)
TTY OR CRT X:
DEVICE ADDRESS?(0-076; 01)

4.3.6.1 Purpose

This is a group of three questions about the system device selected for listing commands, messages, and dumps. If no list device is specified, the operator's console is used by default.

4.3.6.2 Acceptable Replies

Enter one of the following responses to the first question in the group, depending upon which device is selected to be the System Test List device:

0

Specifies the teletypewriter or keyboard-display terminal (TTY/CRT). Only the question related to the TTY/CRT will be output subsequently.

1

Specifies the line printer (LPR). If this is specified, only the question about the line printer will be subsequently output.

2

Specifies both the line printer and the teletypewriter (or keyboard-display terminal).

{<cr>
.}

Specifies the default value "2".

The Device Address of the line printer can be any number from 0 to 076. The default value is "35". The Device Address of the TTY or CRT can be any number from 0 to 076. The default value is "01".

4.3.6.3 Example

⑦ In the sample listing, both the TTY/CRT and the LPR are chosen as System Test List devices. The Device Address of the LPR is specified as 35, by default. The Device Address of the TTY/CRT is specified as 1, by default.

4.3.7 CONFIGURATION QUESTION NUMBER 6

*6.SYSTEM GEN INPUT DEVICE?(0=PT,1=MT,2=DS,3=CR;1)
DEVICE ADDRESS?(0-076; 10)
BIC/BTC EVEN DEVICE ADDRESS?(0-076)
UNIT NO?(0-3)

4.3.7.1 Purpose

This is a system question concerning the device from which the
it programs are loaded during system test generation.

NOTE

Although paper tape (PT), magnetic tape
(MT), disk (DS), and card reader (CR)
units are listed as options, currently
only the magnetic tape device is
supported.

After the operator specifies the device, he is asked to provide
more information about the device address, the BIC/BTC device
address, and the unit number.

4.3.7.2 Acceptable Replies

Enter one of the following responses to the first question
concerning specification of a system generation input device:

- 0 Specifies paper tape.
- 1 Specifies magnetic tape.**
- 2 Specifies disk.

** Only this option is currently supported and is the default
option.

3

Specifies card reader.

{<cr>
.}

Specifies the default value "1".

The device address can be any value to 076. The default value is "10". The BIC/BTC device address can be any even number from 0 to 076. The unit number can be any number from 0 to 3.

4.3.7.3 Example

⑧ In the sample listing shown, the system generation input device is magnetic tape. The device address of the magnetic tape unit is specified to be 10, and the BIC/BTC even device address is 24. Since the operator enters a "." as the response to the question about the unit number, the value "0" is assumed.

4.3.8 CONFIGURATION QUESTION NUMBER 7

*7.SYSTEM GEN OUTPUT DEVICE?(0=PT,1=MT,2=DS,3=CP;1)
DEVICE ADDRESS?(0-076; 10)
BIC/BTC EVEN DEVICE ADDRESS?(0-076)
UNIT NO?(0-3)

4.3.8.1 Purpose

This is a system question about the device to which the configured system test is output. Although paper tape (PT), magnetic tape (MT), disk (DS), and card punch (CP) are listed as options, currently only magnetic tape is supported.

After the operator specifies the system generation output device, he is asked to provide more information about that particular device: the device address, the BIC/BTC even device address, and the unit number.

4.3.8.2 Acceptable Replies

Enter one of the following responses to the first question concerning device specification:

- 0 Specifies paper tape.
- 1 Specifies magnetic tape.**
- 2 Specifies disk.
- 3 Specifies card punch.

{<cr>
.}

Specifies the default value: "1".

The device address can be any value from 0 to 076. The default value is "10". The BIC/BTC even device address can be any even number from 0 to 076. The unit number can be any number from 0 to 3.

** Only this option is currently supported and is the default option.

NOTE

The system generation output device can be the same type of device as the system generation input device, or it can be of a different type.

4.3.8.3 Example

⑨ In the sample listing, both the system generation input device and the system generation output device are magnetic tape devices. The device address of the system generation output device is 11. The BIC/BTC even device address is 22. The magnetic tape unit number is 0.

4.3.9 CONFIGURATION QUESTION NUMBER 8

8.WCS-UP1,15?(Y;N)
EVEN DEVICE ADDRESS?(0-076; 74)

4.3.9.1 Purpose

The purpose of this question is to determine whether or not the Writable Control Store (WCS) unit program must be included in the configured System Test.

NOTE

The WCS unit program is identified by name and by number in this question.

4.3.9.2 Acceptable Replies

Enter one of the following:

Y

Specifies that the WCS unit program is to be included.

N

Specifies that the WCS unit program need not be included. This is the default answer.

{<cr>
.}

Specifies that the default answer "N" is to be used.

If the operator answers "Y", he is questioned about the (even) device address of the WCS. Any even number from 0 to 076 can be specified. The default value is: "74".

4.3.9.3 Example

⑩ In the sample listing, the operator indicates that the WCS unit program is to be included in the configured System Test, and specifies the default value 74 as the even device address.

4.3.10 CONFIGURATION QUESTION NUMBER 9

9.FLOATING POINT PROCESSOR-UP2,15?(Y;N)

4.3.10.1 Purpose

The purpose of this question is to determine whether or not the Floating-Point Processor (FPP) unit program is to be included in the configured System Test. The unit program is identified by name and number.

NOTE

For the V77-800 computer, the FPP unit program is combined with the WCS unit program. For the V77-600 computer, there are two separate unit programs: one for the FPP and one for the WCS.

4.3.10.2 Acceptable Replies

Enter one the following:

Y

Specifies that the FPP unit program is to be included.

N

specifies that the FPP unit program need not be included. This is the default answer.

{<cr>
.}

Specifies that the default answer "N" is to be used.

4.3.10.3 Example

⑪ In the sample listing, the operator indicates that the FPP unit program is to be included in the configured System Test.

4.3.11 CONFIGURATION QUESTION 10

10.MEMORY TEST-UP3?(Y;N)

4.3.11.1 Purpose

This question determines whether or not the memory test unit program is included in the configured System Test.

4.3.11.2 Acceptable Replies

Enter one of the following:

Y

Specifies that the unit program is to be included.

N

Specifies that the unit program need not be included.
This is the default answer.

{<cr>
.}

Specifies that the default answer "N" is to be used.

4.3.11.3 Example

⑫ In the sample listing, the operator indicates that the memory test unit program is to be included in the configured System Test.

4.3.12 CONFIGURATION QUESTION NUMBER 11

11.HOW MANY DISC CONTROLLERS-UP4,5,6,16?(0-4)
DISC 1:
MODEL?(H,0-6;5)
DEVICE ADDRESS?(0-076; 16)
BIC/BTC EVEN DEVICE ADDRESS?(0-076)
BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)
EACH SEEK COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)
WHICH TRACK(T=0-0312/0625),SECTOR(S=0-2)?(T,S)
WHICH PLATTERS(P=0-1) & UNITS(U=0-3)?(PU,...)
DISC 2:
MODEL?(H,0-6;5)
DEVICE ADDRESS?(0-076; 15)
BIC/BTC EVEN DEVICE ADDRESS?(0-076)
BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)
SEEK COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)
WHICH HEAD?(0-023)
WHICH UNITS?(0-3/7)
DISC 3:
MODEL?(H,0-6;5)
DEVICE ADDRESS?(0-076; 14)
COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)
WHICH CYLINDER(C=0-0632/01466),HEAD(H=0-4/022)?(C,H)
WHICH UNITS?(0-3/7)
DISC 4:
MODEL?(H,0-6;5)
DEVICE ADDRESS?(0-076; 12)
BIC/BTC EVEN DEVICE ADDRESS?(0-076)
BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)
CONTROLLER COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)
WHICH TRACK(T=0-0114),SECTOR(S=0-033)?(T,S)
WHICH UNITS?(0-3/7)

NOTE

The questions output after the first question about the model number depend upon the model specified by the operator.

4.3.12.1 Purpose

This question concerns the disk controllers for which unit programs have to be included in the configured System Test. A maximum number of 4 disk controllers can be tested at a time. For each disk controller, questions are asked about the following:

- a. Model
- b. Device Address
- c. BIC/BTC Even Device Address
- d. BIC/BTC COMPLETE Even Interrupt Address
- e. Each SEEK COMPLETE Even Interrupt Address.

Additional questions are asked, which depend upon the type of disk specified.

4.3.12.2 Acceptable Replies

For the question about the number of disk controllers, any number from 0 to 4 can be specified.

The next question is:

```
DISC1:
MODEL?(H,0-6;5)
```

Enter one of the following:

H To obtain a list of all available models.

0-6 Any model number code from 0 to 6 depending upon the type of the disk being used.

```
{<cr>}
.
```

To use the default value 5, which corresponds to a type DH disk.

The following list of disk types and their corresponding model number codes is output when the operator enters "H":

(0=TYPE DB)

(1=TYPE DC)

(2=TYPE DD)

(3=TYPE DE)

(4=TYPE DF)

(5=TYPE DH)

(6=TYPE DJ)

Refer to the System Configuration Guide (Appendix A) and Section 2 for a cross-reference list of disk types and type and feature numbers.

Questions that follow the first question depend upon the model number specified by the operator.

For Models 0 and 4, the following questions are asked:

DEVICE ADDRESS?(0-076;16)
BIC/BTC EVEN DEVICE ADDRESS?(0-076)
BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)
EACH SEEK COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)
WHICH TRACK(T=0-0312/0625),SECTOR(S=0-2)?(T,S)
WHICH PLATTERS(P=0-1) & UNITS(U=0-3)?(PU,...)

Each unit has a unique seek complete interrupt address, each of which must be furnished by the operator (that is, 110, 114, 116). Any track and sector within the specified limits can be selected for the test run. Only the selected address can be used during the test. When more than one upper limit is given in the question, (for example, 0312/0625), the selection depends upon the disk type. For Model 0 disks, specify 0312. For Model 4 disks, specify 0625. Each platter and corresponding unit in the system to be tested must be specified. The platter is the actual physical recording surface inside a unit. A unit is the actual physical drive. Each unit can have 1, 2, or 4 platters. A cross-reference list of disk types (Feature numbers) and the number of platters in each unit is given in Table 4-1.

Table 4-1. Model 4 (Type DF) Disks and Corresponding Platter Numbers

| Disk Type (Feature Numbers) | Number of Platters Per Unit |
|--------------------------------|--------------------------------|
| F3094 | 2 |
| F3096 | 1 |
| F3310 | 2 |

For Models 1, 2, and 3, the following questions are asked:

DEVICE ADDRESS?(0-076;15)
BIC/BTC EVEN DEVICE ADDRESS?(0-076)
BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)
SEEK COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)
WHICH HEAD?(0-023)
WHICH UNITS?(0-3/7)

The current arm position determines the cylinder selected for the test run. No arm motion is performed during the test. A recalibration to cylinder 0 occurs automatically when the disk is powered up. Any head within the specified limits can be selected for the test run. Only the selected address is used during the test. Each unit in the system to be tested must be specified. The upper limit depends upon the disk type.

For Model 5 disks, the following questions are asked

EVEN DEVICE ADDRESS?(0-076; 14)
COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)
WHICH CYLINDER(C=0-0632/01466),HEAD(H=0-4/022)?(C,H)
WHICH UNITS?(0-3/7)

Each unit in the system to be tested must be specified. The upper limit (3 or 7) depends upon the type of disk.

Device-specific information required to answer the above questions must be obtained from the appropriate peripheral equipment manuals.

4.3.12.3 Examples

Example 1:

⑬ In the sample listing, the operator specifies that unit programs are to be included for two disk controllers. The two disks being tested are of type DF and DC. For the type DF (Model 4) disk, track 0 and sector 0 are specified (by default); platter 0 of unit 0 is specified (by default). For the type DC (Model 1) disk, head 0 and unit 0 are specified by default.

Example 2:

```
DISC 1:
MODEL?(H,0-6;5)4.
DEVICE ADDRESS?(0-076; 16)
BIC/BTC EVEN DEVICE ADDRESS?(0-076)20.
BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)100.
EACH SEEK COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)110.
WHICH TRACK(T=0-0312/0625),SECTOR(S=0-2)?(T,S) 100,0.
WHICH PLATTERS(P=0-1) & UNITS(U=0-3)?(PU,...)00,10.
```

A Model 4 (type DF) disk is specified by the operator. The device address is 016; the BIC/BTC even device address is 020; the BIC/BTC COMPLETE even interrupt address is 0100; and the SEEK COMPLETE even interrupt address is 0110. Track 0100 and sector 0 are specified. In response to the question about the number of platters and units, the operator response indicates the following:

- Platter 0 of unit 0; and
- Platter 1 of unit 0.

Example 3:

```
DISC 2:
MODEL?(H,0-6;5)1.
DEVICE ADDRESS?(0-076; 15)
BIC/BTC EVEN DEVICE ADDRESS?(0-076)50.
BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)140.
SEEK COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)142.
WHICH HEAD?(0-023)
WHICH UNITS?(0-3/7).
```

The disk is a Model 1 (type DC) disk. The device address is 015 by default; the BIC/BTC even device address is 050; the BIC/BTC COMPLETE even interrupt address is 0140; and the SEEK COMPLETE even interrupt address is 0142. Head 0 and unit 0 are specified (default values).

Example 4

DISC 3:
MODEL?(H,0-6;5).
EVEN DEVICE ADDRESS?(0-076; 14) .
COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)160.
WHICH CYLINDER(C=0-0632/01466),HEAD(H=0-4/022)?(C,H) .
WHICH UNITS?(0-3/7).

The disk is a Model 5 (type DH) disk. The even device address is 014. The COMPLETE even interrupt address is 0160. Cylinder 0, head 0, and unit 0 are specified (the default values).

4.3.13 CONFIGURATION QUESTION NUMBER 12

12. HOW MANY MAG TAPE CONTROLLERS-UP? (0-4)

MAG TAPE 1:

MODEL?(H,0,1)

DEVICE ADDRESS?(0-076; 10)

BIC/BTC EVEN DEVICE ADDRESS?(0-076)

BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)

MOTION COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)

WHICH UNITS?(0-3/7)

MAG TAPE 2:

MODEL?(H,0,1)

DEVICE ADDRESS?(0-076; 11)

BIC/BTC EVEN DEVICE ADDRESS?(0-076)

BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)

MOTION COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)

WHICH UNITS?(0-3/7)

MAG TAPE 3:

MODEL?(H,0,1)

DEVICE ADDRESS?(0-076; 12)

BIC/BTC EVEN DEVICE ADDRESS?(0-076)

BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)

MOTION COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)

WHICH UNITS?(0-3/7)

4.3.13.1 Purpose

This question concerns the magnetic tape controllers for which unit programs have to be included in the configured System Test. A maximum number of 4 magnetic tape controllers can be tested at a time. One unit program has to be included for each controller. For each controller, questions are asked about the following:

- a. Model;
- b. Device Address;
- c. BIC/BTC Even Device Address;
- d. BIC/BTC COMPLETE Even Interrupt Address;
- e. MOTION COMPLETE Even Interrupt Address;
- f. Units.

4.3.13.2 Acceptable Replies

For the question about the number of magnetic tape controllers, specify any number from 0 to 4.

The next question is:

MAG TAPE 1:
MODEL?(H,0,1)

Enter one of the following responses:

H
To obtain a list of the models supported by the System Test.

{ 0 }
{ 1 }

To specify the type of magnetic tape controller.

The following list is output when the operator enters "H":

0 = NO STATUS WORD
1 = USES STATUS WORD

(Refer to the System Configuration Guide (Appendix A) and Section 2 for a cross-reference list of magnetic tape models and type and feature numbers).

The following is a cross-reference list of magnetic tape type/feature numbers and status word usage by the controller.

| <u>Magnetic Tape</u> <u>T/F Number</u> | <u>Status Word Usage</u> |
|---|--------------------------|
| 0870 | Uses status word |
| F3088 | No status word |
| F3089 | No status word |

The next question is:

DEVICE ADDRESS?(0-076; 10)

Any number from 0 to 076 is acceptable. The default value of the device address for magnetic tape number 1 is 10.

The default value of the device address for magnetic tape number 2 is 11, for magnetic tape number 3 is 12, and so on.

Any even number from 0 to 076 can be specified as the BIC/BTC EVEN DEVICE ADDRESS.

Any even number from 0100 to 0276 can be specified as the BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS.

NOTE

This interrupt is not used by VORTEX and may not be defined in the System Memo. If this is the case, enter the MOTION COMPLETE EVEN INTERRUPT ADDRESS.

Any even number from 0100 to 0276 can be specified as the MOTION COMPLETE EVEN INTERRUPT ADDRESS.

Each unit in the system must be specified. The upper limit (3/7) depends on the type of magnetic tape.

This set of six questions is repeated for each one of the magnetic tape controllers. Device-specific information required to answer the above questions must be obtained from the appropriate peripheral equipment manuals.

4.3.13.3 Example

⑭ In the sample listing, it is specified that unit programs are to be included for 3 magnetic tape controllers. All three are Model 0 units. In the cases where the operator responds with a ".", either an explicitly-defined default value or, when the former is absent, the value "0" is used.

4.3.14 CONFIGURATION QUESTION NUMBER 13

13.HOW MANY LINE PRINTERS-UP8?(0-4)

PRINTER 1:

DEVICE ADDRESS?(0-076; 35)

BIC/BTC EVEN DEVICE ADDRESS?(0-076)

BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)

4.3.14.1 Purpose

This question concerns the line printers for which unit programs have to be included in the configured System Test. The maximum number of line printers that can be tested at a time is 4. For each line printer, questions are asked about the following:

- a. Device Address;
- b. BIC/BTC Even Device Address;
- c. BIC/BTC COMPLETE Even Interrupt Address.

4.3.14.2 Acceptable Replies

For the question about the number of line printers, specify any number from 0 to 4.

For each line printer, the following set of three questions is output:

DEVICE ADDRESS?(0-076; 35)

BIC/BTC EVEN DEVICE ADDRESS?(0-076)

BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)

For the Device Address, any number from 0 to 076 can be specified. The default value is 35. Any even number from 0 to 076 can be specified as the BIC/BTC EVEN DEVICE ADDRESS. Any even number from 0100 to 0276 can be specified as the BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS.

4.3.14.3 Example

⑮ In the sample listing, it is specified that the line printer unit program is to be included for one line printer. This line printer has a device address of 35 (by default), a BIC/BTC EVEN DEVICE ADDRESS of 26, and a BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS OF 106.

4.3.15 CONFIGURATION QUESTION NUMBER 14

14.HOW MANY DCM'S-UP9?(0-4)

DCM 1:

LINE ADAPTER TYPE?(H,0-3)

DEVICE ADDRESS?(0-076; 70)

DCM INTERRUPT ADDRESS ORIGIN?(0100-0260;0260)

BITS PER BYTE?(5-010;010)

LCB MEMORY PAGE?(070-077;075)

WHICH LINES(MAX OF 4)?(00-077)

4.3.15.1 Purpose

The purpose of this question is to determine the number of DCMs for which unit programs are to be included in the configured System Test. The DCM unit program tests the DCM and its associated Line Adapters (LADs). The test currently supports the following LAD types:

F3001-00 Asynchronous LAD RS232 Modem

F3001-01 Asynchronous LAD RS232

F3001-02 Asynchronous LAD, Current Loop

F3001-03 Asynchronous LAD, Relay I/O

F3001-04 Synchronous LAD RS232 Modem

F3001-05 Binary Synchronous Communications LAD

F3006-00 BSC LAD with Wide Band option

The maximum number of DCMs that can be tested at a time is 4. For each DCM, questions are asked about the following:

- a. LAD type;
- b. Device address;
- c. DCM Interrupt Address Origin;
- d. Bits per byte;
- e. LCB Memory Page;
- f. Line identification.

4.3.15.2 Acceptable Replies

For the question about the number of DCMs, specify any number from 0 to 4.

For each DCM, the following questions are asked:

LINE ADAPTER TYPE?(H,0-3)
DEVICE ADDRESS?(0-076; 70)
DCM INTERRUPT ADDRESS ORIGIN?(0100-0260; 260)
BITS PER BYTE?(5-010;010)
LCB MEMORYY PAGE?(070-077;075)
WHICH LINES(MAX OF 4)?(00-077)

As a response to the first question, type: "H" to obtain a listing of the supported LAD types. The following list is output:

0=ASYNCHRONOUS DIRECT CONNECT
1=,, DATA SET
2=SYNCHRONOUS
3=BI-SYNCHRONOUS

Then, depending upon the type of LAD, the appropriate number should be entered.

For the device address, any number from 0 to 076 can be specified. The default value is 70. Any number from 0100 to 0260 can be specified as the DCM Interrupt Address Origin. The default value is 260. The number of bits per byte can be any number from 5 to 010. The default number is 010. The LCB (Line Control Block) memory page can be any number from 070 to 077. The default value is 075. The unit program runs 4 consecutive lines on one LAD. The line number to be tested is specified in response to the question "WHICH LINES?". A maximum of 4 such numbers can be specified. The 4 lines must be of the same type.

4.3.15.3 Example

① In the sample listing, one DCM unit program is included in the configured System Test. The LAD is specified to be of the asynchronous direct connect type. The device address is 66. Default values are used for the DCM Interrupt Address Origin, the number of bits per byte, and the LCB memory page. Two line numbers are specified: 0 and 1.

4.3.16 CONFIGURATION QUESTION NUMBER 15

15.HOW MANY TELETYPES-UP10?(0-7)

TTY n:

DEVICE ADDRESS?(0-076; 01)

4.3.16.1 Purpose

The purpose of this question is to determine the number of teletypewriters that are going to be tested. An equal number of Teletypewriter unit programs will be included in the configured System Test.

NOTE

This unit program has not yet been implemented.

4.3.16.2 Acceptable Replies

Since this unit program has not yet been implemented, specify "0" as the number of teletypewriters. If any other number is specified, the message:

```
END OF LIBRARY
UNIT PROGRAM 10 NOT FOUND
```

will be output when the system generation phase ends.

4.3.16.3 Example

① In the sample listing, 1 teletypewriter with a device address of 1 (the default) is specified. Therefore the message:

```
END OF LIBRARY
UNIT PROGRAM 10 NOT FOUND
```

is output when the system generation phase ends.

4.3.17 CONFIGURATION QUESTION NUMBER 16

16.HOW MANY CRT'S-UP11?(0-7)

CRT n:

DEVICE ADDRESS?(0-076; 01)

READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)

WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)

4.3.17.1 Purpose

This question determines the number of CRTs that are going to be tested. An equal number of CRT unit programs will be included in the configured System Test. For each CRT, questions will be asked about the following:

- a. Device Address;
- b. READ READY COMPLETE Even Interrupt Address; and
- c. WRITE READY COMPLETE Even Interrupt Address.

NOTE

This unit program is currently not implemented.

4.3.17.2 Acceptable Replies

Since this unit program is currently not available, specify "0" as the number of CRTs to be tested. If any other number is specified, the following message:

```
END OF LIBRARY  
UNIT PROGRAM 11 NOT FOUND
```

will be output when the system test generation phase ends.

4.3.17.3 Example

⑱ In the sample listing, 2 CRTs, one with a device address of 1 and the other with a device address of 2, are specified. The READ READY COMPLETE even interrupt address is 140. The WRITE READY COMPLETE even interrupt address is 142. The message: "UNIT PROGRAM 11 NOT FOUND" is output when the system test generation is completed.

4.3.18 CONFIGURATION QUESTION NUMBER 17

17.CARD READER-UP12?(Y;N)
DEVICE ADDRESS?(0-076; 30)
BIC/BTC EVEN DEVICE ADDRESS?(0-076)
BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)

4.3.18.1 Purpose

This question determines whether the Card Reader unit program is to be included in the configured System Test.

NOTE

The Card Reader unit program is currently not implemented.

4.3.18.2 Acceptable Replies

Since the Card Reader unit program is not yet available, answer "N" to the first question. If "Y" is specified, questions about the device address, BIC/BTC even device address, and the BIC/BTC COMPLETE even interrupt address are asked, and at the end of the system test generation phase, the following message is output;

UNIT PROGRAM 12 NOT FOUND

4.3.18.3 Example

① In the sample listing, it is specified that the card reader unit program be included in the configured System Test. The device address is 30 (the default), the BIC/BTC even device address is 22, and the BIC/BTC COMPLETE even interrupt address is 102. After the system test generation phase is completed, the message:

UNIT PROGRAM 12 NOT FOUND

is output.

4.3.19 CONFIGURATION QUESTION NUMBER 18

18.CARD PUNCH-UP13?(Y;N)
DEVICE ADDRESS?(0-076; 31)
BIC/BTC EVEN DEVICE ADDRESS?(0-076)
BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)

4.3.19.1 Purpose

This question determines whether the Card Punch unit program is to be included in the configured System Test. If the operator answers in the affirmative, questions are asked about the following:

- a. Device Address;
- b. BIC/BTC Even Device Address;
- c. BIC/BTC COMPLETE Even Interrupt Address.

If the operator answers "N", the next configuration question is output.

NOTE

The Card Punch unit program is currently not implemented.

4.3.19.2 Acceptable Replies

Since this unit program is not yet available, answer: "N". If "Y" is specified, questions are asked about the device address, etc., before the next configuration question is output. Also, after all the configuration questions have been output, the message:

"UNIT PROGRAM 13 NOT FOUND"

is output.

4.3.19.3 Example

② In the sample listing, it is specified that the Card Punch unit program be included as part of the configured System Test. The device address of the card punch is 31 (the default),

the BIC/BTC even device address is 22, and the BIC/BTC COMPLETE even interrupt address is 102.

After all the configuration questions are asked and satisfactorily answered, the message:

"UNIT PROGRAM 13 NOT FOUND"

is output.

4.3.20 CONFIGURATION QUESTION NUMBER 19

19.PAPER TAPE-UP14?(Y;N)
DEVICE ADDRESS?(0-076; 37)
CHAR READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)

4.3.20.1 Purpose

This question determines whether the Paper Tape unit program (number 14) is to be included in the configured System Test. If the answer is Yes, questions about the device address and the CHAR READY COMPLETE even interrupt address are asked before the final step of the configuration processing begins. If, the answer is No, these questions are not output.

NOTE

The Paper Tape unit program is not currently implemented.

4.3.20.2 Acceptable Replies

Since this unit program is not currently implemented, answer "N".

If "Y" is specified, questions are asked about the device address etc., before the next phase of the configuration process begins. Also, the message "UNIT PROGRAM 14 NOT FOUND" is output.

4.3.20.3 Example

(21) In the sample listing, it is specified that the Paper Tape unit program be included as part of the configured System Test. The device address is 37 (the default), and the CHAR READY COMPLETE even interrupt address is 144.

After all the configuration questions are asked and satisfactorily answered, the message: "UNIT PROGRAM 14 NOT FOUND" is output.

NOTE

This question is the last configuration question. It marks the end of the configuration dialog.

4.3.21 PREPARATION OF THE CONFIGURED SYSTEM TEST

After all the configuration questions have been answered, the selected unit programs are input from the standard System Test tape, and the following actions take place:

1. Parameters specified during the configuration dialog are processed to prepare the configured System Test.
2. Selected unit programs are then read into the second 32K words of memory in map key 1 by EXEC.
3. If an unit program selected by the operator is not found on the standard System Test, the following message is output:

```
END OF LIBRARY  
UNIT PROGRAM XX NOT FOUND
```

(See sample listing (22).)

4. If the first 32K memory module gets filled before all selected unit programs are input, the following message is output:

```
END OF 32K MEMORY  
UNIT PROGRAM XX NOT FOUND
```

Unit programs which are not found will not be loaded. All other unit programs are loaded.

5. An input error is indicated by the message:

```
INPUT ERROR
```

If the input device can be set up to read the first unit program, an attempt to recover can be made by entering a space character; otherwise, no recovery is possible (a rerun is required).

6. Successful completion of the configuration is indicated by the following message:

```
SYSTEM CONFIGURATION DONE  
PREPARE OUTPUT DEVICE-ENTER SPACE WHEN READY
```

(See sample listing (23).)

When the "PREPARE OUTPUT DEVICE-ENTER SPACE WHEN READY" message appears, enter a space character. This causes the configured System Test to be output.

If there is an output error, the message:

OUTPUT ERROR

is generated, followed by the message previously displayed, namely:

PREPARE OUTPUT DEVICE-ENTER SPACE WHEN READY

Attempt a recovery by entering a space character, or enter a Control-C to restart the dialog.

Successful output of the configured System Test is indicated by the message:

SYSTEM GENERATION DONE

NOTE

To make another copy of the configured System Test, prepare the output device again, and enter a space. Procedures for loading and executing the configured System Test are given in Section 6.

4.4 ERROR DETECTION DURING SYSTEM TEST GENERATION

System errors may be detected by the Preliminary Loader, the System Test Generator, the Executive Program (EXEC), or the unit programs. Errors detected by the unit programs are discussed in later sections of the manual.

4.4.1 PRELIMINARY LOADER

During execution of the Preliminary Loader, a Preliminary Instructions Test, and a Preliminary Memory Test are executed prior to loading the rest of the System Test. (Refer to Sections 3 and 5 of this manual).

Any errors detected by these preliminary tests, forces the CPU into HALT mode with an error code of 01 to 052 in the instruction register. These are identical to the codes used by the MAINTAIN III test programs. (Refer to Table 5-2 for a list of the error codes.) If such errors occur, nothing further can be done at the system level. The MAINTAIN III diagnostics must be run at this stage.

If the preliminary tests are successfully executed, the rest of the System Test (standard or configured, depending upon the phase of the operation) is loaded from the input device. An error detected at this stage results in the following message:

LOAD ERROR

and the CPU goes into HALT mode. No recovery is possible. The System Test must be reloaded.

4.4.2 ERRORS DETECTED BY THE SYSTEM TEST GENERATOR

Errors detected by the System Test Generator can be grouped into the following categories:

- Map Errors.
- Configuration Errors.
- Input Errors.
- Output Errors.

4.4.2.1 Map Errors

Map errors are not recoverable. Map error messages have the following format:

MAP TIME OUT ERROR

or

MAP WRITE/READ ERROR,MAP REG xx,(y)(z)

where:

xx

Is the map register number. Can be any number from 00 to 077.

y Is the data written.

z Is the data read.

4.4.2.2 Configuration Errors

An invalid (unacceptable) response to a configuration question generates the following error message:

??

Enter the correct reply, or enter a Control-C to return to the beginning of the configuration dialog.

If the response to a question is incorrectly entered:

1. Type a back-arrow (←) to delete the previous character, or
2. Type a back-slash (\) to delete the entire reply.

Reenter the response correctly.

4.4.2.3 Input and Output Errors

Input and output errors have already been described in this section under "Preparation of the Configured System Test".

Section 5

Configured System Test: Operational Theory

5.1 GENERAL

This section describes how the various components of the configured System Test function. User interactions with the test and operational procedures are discussed in later sections.

5.2 PURPOSE

The configured System Test is generated from the standard System Test. It is a tailored version of the standard System Test configured for a given system configuration.

5.3 COMPONENTS

The configured System Test has the following components:

1. Preliminary Test
 - Instructions Test
 - Memory Test
2. Preliminary Loader
3. Executive Program
4. Unit programs

The first three components are identical to the corresponding components in the standard System Test. The System Test Generator is not one of the components of the configured System Test. The fourth component, namely, the unit programs, is the set of programs selected from the Library of unit programs which is a part of the standard System Test. The selection is specified during the configuration dialog, and is made during configured System Test generation.

5.3.1 PRELIMINARY INSTRUCTIONS TEST

The Preliminary Instructions Test portion of the System Test is identical to the Preliminary Instructions Test portion of the executive test program of MAINTAIN III. (Refer to Section 2, MAINTAIN III TEST PROGRAMS User Manual, UP-8672).

The test validates basic CPU operations.

The Preliminary Instructions Test validates central processing unit (CPU) operations by testing the machine instructions listed in Table 5-1. Successful execution of this test indicates that the System Test can be correctly loaded.

After the bootstrap routine loads the Preliminary Loader and the Preliminary Tests, it jumps to address 007000. Then, the Preliminary Instructions Test is automatically executed, starting at address 007002. Following successful execution of the instructions test, the Preliminary Memory Test is automatically executed. Following successful execution of this test, the Preliminary Loader automatically loads the rest of the System Test.

Table 5-1. Preliminary Instructions Test Summary

| Mnemonic | Description |
|----------|---|
| ADD | Add memory to A register |
| ADDI | Add immediate |
| ANAI | AND immediate |
| DAR | Decrement A register |
| DBR | Decrement B register |
| DECR 02 | Set B register to -1 |
| DXR | Decrement X register |
| ERA | Exclusive-OR memory and A register |
| ERAI | Exclusive-OR immediate |
| IAR | Increment A register |
| IBR | Increment B register |
| INCR 03 | Set A and B registers to +1 |
| IXR | Increment X register |
| JAN | Jump if A register negative |
| JAP | Jump if A register positive |
| JAZ | Jump if A register zero |
| JBZ | Jump if B register zero |
| JIF 011 | Jump if A register = 0 and OVFL is set |
| JMP | Jump (unconditional) |
| JMPM | Jump and mark (unconditional) |
| JMP* | Jump indirect |
| JOF | Jump if overflow indicator set |
| JXZ | Jump if X register zero |
| LDA | Load A register |
| LDAI | Load A register immediate |
| LDB | Load B register |
| LDBI | Load B register immediate |
| LDX | Load X register |
| LDXI | Load X register immediate |
| LLRL | Load logical rotation left |
| LLSR | Long logical rotation right |
| LRLA | Logical rotation left A register |
| LSRA | Logical shift right A register |
| MERG 032 | Transfer ORed A and B registers to B register |
| NOP | No operation |
| ORAI | Inclusive-OR immediate |
| ROF | Reset overflow indicator |
| STA | Store A register |
| STAI | Store A register immediate |
| STB | Store B register |
| STX | Store X register |
| SUB | Subtract memory from A register |
| TBA | Transfer B register to A register |
| TBX | Transfer B register to X register |
| TXA | Transfer X register to A register |
| TZA | Transfer zero to A register |
| TZB | Transfer zero to B register |
| TZX | Transfer zero to X register |
| XAZ | Execute if A register zero |
| XBZ | Execute if B register zero |
| XIF 022 | Execute if B register = 0 and A register = ≥ 0 |

If an error is detected by the Preliminary Instructions Test, the program halts with the error code in the instruction register. Preliminary instructions test error codes are listed in Table 5-2.

To continue program execution after an error halt, press START or RUN. To loop on the subtest in error:

- a. Set Sense Switch 2.
- b. Refer to the program listing for the jump address specified by the preceding JSS2 instruction, and set the P register to that address.
- c. Press START or RUN.

Refer to the program listing for the significance of the A, B, and X registers after an error halt, and to the applicable maintenance manual for correction procedures.

5.3.2 PRELIMINARY MEMORY TEST

This test is automatically executed after successful execution of the Preliminary Instructions Test. The test verifies correct operation of the first 32K of memory (with the exception of the area in memory where the System Test resides). Memory addresses 000044 through 017777 are tested in two passes. The first pass checks each address with a pattern of 0125252. The original contents of memory are saved and restored by the program.

If an error is detected in the memory test, the program halts with 000077 in the instruction register, the address of the faulty cell in the X register, and the bits in error in the A register. To continue the test, press START or RUN. To loop on an error:

- a. Set Sense Switch 2.
- b. Press START or RUN.

Following successful execution of the Preliminary Memory Test, the Preliminary Loader loads the rest of the configured System Test.

Table 5-2. Preliminary Instruction Test Error Codes

| Error Code | Instruction Subtest |
|------------|---------------------|
| 000001 | TZA/DAR/JAZ/JAN |
| 000002 | LDA/IAR/STA |
| 000003 | LDB/JBZ/TZB |
| 000004 | IBR/DBR |
| 000005 | LDX/JXZ/TZX |
| 000006 | IXR/DXR |
| 000007 | LDAI/JAN |
| 000010 | LDAI/ERA/JAN |
| 000011 | ERA I/JAP |
| 000012 | LDBI/TBA |
| 000013 | LDXI/TXA |
| 000014 | LDB/TBX |
| 000015 | LDA/ERA |
| 000016 | LDA/STA |
| 000017 | LDB/STB |
| 000020 | LDX/STX |
| 000021 | XAZ |
| 000022 | XBZ |
| 000023 | ROF/SOF/JOE |
| 000024 | ROF/JOE/JMP |
| 000025 | JMPM/(JMP) |
| 000027 | LRLA |
| 000030 | LLSR |
| 000031 | LLSR |
| 000032 | LLRL |
| 000033 | LLRL |
| 000034 | ADD |
| 000035 | ADDI/ORAI |
| 000036 | SUB |
| 000037 | NOE |
| 000040 | INCR 03 (005103) |
| 000041 | DECR 02 (005302) |
| 000042 | MERG 032 (005032) |
| 000043 | LSRA |
| 000044 | LDA |
| 000045 | STA |
| 000046 | ANAI |
| 000047 | STAI |
| 000050 | XIF 022 (003022) |
| 000051 | JIF 011 (001011) |

5.3.3 PRELIMINARY LOADER

The Preliminary Loader is a binary loader which loads formatted object data into computer memory, computes the check-sum, and transfers program control as directed. The Loader computes the check-sum of each record of the System Test (object) as it is loaded, and compares the result with the expected value in the check-sum frames at the end of each record.

If a check-sum error is detected during the program loading, the program stops, and the following message is output:

CHECKSUM ERROR X = xxxxxx

where:

xxxxxx

Is the error address in the X register.

After a check-sum error halt:

1. Run the MAINTAIN III Test Programs to determine whether the tape drive is malfunctioning.
2. If the tape drive appears to be functioning properly, redo the System Test generation.

5.3.4 EXECUTIVE PROGRAM

The Executive Program (EXEC) controls the execution of the System Test. It performs the scheduling and mapping functions required to execute the unit programs in a real-time, multiprogrammed, mapped, interrupt-driven environment. The Executive also performs an error-logging function. A count of errors detected by each unit program is maintained. The EXEC provides an interface which allows the operator to perform various operational and debugging functions. The functions are performed using commands which are executed by the EXEC. All interrupt and I/O functions are performed by the EXEC in response to requests (service calls) made by the unit programs.

5.3.5 UNIT PROGRAMS

The unit programs in the configured System Test are those selected from the Library of Unit Programs in the standard System Test. The selection is specified by the operator during the configuration dialog. It is effected by the System Test Generator (refer to Section 2) during the system test generation process.

All the unit programs run under the control of the EXEC and conform to certain interface requirements. All the unit programs have a standard identification format.

5.3.5.1 Identification

Each unit program has a standard identification (ID).

This ID has the following format:

M4xxnn.v

where:

- M4 Is the prefix common to all the unit programs.
- xx Is a two-character literal identification.
- nn Is a two-digit decimal number identification.
- v Is the version level (0 or A - Z).

The IDs of all the currently available unit programs are listed in Table 5-3.

Table 5-3. Unit Program IDs

| Unit Program Name | Unit Program ID | UP-No. |
|---|-----------------|--------|
| V77-600 Writable Control Store | M4WC01 | UP1 |
| V77-600 Floating-Point Processor | M4FP02 | UP2 |
| Memory | M4ME03 | UP3 |
| Cartridge Disk | M4DF04 | UP4 |
| Disk Memory and Controller | M4DC05 | UP5 |
| Direct Memory Access Disk | M4DH06 | UP6 |
| Magnetic Tape | M4MT07 | UP7 |
| Line Printer | M4LP08 | UP8 |
| Data Communications Multiplexer | M4DM09 | UP9 |
| V77-800 Writable Control Store and Floating-Point Processor | M4WC15 | UP15 |

Examples:

In the following unit program ID:

M4WC01.0

M4 is the unit program ID prefix. WC is the two-character literal identification. The letters "WC" indicate that the unit program being discussed is the WCS unit program. 01 is the two-digit decimal number identification. The version level is 0.

Consider the following unit program ID:

M4MT07.A

M4 is the unit program ID prefix. The letters "MT" indicate that the unit program is the Magnetic Tape unit program. 07 is the decimal number identification. The version level is A.

5.3.5.2 Assembly

All the unit programs are assembled relative to 0, and start at location 01000 with a jump instruction to the program execution starting address. The contents of locations 01000 to 01007 are listed in Table 5-4.

Table 5-4. Contents of location 01000 to 01007

| Location address | Contents |
|------------------|--|
| 01000,1 | Jump instruction to execution starting address |
| 01002 | M4 |
| 01003 | Two-character literal identification |
| 01004 | Two-digit decimal number |
| 01005 | .v where v is the version level |
| 01006,7 | Alternate entry point |

Locations 01010 to 01027 are used by the EXEC to pass parameters to the unit programs. Refer to Table 5-5.

Table 5-5. Contents of Locations 01010-01027

| Location | Contents |
|----------|---|
| 01010 | Map key assigned to unit program |
| 01011 | Model number, unused if not applicable |
| 01012 | Device address to be used by the unit program |
| 01013 | BIC/BTC address, if any |
| 01014-27 | Depends upon unit program |

5.3.5.3 Executive Service Calls

All interrupt and I/O functions required by the unit programs are performed by the EXEC. The unit programs request these services using Executive Service calls. A call is made by executing a jump to the service routine address. Data registers are used to pass information between the EXEC and the unit programs. The call name, location (address) of the service routine being called, and the function of the call are described in Table 5-6. Data passed between the EXEC and the unit programs during execution of the service calls in the A, B, X, and R3 registers is described in Table 5-7. Transfer of data between the unit program and the device controller is accomplished by use of a BIC (Buffer Interlace Controller) or direct memory access depending upon the type of controller. Interrupts are processed by use of a PIM (Priority Interrupt Module).

Table 5-6. Executive Service Calls

| Call Name | Location | Function |
|-----------|----------|---|
| M\$SMEM | 0400 | Allocates physical memory. |
| M\$RSUM | 0401 | Resume EXEC interrupt handler. |
| M\$SBIC | 0402 | Allocates BIC/BTC or I/O device. |
| M\$RBIC | 0403 | Releases a previously selected BIC/BTC or I/O device. |
| M\$PREQ | 0405 | Prints a message on the operator's console, or system list device. |
| M\$CCBA | 0407 | Sets Channel Control Block (CCB) address in Unit Address Word Table. |
| M\$EXIO | 0410 | Executes one of the following I/O instructions: EXC EXC2 SEN CIA CIB OAR OBR |
| M\$FINI | 0411 | Indicates that the unit program has completed execution and can be rescheduled. |
| M\$TRAP | 0417 | Special debugging call generated by the operator command: "TRAP". (Refer to Section 6). |

Table 5-7. Data Passed By Executive Service Calls

| Call Name | Data Required by Executive | | Data Returned by Executive | |
|-----------|----------------------------|--|----------------------------|---|
| | Register | Data | Register | Data |
| MSSMEM | A | Starting logical address of required memory. | A | Starting physical address, low 16 bits. |
| | B | Ending logical address of required memory. | B | Ending physical address, low 16 bits. |
| | | | X | Physical address, high 4 bits. |
| MSSBIC | A | BIC/BTC even address or device address. | | None |
| | B | Time delay. | | |
| M\$RBIC | A | BIC/BTC even address or device address. | | None |
| M\$PREQ | A | <p>Logical address of message table in unit program's map. A positive address (bit 15=0) indicates a non-error message. A negative address (bit 15=1) indicates an error message.</p> <p>Message Table Format:</p> <p>+ve address=starting address of an ASCII string. -ve address=address of an octal word +1 Generates a <cr-lf> pair. -1 Suppresses the <cr-lf> pair. 0 Terminates the message and generates a <cr-lf>.</p> | | None |

Table 5-7. Data Passed By Executive Service Calls (Continued)

| Call Name | Data Required by Executive | | Data Returned by Executive | |
|-----------|----------------------------|---|----------------------------|--|
| | Register | Data | Register | Data |
| M\$CCBA | A | Logical address of CCB in unit-program's map. | A | Unit Address Word Table address in map 0. |
| | B | Drive unit number | B | CCB physical address, low 16 bits. |
| | | | X | CCB physical address, high 4 bits. |
| M\$EXIO | A | OAR data, if any | A | 0 if the instruction is an SEN which jumped. |
| | B | OBR data, if any; or time delay | | 1 if the instruction is an SEN which did not jump. |
| | X | I/O instruction | | <data> if the instruction is a CIA. |
| | R3 | Interrupt expected code: 0=no interrupt expected +1=BIC/BTC complete interrupt to be processed. -1=device interrupt to be processed. +2=return with interrupts off. | B | 0 if lapsed time delay. Non-zero (remaining time) if time delay did not lapse. <data> if the instruction is a CIB. |
| M\$FINI | | None. | | None. |
| M\$RSUM | A | 0 to post interrupt received in TIDB status word. 1 to inhibit posting. | | None. |
| | X | TIDB address passed by EXEC to caller. | | |

5.3.5.4 Operational Theory

Each unit program handles one controller and all units attached to that controller. Unit program operation for I/O controllers consists of the following steps:

1. Records are written to each unit on the controller, alternating between units.
2. The records are then read back into memory from the units.
3. The data read is compared with the data written to determine whether a valid read/write operation has taken place.

For this operation, all the memory from the end of the unit program through the end of its 32K memory map is used as buffer space. Variable size records are moved through this buffer space and written out.

For Magnetic Tape controllers, the tape is backspaced between the read and write operations. If the end of the tape is reached before the program ends, the tape is rewound, and the writing is continued.

For non-I/O devices and devices which only output data, action appropriate to the particular device is performed. Error messages are output when either erroneous results are obtained or equipment malfunctions occur.

Once a program is scheduled for operation by the EXEC, it continues in operation until one of the following occurs:

- a. For a non I/O device, function of the program is completed.
- b. For other devices, all available buffer space has been used to output data on each available unit on the assigned controller.

When an error occurs, an error message is output, and that unit is flagged as down.

NOTE

The only external input provided to a unit program is that provided by the EXEC at system generation time. The only output from a unit program consists of records written to the device and any error messages that may be generated.

Once execution of the System Test is started, it operates continually. The operation is automatic, and requires no operator intervention. Execution of the System Test can be stopped either by shutting down the system, or by entering a Control-C.

If at any time during the execution of the System Test the computer is forced into STEP mode, restart at location 0.

There is no operator interface for a unit program. Data required by the unit programs and responses to any error message that may be generated are both handled exclusively by the EXEC.

Section 6 System Test Operation

6.1 OVERVIEW

This section describes the following phases of System Test operation:

- Initial preparation
- Loading the configured System Test
- Operator interaction with the test.

6.2 PREPARATION

Place all units to be tested in the READY or online state, as follows:

1. Line printer: Load with paper, and place online.
2. Teletypewriter or CRT terminal: Place online.
3. Magnetic Tape Units: Mount the configured System Test tape on the system boot device. Mount scratch tapes which are write-enabled, on the other magnetic tape units. Position the tapes to load point. Press ON-LINE.

CAUTION

The configured System Test tape
should not be write-enabled.

4. Disk units: Mount disk packs on units. Use formatted, scratch disk packs. Place unit in READY mode. Refer to the VORTEX II System Generation User Guide/Programmer Reference, UP-9083, for information about disk formatting.

6.3 LOADING THE CONFIGURED SYSTEM TEST

Load the configured System Test tape using the procedures described in Section 3. Preparation and procedures for loading the Preliminary Loader are identical to those described in Section 3. After the Preliminary Loader is loaded, and it loads the rest of the configured System Test, the following message is output to the operator's console:

```
V70-SERIES SYSTEM TEST
*****
FOR COMMAND LIST,TYPE HE.
EXEC PROMPTER IS EX**
ENTER SPACE IF OUTPUT PAUSES
MEMORY SIZE= 0XXXX
*****
EX**
```

The operator can now enter any one of the operator commands described in the following paragraphs.

6.4 OPERATOR COMMANDS

Once the configured System Test (Test) is loaded, the operator can direct the Test using operator commands. The operator commands are simple and short. Operator commands are also referred to as Executive commands, since they are directives to the Executive Program (EXEC). They allow the operator to perform a variety of Test control functions, including the following:

- Select unit programs to be scheduled for execution.
- Debug unit programs.
- Alter the map key and pages assigned to a unit program scheduled for execution.
- Examine and alter the contents of a specific location in a specific map key.
- Examine the Error Log Table for the preceding run.

The operator commands can be grouped into the following categories:

- Tutorial/Reference Commands
 - HELP
 - INDEX
 - LOG

- Executive Commands
 - RUN
 - ENABLE
 - DISABLE

- Manipulation Commands
 - CHANGE
 - CHANGE MAP
 - EDIT
 - INSERT
 - KEY
 - REG
 - SEARCH
 - TRAP
 - VIEW

In the following paragraphs, these commands are discussed in the order presented above. In the command summary (Section 6.5), the commands are listed in alphabetic sequence.

If the computer is forced into STEP mode at any time during the execution of the System Test, restart at location 0.

6.4.1 USING THE OPERATOR COMMANDS

The Test informs the operator that it is ready for execution, by sending the following message to the operator's console:

```
V70-SERIES SYSTEM TEST
*****
FOR COMMAND LIST,TYPE HE.
EXEC PROMPTER IS EX**
ENTER SPACE IS OUTPUT PAUSES
MEMORY SIZE= 0XXXK
*****
```

The Test outputs the prompt character: "EX**". Operator commands can be entered only after this prompt appears.

The operator commands have some features (requirements, parameters, and options) that are common to many of them. These common features are:

- If a command is erroneously entered, type a back-slash (\) character before terminating the command. This erases the line, and the command can now be reentered correctly.

Example:

```
EX**  
C\  
V60\  
V67020,67037,0.
```

The first back-slash erases the "C" command. The second back-slash erases the "V60" command.

- Multiple fields must be separated by commas.

Example:

```
RU1,2,3.
```

Numbers 1, 2, and 3 must be separated by commas.

- If one optional field is omitted, but a following optional field is included, a comma must be entered in the position of the omitted field.
- Each command must be terminated with a <cr> or ".". In most cases, the two can be used interchangeably. Cases where they cannot be used interchangeably are flagged as such in the manual, and the differences in usage are explained.

Example:

The HELP command can be entered either as:

```
HE.  
or  
HE<cr>
```

NOTE

The <cr> is shown here and elsewhere in this manual for explanatory purposes only. It does not echo on the terminal.

- All numbers required by a command must be entered in octal notation, with the exception of unit program numbers. The unit program number is the two-digit decimal number which is part of the unit program ID. (See Section 5).

Example:

```

EX**
RU/FDB. ①
M4LP08.0 DA=035 KY=015 PP=0153,0154 BP=0,01
*****
EX**
K.
M4LP08.0 DA=035 KY=015 PP=0153,0154 BP=0,01
EX**
C1575.(030300)

```

- ① The unit program number entered as a part of the RUN command is a decimal number.
- ② "1575" is an octal number. Note, however, that the leading zero may be omitted.
- If the command results in program output, the output is automatically sent to the operator's console (teletypewriter or keyboard-display terminal), unless the /L option is specified.
 - To select the line printer as the list device, use the /L option if the command has that option.

Example:

If the HELP command is entered as:

```
HE/L.
```

the resulting list is output on the line printer.

- If the selected list device is not available or not ready, the output is sent to the other (alternate) list device.
- When the output is being listed on the operator's console, there is a pause after every 20 lines are output. The operator can then either continue or terminate the output by entering the following:

- Type a space character to continue.
- Type any character other than a space character to terminate.

This feature gives the operator enough time to view the data when it is being displayed on a keyboard-display terminal.

The Test prompt character is EX**. The EX stands for "EXEC". EXEC is the Executive Program which is the chief controlling component of the Test.

The EX** prompt can be invoked any time during the operation of the Test by entering the Control-C character. This reinitializes the system, but saves the map keys of the unit programs. Since the map keys are saved, the user can subsequently access the unit programs in their respective maps.

NOTE

The EX** prompt can also be invoked with a control panel or microprocessor interrupt. The map keys of the unit programs are not saved.

NOTE

To run the Test in Debug mode:

1. Wait for the EX** prompt to appear.
2. Enter the RUN command specifying the /D option. (Refer to Section 6.4.2 for the purpose, format, and usage of the RUN command).

A "/D" following the RUN command causes the requested programs to be stored in their respective maps, followed by a return to EXEC (EX**). This allows the operator to examine and/or alter locations within these unit programs. The Test will continue in the Debug mode of operation until the EX** prompt is invoked by entering <Control-C>.

6.4.2 HELP COMMAND

6.4.2.1 Purpose

The HELP command assists the user by providing him with tutorial information about all the operator commands. The command causes all the operator commands, along with a brief description of each, to be listed on the list device.

6.4.2.2 Format

The HELP command has the following format:

```
HE[/L] { <cr> }  
          { . }
```

where:

/L

Selects the line printer as the list device.

6.4.2.3 Usage

The command can be terminated either with a <cr> or with a ".".

To use the operator's console (teletypewriter or keyboard-display terminal) as the list device, enter the following:

```
HE.
```

or

```
HE<cr>
```

To send the listing to the line printer, enter the following:

```
HE/L.
```

or

```
HE/L<cr>
```

NOTE

The format for the command uses an abbreviated form (HE) of the command name (HELP).

6.4.2.4 Example

In the following example, the operator types "HE" after the EX** prompt, and obtains a listing of all the operator commands. After the list is completed the EX** prompt appears again. The operator can now enter a new command.

```
EX**
HE.
EXECUTIVE COMMANDS
NOTE:ALL COMMANDS END WITH CR OR .
  /L=OUTPUT TO LPR
  *K=MAP KEY
HELP:PRINT COMMAND LIST
  HE/LJ
INDEX:PRINT UNIT PROGRAM LIST
  IN/LJ
RUN:RUN UNIT PROGRAMS
  RU/LJ[FJ/DJCN1,N2,...J
  /F=FIXED MODE
  /D=DEBUG MODE
ENABLE:ENTER UNIT PGMS IN RUN LIST
  EN1,N2,...J
DISABLE:DELETE UNIT PGMS FROM RUN LIST
  D1,N2,...J
EDIT:DISPLAY/ALTER KEYS,PAGES
  ED(CR OR .)EKJ,PJ
CHANGE MAP:ALTER NEXT MAP KEY,PAGE
  CKJ,PJ
LOG:PRINT ERROR LOG TABLE
  LO/LJ
REGISTER:DISPLAY/ALTER REGISTERS
  (A)B)X)R)N)I,KJ(CR OR .)D)T)
NOTE:D=NEW DATA,IF ANY
  T=COMMA DISPLAYS/ALTERS NEXT REGISTER
  *OTHER TERMINATES COMMAND
KEY:DISPLAY KEYS/SET DEBUG MAP KEY
  K/LJCNJ
CHANGE:DISPLAY/ALTER MEMORY IN K
  C(X)I,KJ(CR OR .)D)T)
NOTE:D=NEW DATA,IF ANY
  T: ,=NEXT; .=PREV; *=INDIRECT; CR=END
VIEW:FROM X TO Y,IN K,ON LPR OR TTY
  VE/LJ(X)I,YJ,KJ
INSERT:INSERT P INTO X TO Y,IN K
  I(X),(Y)E,PJ,KJ
SEARCH:FROM X TO Y,IN K,FOR P MASKED BY M
  SE/LJ(X),(Y)E,PJ,MJ,KJ
TRAP:TRAP AT X,GO TO Y,IN K
  TXJ,YJ,KJ ERROR LOG TABLE
EX**
```

Each one of these commands is discussed in detail in following subsections.

6.4.3 INDEX COMMAND

6.4.3.1 Purpose

The INDEX command lists the IDs of all the unit programs residing in memory in map key 0. The command assists the operator by identifying all the unit programs that he can run. The unit programs are identified by ID, device address (DA), and TIDB (Task Identification Block). The TIDB contains all the information about a task during its execution.

6.4.3.2 Format

The command has the following format:

```
IN[/L] {<cr>
        .}
```

where:

/L

Selects the line printer as the list device.

6.4.3.3 Usage

The command can be terminated either with a <cr> or with a ".".

To use the operator's console as the list device, enter the following:

```
IN.
```

or

```
IN<cr>
```

To use the line printer as the list device, enter the following:

```
IN/L.
```

or

```
IN/L<cr>
```

NOTE

The format for the command uses an abbreviated form (IN) of the command name (INDEX).

6.4.3.4 Example

The example shown here is an excerpt from a listing obtained during System Test operation on a V77-600 system.

```
EX**  
IN  
EXEC00.C DA=0 TIDB=017400  
M4WC01.0 DA=074 TIDB=017500  
M4ME03.0 DA=0 TIDB=017600  
M4DH06.A DA=014 TIDB=017700  
M4HT07.A DA=010 TIDB=020000  
M4LP08.0 DA=035 TIDB=020100  
M4DH09.0 DA=070 TIDB=020200  
EX**
```

In response to the operator command: "IN" a list is output. The first program on the list is the Executive Program EXEC. All the other programs on the list are unit programs residing in map key 0 of memory.

For each program, the program ID, Device address, and TIDB are listed.

Refer to Section 5 for the format of the program ID and a cross-reference list of unit program names and IDs.

In the example:

M4WC01.0

Is the V77-600 Writable Control Store (WCS) unit program.

M4ME03.0

Is the Memory unit program.

M4DH06.A

Is the Direct Memory Access Disk (Type DH) unit program.

M4MT07.A

Is the Magnetic Tape unit program.

M4LP08.0

Is the Line Printer unit program.

M4DM09.0

Is the DCM unit program.

6.4.4 LOG COMMAND

6.4.4.1 Purpose

The LOG command is used to list the entries made in the Error Log during the previous run of the System Test. The Error Log maintains a record of the number of errors detected by each unit program. Each entry shows the decimal number code of a unit program and the number of errors detected by it. The Error Log is automatically initialized each time a new RUN command is entered.

6.4.4.2 Format

The LOG command has the following format:

```
LO[/L] { <cr>
        . }
```

where:

/L

Selects the line printer as the list device.

6.4.4.3 Usage

The Error Log is particularly useful when the System Test is being run over a long period of time, without operator intervention, and without output of error messages to the line printer. By analyzing the Error Log, the operator can:

- a. Determine which unit programs have detected errors, and
- b. Rerun these unit programs.

He can thus focus on the problem areas quickly. The operator can also use the Error log as a reference point, to gauge the progress he is making in detecting and remedying errors.

To use the teletypewriter or keyboard-display terminal as the list device, enter:

```
LO<cr>
```

or

```
LO.
```

To use the line printer as the list device, enter the following:

LO/L<cr>
or
LO/L.

6.4.4.4 Example

```
EX**  
LO.  
ERROR LOG TABLE  
UNIT PROGRAM ERRORS  
  0      01  
  01     0  
  02     0  
  03     0  
  04     0  
  05     0  
  06     0  
  07     0  
  08     0  
  09     0  
  010    0  
  011    0  
  012    0  
  013    0  
  014    0  
  015    0
```

6.4.5 RUN COMMAND

6.4.5.1 Purpose

The RUN command identifies the unit programs (by decimal number codes) which are to be scheduled for execution, and executes them.

6.4.5.2 Format

The command has the following format:

$$\text{RU}[/\text{L}] [/\text{F}] [/\text{D}] [\text{n1}, \text{n2}, \text{n3}, \dots] \left\{ \begin{array}{l} \langle \text{cr} \rangle \\ \cdot \end{array} \right\}$$

where:

/L
Selects the line printer as the list device.

/F
Specifies the fixed mode of operation.

/D
Specifies the debug mode of operation.

n1, n2, n3, ...

Are the decimal number codes of the unit programs to be scheduled for execution. This list of unit programs is the "run list".

6.4.5.3 Usage

The command can be terminated either with a <cr> or with a "." following the run list. The list of unit programs (n1, n2, ...) specified in the RUN command, is the "run list". If no unit programs are specified, the character which terminates the command determines what action is taken as follows:

- If the command is terminated with a <cr>, all the unit programs in the configured System Test, are automatically included in the run list.
- If the command is terminated with a ".", the run list specified in the previous RUN command is used again.

To use the line printer as the list device, use the /L option.

The System Test has three modes of operation:

- Normal (or variable) mode. (This is the normal or default mode of operation).
- Fixed mode. (This is specified by the /F option).
- Debug mode. (This is specified by the /D option).

In the normal or variable mode of operation, after each unit program completes execution, it is assigned a new map key and pages when it is subsequently rescheduled. This new map key is the first free (available) map key.

In the fixed mode of operation, each unit program is initially assigned a map key and pages. Each unit program continues to use the map key and pages originally assigned to it whenever it is rescheduled.

In the debug mode of operation, the unit programs specified in the RUN command are stored in their respective maps. After the Executive Program (EXEC) issues the EX** prompt, the operator can enter any one of the operator commands, with the exception of the RUN command. Some of the operator commands, however, are particularly suitable for use in this mode of operation. These commands allow the operator to perform the following debugging operations:

- Examine and/or alter locations within a unit program
 - CHANGE command
 - VIEW command
 - INSERT command
 - SEARCH command
- Alter the map key and pages assigned to a particular unit program
 - CMAP command
 - EDIT command
 - KEY command
- Display the contents of a specified register at the time of the previous interrupt
 - REG command

- Set a breakpoint at a particular location
 - TRAP command

The operator can use the /D option either:

1. in combination with the normal (variable) mode, or
2. in combination with the fixed mode.

NOTE

If more than one slash option is chosen, the options need not be separated by slashes. Only one slash (immediately after the "RU") is required.

Example:

To run unit program 4 in both the fixed and debug modes simultaneously, enter:

RU/FD4.

Do not enter:

RU/F/D4.

NOTE

To exit from the debug mode, enter a <Control-C> character. This causes the EX** prompt to reappear. The RUN command can be entered only after a EX** prompt.

6.4.5.4 Examples

Example 1:

EX**

RU

```
M4WC01.0 DA=074 KY=01 PP=0100 BP=0,0
M4ME03.0 DA=0 KY=02 PP=0101 BP=0,0
M4DH06.A DA=014 KY=03 PP=0102,0103,0104 BP=0,0
M4MT07.A DA=010 KY=04 PP=0105,0106,0107 BP=0,0
M4LP08.0 DA=035 KY=05 PP=0110,0111 BP=0,0
M4DM09.0 DA=070 KY=06 PP=0112,0113,077 BP=0,0
*****
```

In this example, after the EX** prompt, the operator enters the RUN command. The run list specifies all the unit programs in the configured System Test. The EXEC lists descriptive information about each unit program in the run list, as it is executed. The descriptive information provided consists of the following:

- Program ID
- Device Address
- Map key
- Physical pages assigned to the unit program
- Physical pages assigned to the temporary work-area (buffer) which the unit program uses.

The following information is provided about the first unit program in the run list:

```
M4WC01.0 DA=074 KY=01 PP=0100 BP=0,0
```

where:

M4WC01.0

Is the program ID. (See the INDEX command for a description of the program ID).

DA=074

Gives the Device Address.

KY=01

Is the map key assigned to this unit program.

PP=0100

Is the physical page assigned to the unit program. Each time the program is rescheduled, the EXEC assigns pages to it.

BP=0,0

This information relates to the buffer used by the unit program, during execution. The first number (0) is the address of the first physical page assigned to the buffer. The second number (0) is the actual number of pages used by the buffer. These pages are released when the unit program completes execution. Unit programs request buffer pages using the M\$MEM call. (Refer to Section 5.)

The same format is used to provide descriptive information about all the unit programs in the run list.

NOTE

If more than one page is assigned to the unit program, each page is listed.

For example:

M4MT07.A DA=010 KY=04 PP=0105,0106,0107 BP=0,0

This unit program has been assigned physical pages 0105, 0106, and 0107.

By entering a <Control-C>, the operator can invoke the EX** prompt. After the EX** prompt appears, the operator can enter another command.

Example 2:

EX**

RU4

*UNEXPECTED INTERRUPT

M4DM09.0 DA=070 KY=0 PP=0115,0116,077 BP=0470,01 TM=0:0:0

000001 002731 017600 000002 002734 002734 052525 125252 002246

EX**

In this example, the run list consists of unit program 4. After the descriptive information about the unit program is listed, error messages detected by the unit program are listed.

Example 3:

```

EX**
RU1,2,3,7.-----①
  UNIT PROGRAM 02 NOT FOUND

M4WC01.0 DA=074 KY=01 PP=0100 BP=0,02
M4ME03.0 DA=0 KY=02 PP=0101 BP=0,03
M4MT07.A DA=010 KY=03 PP=0102,0103,0104 BP=0,01
*****
*UNEXPECTED INTERRUPT
M4DM09.0 DA=070 KY=0 PP=0112,0113,077 BP=0242,01 TM=0:0:0
000001 002733 017200 000002 002734 002734 052525 125252 002246

EX**
RU1,7.-----②
M4WC01.0 DA=074 KY=01 PP=0100 BP=0,01
M4MT07.A DA=010 KY=02 PP=0101,0102,0103 BP=0,02
*****
EX**
CNS,300.
EX**
RU.-----③
M4WC01.0 DA=074 KY=05 PP=0300 BP=0,01
M4MT07.A DA=010 KY=06 PP=0301,0302,0303 BP=0,03
*****
EX**

```

- ① This RUN command has a run list consisting of unit programs 1, 2, 3, and 7. This command could have been terminated with a <cr> instead of a ".". This would not have changed the effect of the command in any way, since the two can be interchangeably used as command terminators, provided they are preceded by a run list.
- ② This RUN command specifies a run list consisting of unit programs 1 and 7.
- ③ The third RUN command does not specify a run list. Since the command is terminated with a ".", the run list specified in the previous command (namely 1 and 7) is used.

Example 4:

```
EX**
RU/D3

M4HE03.0 DA=0 KY=01 PP=0100 BP=0,03
*****
EX**
K0
<Control-C>
EX**
```

The operator wants to run unit program number 3 in the debug mode. After the EX** prompt appears, the operator can enter any command with the exception of the RUN command. (The command K shown in this example is explained later in this section.)

In order to exit from the debug mode of operation, the operator enters a <Control-C> character. Once he exits from the debug mode, the EX** prompt reappears.

NOTE

The <Control-C> character is shown in this example for illustrative purposes only. It does not echo on the terminal.

6.4.6 DISABLE COMMAND

6.4.6.1 Purpose

This command allows the operator to make specific unit programs or units inactive, that is, disable them. Disabled programs will not be executed by the EXEC, even if the operator requests that they be run (using the RUN command). In order to make a disabled program active again, the ENABLE command must be used.

6.4.6.2 Format

The DISABLE command has the following format:

$$DI[n_1, n_2, \dots] \left\{ \begin{array}{l} \langle cr \rangle \\ . \end{array} \right.$$

where:

n_1, n_2, \dots

Are the unit programs that are being disabled. The significance of n varies with the command terminator as in the case of the ENABLE command.

6.4.6.3 Usage

Use this command to disable specific unit programs. If no programs are specified in the command, the entire run list will be disabled. The ENABLE command must be used to reactivate disabled programs.

As in the case of the ENABLE command, the significance of n varies with the command terminator. If the command is terminated with a $\langle cr \rangle$, n has the following format:

xyy

where:

xx

Is the unit, platter, or line number.

yy

Is the device address.

If the command is terminated with a ".", n is the two-digit decimal number identification of the program.

6.4.6.4 Examples

Example 1:

```
DI2166<cr>
```

This specifies that the unit program for line number 21 and device address 66 is to be disabled.

Example 2:

```
DI2,7.
```

This specifies that unit programs 2 and 7 are to be disabled.

Example 3:

```
EX**
RU
M4UC01.0 DA=074 KY=01 PP=0100 BP=0,01
M4NE03.0 DA=0 KY=02 PP=0101 BP=0,03
M4NT07.A DA=010 KY=03 PP=0102,0103,0104 BP=0,01
M4DM09.0 DA=070 KY=04 PP=0105,0106,077 BP=0,01
*****
M4DM09.0 DA=070 KY=04 PP=0105,0106,077 BP=0137,01 TM=0:0:033
DCM TIMEOUT ERROR ON LINE 000003

EX**
DI070,170,270,370

EX**
RU
M4UC01.0 DA=074 KY=01 PP=0100 BP=0,02
M4NE03.0 DA=0 KY=02 PP=0101 BP=0,02
M4NT07.A DA=010 KY=03 PP=0102,0103,0104 BP=0,02
*****
+UNEXPECTED INTERRUPT
M4DM09.0 DA=070 KY=0 PP=0105,0106,077 BP=0711,01 TM=0:0:0
000001 002720 017200 000002 002734 002734 052525 125252 002246

EX**
RU
M4UC01.0 DA=074 KY=01 PP=0100 BP=0,01
M4NE03.0 DA=0 KY=02 PP=0101 BP=0,02
M4NT07.A DA=010 KY=03 PP=0102,0103,0104 BP=0,03
*****
```

In this example, the DCM unit program (M4DM09) is disabled by specifying the appropriate line numbers (0,1,2, and 3) and device address (70).

6.4.7 ENABLE COMMAND

6.4.7.1 Purpose

This command allows the operator to enable specific unit programs or units to be scheduled for execution. It is required only if these unit programs have been previously disabled by means of the DISABLE command.

6.4.7.2 Format

The ENABLE command has the following format:

$$\text{EN}[n_1, n_2, \dots] \left\{ \begin{array}{l} \langle \text{cr} \rangle \\ . \end{array} \right\}$$

where:

n_1, n_2, \dots

Are the unit programs that are being enabled to be scheduled for execution. The format of n varies, as explained in the following paragraph.

6.4.7.3 Usage

Use this command to reactivate a disabled program. If no unit programs are specified in the command, all the programs specified in the current run list are made active. The run list (see RUN command) specifies the unit programs which are to be scheduled for execution by the EXEC. If any one of these programs is disabled, it cannot execute until it is made active by an ENABLE command.

For this command, the command terminators are not interchangeable. If the command is terminated with a $\langle \text{cr} \rangle$, n has the format:

xyy

where:

xx

Is the unit, platter, or line number.

yy Is the device address.

If however, the command is terminated with a ".", n refers to unit program numbers (i.e., the two-digit decimal number identification).

6.4.7.4 Examples

Example 1:

```
EN3470<cr>
```

This specifies that the unit program for line number 34 and device address 70 must be enabled to be scheduled for execution.

Example 2:

```
EN110<cr>
```

This specifies that the unit program for unit number 1 and device address 10 must be enabled to be scheduled for execution.

Example 3:

```
EN1,3,7.
```

This specifies that unit programs 1, 3, and 7 must be enabled to be scheduled for execution.

6.4.8 CHANGE COMMAND

6.4.8.1 Purpose

The purpose of the CHANGE command is to display the contents of a particular location in a specific map key. This is useful when debugging a unit program. The command also gives the operator the capability to alter the contents of the location after examining it. This feature is an option.

6.4.8.2 Format

The command has the following format:

$$C(x) [,k] \left\{ \begin{array}{c} \langle cr \rangle \\ . \end{array} \right\} [d] \left\{ \begin{array}{c} \langle cr \rangle \\ ' \\ : \\ * \end{array} \right\}$$

where:

- x
Is the location whose contents are to be displayed.
- k
Is the map key of the desired location.
- d
Is a positive or negative octal number. It replaces the previous contents of location x.

$$\left\{ \begin{array}{c} \langle cr \rangle \\ ' \\ : \\ * \end{array} \right\}$$

One of these characters must be entered after d, depending on which one of the following options is desired:

- $\langle cr \rangle$ terminates the command.
- ,
- displays the contents of the next location, that is location (x+1).
- .
- displays the contents of the previous location, that is location (x-1).

* displays the contents of the indirect address, that is the address in location x.

6.4.8.3 Usage

To display the contents of location x in map key k, enter the following:

Cx,k<cr>

or

Cx,k.

To display the contents of location x in map key k, and then alter the contents of the location to d, the command format must be as follows:

Cx,k,d $\left\{ \begin{array}{l} \text{cr} \\ , \\ . \\ * \end{array} \right\}$

If only the location which has been displayed has to be examined and altered, terminate the command with a <cr>, thus:

Cx,k,d<cr>

If locations following or preceding the first location also need to be examined and altered, one of the other three command terminators ("",".", or "**") must be used. This automatically displays the address and contents of the appropriate location. The operator does not have to enter a new CHANGE command for each successive location.

A copy of each unit program is kept in map 0 of the first 32K words of memory. When a unit program is selected by the RUN command for execution, it is moved from map 0 to the key and physical pages assigned to it. When the operator changes a location in a unit program, this change is effected both in the key in which it is being run and in the permanent map 0 copy. Consequently, as the unit program moves around in memory, the changed location retains its new value (since the change has also been made in the permanent map 0 copy).

If the CHANGE command is entered after the EX** prompt without a map key being specified, the map key last selected is assumed.

The map key corresponding to each one of the unit programs being executed can be determined using a KEY command. (The KEY command

is described later in this section.) If there are indications (error messages) that a particular unit program must be scrutinized more closely, the map key of this program can be determined using the KEY command. Then, the contents of various locations in this map key can be examined (and altered if necessary) using the CHANGE command.

NOTE

The normal mode of operation for unit programs is the variable mode. In this mode, each time the unit program is rescheduled, it is assigned a new map key. Therefore, the map key must be ascertained before the CHANGE command is used. When the fixed mode of operation is being used, the map keys of the unit programs do not change after execution.

6.4.8.4 Examples

Example 1:

```
EX**  
C1651.(005001)2000,  
001652(005000)3157,  
001653(002000)5001,  
001654(003157)55070
```

In this example, the operator directs the EXEC to display the contents of location 1651. Since he terminates this part of the command with a ".", he can exercise the option to alter the contents of the location later. No map key is specified. Therefore, the map key last selected is assumed.

The numbers enclosed by the parentheses are the contents of location 1651. After they are displayed by the system (that is, output to the list device), the operator changes the contents to (octal) 2000. (Note that the leading zero for the octal number is omitted). The command is terminated with a comma (,). This causes the contents of the next location, namely 001652 to be displayed. The contents (005000) are changed to (octal) 3157. This command is also terminated with a comma. So the contents of the next location are displayed. After the contents of location 001654 are displayed and altered, the operator terminates the command with a <cr>. This terminates the CHANGE command, and the EX** prompt appears.

Example 2:

```
EX**
067006.(000000)7071, ①
067007(000000)13, ②
067010(000000)26914,
067011(000000)15031,
067012(000000)1016,
067013(000000)5712,
067014(000000)15001,
067015(000000)6455,
067016(000000)67023,
067017(000000)6150.
067016(067023)15777,
067017(006150)1000, ③
067017(001000).
067016(15777).
067015(006455).
067014(015001).
067013(005712) ④
EX**
```

- ① The contents of location 67006 are displayed and then altered to (octal) 7071. The command is terminated with a comma.
- ② The contents of the next location (067007) are automatically displayed, since the previous command was terminated with a comma.
- ③ After the contents of location 067020 are displayed, the operator enters a "." This results in the contents of the previous location (067017) being displayed next.
- ④ After the contents of location 067013 are displayed, the operator terminates the command by entering a <cr>.

Example 3:

```
EX**
C1651,0.(002000)2002, ①
001652(004654)4666, ②
001653(014677) ③
EX**
```

- ① The contents of location 1651 in map key 0 is displayed. A comma is then entered.
- ② Since the previous command was terminated with a comma, the next address (001652) and the contents of the location with that address are automatically displayed. After the display, the contents are altered to (octal) 4666, and a comma is entered.
- ③ After the contents of location 001653 are automatically displayed, the operator terminates the command by entering a <cr>.

6.4.9 CHANGE MAP COMMAND

6.4.9.1 Purpose

This command permits the operator to specify the starting map key and page to be assigned to the unit programs in the run list. The EDIT and CHANGE MAP commands are similar in that they both affect the map key and page assignment of unit programs in the run list. They differ in that the EDIT command applies to individual programs, whereas the CHANGE MAP command affects the entire run list.

6.4.9.2 Format

The CHANGE MAP command has the following format:

$$CM[k] [,p] \begin{Bmatrix} \langle cr \rangle \\ . \end{Bmatrix}$$

where:

k

Is the map key to be assigned for the next run. Can be any octal number from 1 to 17.

p

Is the starting memory page to be assigned for the next run. It can be any octal number from 100 to 3777.

6.4.9.3 Usage

Use this command to alter the map key and page assignment of the entire run list. The starting map key and memory page, that is the map key and page of the first program in the run list, is specified in the command. Since the map key and page assignment for the other programs in the run list follow in numerical sequence, the map key and page assignment of the entire run list is changed.

Command terminators can be used interchangeably.

6.4.9.4 Examples

```

EX**
RU1,7. ①
M4UC01.0 DA=074 KY=01 PP=0100 BP=0,01
M4MT07.A DA=010 KY=02 PP=0101,0102,0103 BP=0,02
*****
EX**
CH5,300. ②
EX**
RU. ③
M4UC01.0 DA=074 KY=05 PP=0300 BP=0,01
M4MT07.A DA=010 KY=06 PP=0301,0302,0303 BP=0,03
*****
EX**
RU1,3,6,7,8. ④
M4UC01.0 DA=074 KY=01 PP=0100 BP=0,01
M4NE03.0 DA=0 KY=02 PP=0101 BP=0,03
M4DH06.A DA=014 KY=03 PP=0102,0103,0104 BP=0,01
M4MT07.A DA=010 KY=04 PP=0105,0106,0107 BP=0,01
M4LP08.0 DA=035 KY=05 PP=0110,0111 BP=0,0
*****
M4DH06.A DA=014 KY=03 PP=0102,0103,0104 BP=0114,01 IM=0:0:02
DC 000014 LAST CCB EXECUTED-NONE
UNIT BUSY - ERROR
PRIMARY STATUS WORD 00
EX**
CH5,300. ⑤
EX**
RU ⑥
M4UC01.0 DA=074 KY=05 PP=0300 BP=0,01
M4NE03.0 DA=0 KY=06 PP=0301 BP=0,03
M4DH06.A DA=014 KY=07 PP=0302,0303,0304 BP=0,01
M4MT07.A DA=010 KY=010 PP=0305,0306,0307 BP=0,02
M4LP08.0 DA=035 KY=011 PP=0310,0311 BP=0,0
M4DH09.0 DA=070 KY=012 PP=0312,0313,077 BP=0,01
*****

```

① Unit programs 1 and 7 are run. Initially, unit program 1 is assigned map key 01 and page 0100. This is the starting assignment, and assignments for the other programs in the run list follow in sequence: unit program 7 is assigned map key 02 and pages 0101, 0102, and 0103.

② The CHANGE MAP command sets the starting map key to be 05 (leading zero is omitted in example), and the starting memory page to be 0300, for the next run.

- ③ The RUN command is entered again. Since it is terminated with a ".", the current run list is used. Because of the preceding CHANGE MAP command, unit program 1 is assigned map key 05 and page 0300, and unit program 7 is assigned map key 06 and pages 0301, 0302, and 0303.
- ④ The run list consists of unit programs 1, 3, 6, 7, and 8. Initially, unit program 1 is assigned map key 01 and page 0100. Assignments for the other unit programs follow in sequence.
- ⑤ The CHANGE MAP command sets the starting map key to be 05 and the starting memory page to be 0300, for the next run.
- ⑥ The RUN command is terminated with a <cr>, and so the entire index of unit programs is made the run list. Because of the preceding CHANGE MAP command, the starting map key and page assignments are 05 and 0300, respectively.

6.4.10 EDIT COMMAND

6.4.10.1 Purpose

The EDIT command provides the operator with the capability of altering the map key and pages assigned to a particular unit program in the run list.

6.4.10.2 Format

The EDIT command has the following format:

$$\text{ED} \left\{ \begin{array}{c} \cdot \\ \text{cr} \end{array} \right\} [k] [,p] \left\{ \begin{array}{c} \cdot \\ \text{cr} \end{array} \right\}$$

where:

k

Is the octal key (1 to 17).

p

Are the new page numbers (octal 100-3777).

6.4.10.3 Usage

It can be terminated with either a <cr> or a ".".

After the command is entered, the program ID of the first unit program in the run list is listed, with its assigned key and pages being used. The key and/or pages can be altered by entering new values for the key (k) and page numbers (p). The new value for k can be any octal number from 1 to 17. The new page numbers can have values from (octal) 100 to 3777.

If the pages are not being altered, terminate the command after specifying "k", as follows:

ED<cr>k<cr>

or

ED.k.

This process is repeated for all the unit programs in the run list.

CAUTION

The EXEC does not check for errors. Therefore, it is the operator's responsibility not to assign the same map key or pages to different unit programs.

NOTE

The RUN command which follows the EDIT command must specify the fixed memory mode (/F). If this is not specified, the new values for k and p will be ignored.

The CHANGE MAP (CMAP) and EDIT commands are similar. However, EDIT is used on a program-by-program basis, whereas CMAP sets the starting key and page assignment for the entire run list.

6.4.10.4 Examples

```
EX**
K----- ①
M4UC01.0 DA=074 KY=01 PP=0100 BP=0750,01
NAME03.0 DA=0 KY=03 PP=0745 BP=0751,03
M4DH06.A DA=014 KY=07 PP=0302,0303,0304 BP=0316,01
M4MT07.A DA=010 KY=010 PP=0305,0306,0307 BP=0744,01
M4LP08.0 DA=035 KY=011 PP=0310,0311 BP=0,0
M4DH09.0 DA=070 KY=012 PP=0312,0313,077 BP=0370,01
EX**
ED----- ②
M4UC01.0 DA=074 KY=01 PP=0100 BP=0750,01:01,0100 ③
NAME03.0 DA=0 KY=03 PP=0745 BP=0751,03:02,0101. ④
M4DH06.A DA=014 KY=07 PP=0302,0303,0304 BP=0316,01:03,0102.
M4MT07.A DA=010 KY=010 PP=0305,0306,0307 BP=0744,01:04,0103.
M4LP08.0 DA=035 KY=011 PP=0310,0311 BP=0,0:04,010
010 INVALID ⑤
M4DH09.0 DA=070 KY=012 PP=0312,0313,077 BP=0370,01:06,114

EX**
K----- ⑥
M4UC01.0 DA=074 KY=01 PP=0100 BP=0750,01
NAME03.0 DA=0 KY=03 PP=0745 BP=0751,03
M4DH06.A DA=014 KY=07 PP=0302,0303,0304 BP=0316,01
M4MT07.A DA=010 KY=010 PP=0305,0306,0307 BP=0744,01
M4LP08.0 DA=035 KY=011 PP=0310,0311 BP=0,0
M4DH09.0 DA=070 KY=012 PP=0312,0313,077 BP=0370,01
EX**
CMAP----- ⑦
```

- ① By using the KEY command, the operator lists the IDs of all the unit programs in the run list, their current key and pages.
- ② The EDIT command, terminated with a <cr>, is entered.
- ③ Information about the first program in the run list is displayed. After the ":", the operator enters the new values for key and page and a <cr>.
- ④ Information about the next unit program in the run list is displayed. The operator changes the map key of this program to 02, and the page to 0101. He then enters a ".". Information about the next program is then displayed, and so on, until all the unit programs in the run list are edited. (The operator has the option to alter or retain existing values.)
- ⑤ The operator attempts to change the value of <p> to 010. This is invalid, since only values from 0100 to 03777 are acceptable. The change is rejected, and the original value retained.
- ⑥ By using the KEY command, the operator verifies the new values assigned to the unit programs.
- ⑦ The unit programs are run in the fixed mode.

6.4.11 INSERT COMMAND

6.4.11.1 Purpose

The INSERT command stores a specified bit pattern into locations specified by the operator.

6.4.11.2 Format

The INSERT command has the following format:

$$I(x),(y)[,p][,k] \left\{ \begin{array}{l} \langle cr \rangle \\ . \end{array} \right\}$$

where:

- x
Is the address of the first memory location into which the bit pattern is to be stored.
- y
Is the address of the last memory location into which the bit pattern is to be stored.
- p
Is the bit pattern. The default value for p is -1(177777).
- k
Is the map key.

6.4.11.3 Usage

This command is useful for debugging purposes. The <cr> and "." can be interchangeably used as command terminators.

6.4.11.4 Examples

```
EX**  
I67000,67010,10101. ①  
EX**  
V67000,67077,0. ②
```

```
(067000) 010101 010101 010101 010101 010101 010101 010101 010101  
(067010) 010101 000000 000000 000000 000000 000000 000000 000000  
(067020) 000000 000000 000000 000000 000000 000000 000000 000000  
(067030) 000000 000000 000000 000000 000000 000000 000000 000000  
(067040) 000000 000000 000000 000000 000000 000000 000000 000000  
(067050) 000000 000000 000000 000000 000000 000000 000000 000000  
(067060) 000000 000000 000000 000000 000000 000000 000000 000000  
(067070) 000000 000000 000000 000000 000000 000000 000000 000000
```

```
EX**  
I67000,67077,10101,0. ③  
EX**  
V67000,67077,0. ④
```

```
(067000) 010101 010101 010101 010101 010101 010101 010101 010101  
(067010) 010101 010101 010101 010101 010101 010101 010101 010101  
(067020) 010101 010101 010101 010101 010101 010101 010101 010101  
(067030) 010101 010101 010101 010101 010101 010101 010101 010101  
(067040) 010101 010101 010101 010101 010101 010101 010101 010101  
(067050) 010101 010101 010101 010101 010101 010101 010101 010101  
(067060) 010101 010101 010101 010101 010101 010101 010101 010101  
(067070) 010101 010101 010101 010101 010101 010101 010101 010101
```

- ① The bit pattern 10101 is inserted in locations 67000 through 67010.
- ② After the insertion is made, the contents of locations 67000 through 67077 in map key 0 are viewed. (The VIEW command is discussed later in this section.)
- ③ The bit pattern 10101 is inserted in locations 67000 through 67077 in map key 0.
- ④ The contents of locations 67000 through 67077 in map key 0 are viewed.

6.4.12 KEY COMMAND

6.4.12.1 Purpose

This command has two options. One option allows the operator to set the debug map key to a particular value. This value is then used by all subsequent memory-accessing commands, until the value is altered by another KEY command. The second option allows the operator to list the ID of each unit program in the run list, with its current key and pages.

6.4.12.2 Format

The KEY command has the following format:

$$K[/L] [k] \left\{ \begin{array}{c} \langle cr \rangle \\ . \end{array} \right\}$$

where:

/L

Specifies that the list is to be sent to the line printer. This option is applicable only when the debug map key "k" is omitted.

k

Is the debug map key.

6.4.12.3 Usage

The $\langle cr \rangle$ and the "." can be interchangeably used as command terminators.

To set the debug map key to "k", enter the following:

$$Kk \left\{ \begin{array}{c} \langle cr \rangle \\ . \end{array} \right\}$$

To list the IDs of all the unit programs in the run list, with the current key and pages corresponding to each, on the operator's console, enter one of the following:

K<cr>

or

K.

To obtain the above listing on the line printer, enter one of the following:

K/L<cr>

or

K/L.

Note that the /L option is applicable only when k is not specified in the command.

6.4.12.4 Examples

```
EX**
K. ①
M4WC01.0 DA=074 KY=01 PP=0100 BP=01410,02
M4ME03.0 DA=0 KY=013 PP=01216 BP=01405,03
M4DH06.A DA=014 KY=012 PP=0773,0774,0775 BP=01000,01
M4MT07.A DA=010 KY=04 PP=0105,0106,0107 BP=01400,01
M4LP08.0 DA=035 KY=014 PP=01335,01336 BP=0,0
M4DM09.0 DA=070 KY=06 PP=0112,0113,077 BP=0164,01
EX**
REGA.05 INVALID
EX**
REGB.05 INVALID
EX**
K1. ②
EX**
```

① A list of all the unit programs in the run list is obtained by entering this command.

② The debug map key is set to the value 01. This value is then used by subsequent memory-accessing commands, until it is altered by another KEY command.

6.4.13 REG COMMAND

6.4.13.1 Purpose

The purpose of this command is to display the contents of a specified register at the time of the previous interrupt. A, B, X, or R registers can be specified in this command. These registers are referred to as pseudo registers.

The command provides the operator with the optional capability of altering or retaining the current value of the specified register.

6.4.13.2 Format

The REG command has the following format:

$$\text{REG}([A][B][X][R[n]]) \left\{ \begin{array}{c} \langle \text{cr} \rangle \\ . \end{array} \right\} [d] \left\{ \begin{array}{c} \langle \text{cr} \rangle \\ . \end{array} \right\}$$

where:

A, B, and X

Are the pseudo A, B, and X registers. They are equivalent to the R0, R1, and R2 registers.

R[n]

Refers to pseudo registers 0 to 7. n can have any value from 0 to 7. If n is not specified, or is greater than 7, all the registers are displayed.

d

The new positive or negative octal value which is to replace the current value of the specified register.

6.4.13.3 Usage

To display the contents of a particular register, but not alter it, enter the following:

$$\text{REG} \left\{ \begin{array}{c} A \\ B \\ X \\ R[n] \end{array} \right\} \langle \text{cr} \rangle$$

To display the contents of a particular register, and then alter it, enter:

$$\text{REG} \left\{ \begin{array}{l} \text{A} \\ \text{B} \\ \text{X} \\ \text{R}[n] \end{array} \right\} .d \left\{ \begin{array}{l} \langle \text{cr} \rangle \\ . \end{array} \right\}$$

NOTE

The first "." provides the operator with the option of altering or retaining the current value of the register. The next <cr> or "." terminates the command.

If the R register is selected, and n is not specified, as follows:

REGR<cr>

all the registers (0 to 7) will be displayed.

6.4.13.4 Examples

```

EX** ①
REGB.(000303)(000764)(103175)(000000)(001221)(000167)(052525)(125252)
EX**
REGX.(000303)(000764)(103175)(000000)(001221)(000167)(052525)(125252)
EX**
REGR
(000303)(000764)(103175)(000000)(001221)(000167)(052525)(125252)
EX**
REGR1 — ③
(000764)

```

- ① The contents of register B are displayed. By entering a "." after "REGB", the operator reserves the right to alter or retain the contents of the register after they are displayed by the system.
- ② The contents of the register are displayed by the system.
- ③ The contents of register R1 are displayed.

6.4.14 SEARCH COMMAND

6.4.14.1 Purpose

The SEARCH command allows the operator to examine all locations within a specified range and in a specified map key for a particular bit pattern.

6.4.14.2 Format

The SEARCH command has the following format:

$$S[/L](x),(y)[,p][,m][,k] \begin{cases} \langle cr \rangle \\ . \end{cases}$$

where:

/L

Specifies that the listing is to be output on the line printer.

x

Location at which the search is to begin.

y

Location at which the search is to end.

p

Is the bit pattern for which the search is conducted. The default value is -1(177777).

m

Is the mask which masks the bit pattern, p. The default value is -1(177777).

k

Is the map key.

6.4.14.3 Usage

Use this command to search for a specific bit pattern, either masked or unmasked by another pattern. A mask is a pattern of characters, (in this case, a bit pattern), that is used to control the retention or elimination of portions of another pattern of characters.

Example:

If the bit pattern 77 (i.e., 111 111) is masked by 55 (i.e., 101 101) the portions of the patterns that are retained (i.e., searched for) are the following:

1Ø1 1Ø1

Therefore, locations with the bit patterns 77, 75 (i.e., 111 101), and 55 are found as a result of the search.

When a location is found, whose contents match the bit pattern specified by the operator, the memory location and its contents are output.

6.4.14.4 Examples

Example 1:

S/L1000,1177,77,55,0.

Locations from 1000 to 1177, in map key 0, are searched for the bit pattern 77, masked by 55. The resulting matches are listed on the line printer (/L option), as follows:

001051(004155)
001066(004155)
001105(014677)
001147(010475)
001157(001155)

Example 2:

S/L1000,1177,55,55,0.

Locations from 1000 to 1177, in map key 0, are searched for the bit pattern 55 masked by 55, and the matches are listed on the line printer. This command results in the following output:

001051(004155)
001066(004155)
001105(014677)
001147(010475)
001157(001155)

: patterns

for the
ent to

pattern
are not

6.4.15 TRAP COMMAND

6.4.15.1 Purpose

This command provides the operator with a breakpoint facility for debugging unit programs.

6.4.15.2 Format

The TRAP command has the following format:

$$T[x][,y][,k] \left\{ \begin{array}{l} \langle cr \rangle \\ . \end{array} \right\}$$

where:

x

Is the location where the breakpoint is set. If omitted, the command is interpreted as a jump to location y.

y

Is the location to which a jump is executed when the breakpoint is reached. If omitted, the command is interpreted as a jump to the previous trap address.

k

Is the map key.

6.4.15.3 Usage

This command is very useful for debugging purposes. In the command, if <x> is omitted, the command is interpreted as a jump to location <y>. If <y> is omitted, the command is interpreted as a jump to the previous trap address.

The following three options are available for continuing from the trap:

1. A new trap can be set before continuing, by entering the following:

$$T[x][,y] \left\{ \begin{array}{l} \langle cr \rangle \\ . \end{array} \right\}$$

2. To continue at address y after the trap, enter:

$$T, [y] \begin{Bmatrix} \langle cr \rangle \\ . \end{Bmatrix}$$

3. To continue at the previous trap address, enter:

$$T \begin{Bmatrix} \langle cr \rangle \\ . \end{Bmatrix}$$

If the trap address is encountered during program execution, a message is output, which has the following format:

```
*x(y)(z) KY=k  
r0 r1 r2 r3 r4 r5 r6 r7 pc
```

where:

*

Is the EXEC message.

x

Is the trap address.

y

Represents the contents of trap address.

z

Represents the contents of (trap address + 1).

r0-r7

Are the registers 0 to 7. (Registers 0, 1, and 2 are registers A, B, and X, respectively).

pc

Is the program counter for return (i.e., trap address +2).

If the trap address is never encountered by the executing program, the trap gets cleared when the next RUN command is executed.

NOTE

A trap is set by storing a jump to M\$TRAP at the specified address in the unit program. (Refer to Section 5 for information about M\$TRAP). The original contents of the two locations are saved in the TIDB (Task Identification Block). The trap is also stored into the permanent copy of the program in map.0. Consequently, the program "remembers" the trap even when it is moved around in memory. This is particularly useful when a trap is set in the error detection code to debug intermittent problems.

6.4.15.4 Examples

```
EX**
T1754,1000 — (1)

*001754(001010)(001770) KY=01
000001 000020 101021 000000 002203 177777 052525 125252 001756
```

```
EX**
T,1770 — (2)

M4DF04.A DA=016 KY=01 PF=0100,0101,0102 BP=0103,01 TM=0:0:02
BIC 000020 ABNORMAL BIC DEVICE STOP READ ERROR
INITIAL ADDRESS 004000
FINAL ADDRESS WAS 004740 SHOULD BE 004740
```

```
EX**
RU/DA

M4DF04.A DA=016 KY=01 PF=0100,0101,0102 BP=0,03
*****
EX**
T1754,1000,1 — (3)

*001754(001010)(001770) KY=01
000001 000020 101021 000000 002203 177777 052525 125252 001756
```

- (1) The operator specifies that a trap is to be set at location 01754. After the trap is encountered a jump to location 01000

is to be made. When the trap is encountered, the EXEC outputs the following information:

- a. The trap address (001754)
- b. The contents of the trap address (001010)
- c. The contents of trap address + 1 (001770)
- d. Map key (01)
- e. Contents of registers 0 to 7
- f. Program counter for return (001756).

- ② The command is interpreted as a jump to 01770. Execution will continue from this location after the trap is encountered.
- ③ A trap is set at location 01754, and after that trap is encountered, a jump to location 01000 in map key 01 is specified.

6.4.16 VIEW COMMAND

6.4.16.1 Purpose

The VIEW command displays the contents of specified memory locations.

6.4.16.2 Format

The VIEW command has the following format:

```
V[/L](x)[,y][,k] {<cr>}
                  { . }
```

where:

- x
Is the address of the first memory location whose contents are to be displayed. It is the lower limit of the range of locations.
- y
Is the address of the last memory location (the upper limit of the range) whose contents are to be displayed.
- k
Is the map key.

6.4.16.3 Usage

To view the contents of memory locations x through y, in map key k, on the operator's console, enter the following:

```
Vx,y,k.  
or  
Vx,y,k<cr>
```

NOTE

The <cr> and "." can be interchangeably used as command terminators.

If <y> is omitted, the dump continues until some character is entered on the operator's console. If this character is a comma, the dump continues. All other characters terminate the dump.

6.4.16.4 Examples

Example 1:

```
EX**
U17600

(017600) 017300 100000 000007 000000 000001 002731 017600 000002
(017610) 002734 002734 052525 125252 002246 146716 006746 006546
(017620) 000002 000000 000030 052525 125252 002006 000000 013473
(017630) 000000 000000 000000 000070 000010 000077 140070 140170
(017640) 000000 000000 000000 000000 000000 000000 000000 000000
EX**
```

In the above example, the operator specifies that he wants to view the contents of locations beginning with location 17600. The last location to be viewed is not specified. The display is terminated when the operator enters some character (other than a comma) on the operator's console. If the operator had entered a comma, the display would have continued.

Example 2:

```
EX**
U67000,67077,0.

(067000) 000000 000000 000000 000000 000000 000000 000000 000000
(067010) 000000 000000 000000 000000 000000 000000 000000 000000
(067020) 000000 000000 000000 000000 000000 000000 000000 000000
(067030) 000000 000000 000000 000000 000000 000000 000000 000000
(067040) 000000 000000 000000 000000 000000 000000 000000 000000
(067050) 000000 000000 000000 000000 000000 000000 000000 000000
(067060) 000000 000000 000000 000000 000000 000000 000000 000000
(067070) 000000 000000 000000 000000 000000 000000 000000 000000
```

In this example, the operator specifies that he wants to view the contents of locations 067000 through 067077 in map key 0.

6.5 SUMMARY OF OPERATOR COMMANDS

| <u>Command Name</u> | <u>Function</u> | <u>Reference Section Number</u> |
|---------------------|--|---|
| CHANGE | Displays contents of a particular location in a specific map key. | 6.4.8 |
| CHANGE MAP | Specifies starting map key and page assignment for all the unit programs in the run list. | 6.4.9 |
| DISABLE | Disables specific unit programs. | 6.4.6 |
| EDIT | Alters the map key and pages assigned to a specific unit program in the run list. | 6.4.10 |
| ENABLE | Reactivates unit programs previously disabled by the DISABLE command. | 6.4.7 |
| HELP | Lists tutorial information about all the operator commands. | 6.4.2 |
| INDEX | Lists all the unit programs residing in map key 0 by ID, device address, and TIDB. | 6.4.3 |
| INSERT | Stores a specified bit pattern into locations specified by the operator. | 6.4.11 |
| KEY | Has two options. One option sets the debug map key to a particular value. The other option lists the ID of each unit program in the run list with its current key and pages. | 6.4.12 |

| <u>Command Name</u> | <u>Function</u> | <u>Reference Section Number</u> |
|---------------------|--|---------------------------------|
| LOG | Lists the entries made in the Error Log during the previous run of the system Test. | 6.4.4 |
| REG | Displays (and optionally alters) the contents of a specified register: A, B, X, or R. | 6.4.13 |
| RUN | Executes unit programs scheduled for execution, and lists information about them. | 6.4.5 |
| SEARCH | Searches through all the locations within a specified range for a specified bit pattern. | 6.4.14 |
| TRAP | Sets breakpoints | 6.4.15 |
| VIEW | Displays contents of specified memory locations. | 6.4.16 |

6.6 EXECUTIVE SERVICE CALLS

All interrupt and I/O functions required by the unit programs are performed by the Executive Program (EXEC). The unit programs request these services by issuing Executive service calls. These calls execute a jump instruction to the address of the called service routine. For more details about Executive Service calls, refer to Section 5.

6.7 SAMPLE SESSION

```
V70-SERIES SYSTEM TEST
*****
FOR COMMAND LIST,TYPE HE.
EXEC PROMPTER IS EX**
ENTER SPACE IF OUTPUT PAUSES
MEMORY SIZE= 0256K
*****
EX**
HE.
EXECUTIVE COMMANDS
NOTE:ALL COMMANDS END WITH CR OR .
/L=OUTPUT TO LPR
K=MAP KEY
HELP:PRINT COMMAND LIST
HEC/LJ
INDEX:PRINT UNIT PROGRAM LIST
INC/LJ
RUN:RUN UNIT PROGRAMS
RUC/LJ[FJ/DJ]N1,N2,...J
/F=FIXED MODE
/D=DEBUG MODE
ENABLE:ENTER UNIT PGMS IN RUN LIST
ENC1,N2,...J
DISABLE:DELETE UNIT PGMS FROM RUN LIST
DIC1,N2,...J
EDIT:DISPLAY/ALTER KEYS,PAGES
ED(CR OR .)[K]C,PJ
CHANGE MAP:ALTER NEXT MAP KEY,PAGE
CHKJ,C,PJ
LOG:PRINT ERROR LOG TABLE
LOC/LJ
REGISTER:DISPLAY/ALTER REGISTERS
([A][B][X][R][N][J])[C,K](CR OR .)[D](T)
NOTE:D=NEW DATA,IF ANY
T=COMMA DISPLAYS/ALTERS NEXT REGISTER
=OTHER TERMINATES COMMAND
KEY:DISPLAY KEYS/SET DEBUG MAP KEY
KC/LJ[N]
CHANGE:DISPLAY/ALTER MEMORY IN K
C(X)[C,K](CR OR .)[D](T)
NOTE:D=NEW DATA,IF ANY
T: ,=NEXT; .=PREV; *=INDIRECT; CR=END
VIEW:FROM X TO Y,IN K,ON LPR OR TTY
VC/LJ(X)[C,Y]C,KJ
INSERT:INSERT P INTO X TO Y,IN K
I(X),(Y)[C,P]C,KJ
SEARCH:FROM X TO Y,IN K,FOR P MASKED BY M
SC/LJ(X),(Y)[C,P]C,MJ,C,KJ
TRAP:TRAP AT X,GO TO Y,IN K
T[X]C,[Y]C,KJ ERROR LOG TABLE
```

EX**
LO.
ERROR LOG TABLE
UNIT PROGRAM ERRORS

| | |
|-----|---|
| 0 | 0 |
| 01 | 0 |
| 02 | 0 |
| 03 | 0 |
| 04 | 0 |
| 05 | 0 |
| 06 | 0 |
| 07 | 0 |
| 08 | 0 |
| 09 | 0 |
| 010 | 0 |
| 011 | 0 |
| 012 | 0 |
| 013 | 0 |
| 014 | 0 |
| 015 | 0 |

EX**

IN.

EXEC00.A DA=0 TIDB=016400
M4UC01.0 DA=074 TIDB=016500
M4FP02.0 DA=0 TIDB=016600
M4NE03.0 DA=0 TIDB=016700
M4DF04.A DA=016 TIDB=017000
M4DC05.A DA=015 TIDB=017100
M4DH06.A DA=014 TIDB=017200
M4HT07.A DA=010 TIDB=017400
M4LP08.0 DA=035 TIDB=017500
M4DH09.0 DA=070 TIDB=017600
M4DH09.0 DA=066

EX**

DI4,5.

EX**

DI8.

EX**

RU1.

M4UC01.0 DA=074 KY=01 PP=0100 BP=0,0

EX**

K.

M4UC01.0 DA=074 KY=02 PP=0612 BP=0624,02

EX**

RU.

M4UC01.0 DA=074 KY=01 PP=0100 BP=0,02

EX**
K.
M4WC01.0 DA=074 KY=03 PP=0425 BP=0503,01
EX**
C1030.(003306)

EX**
C1030,3.(006030)

EX**
RU.
M4WC01.0 DA=074 KY=01 PP=0100 BP=0,01

EX**
K.
M4WC01.0 DA=074 KY=03 PP=0425 BP=0743,01
EX**
C333.(000003)

EX**
RU2.
M4FP02.0 DA=0 KY=01 PP=0100 BP=0,0

EX**
K.
M4FP02.0 DA=0 KY=016 PP=0533 BP=0,0

EX**
ENB.
EX**
RUB

M4LP00.0 DA=035 KY=01 PP=0100,0101 BP=0,01

EX**
RU3.
M4ME03.0 DA=0 KY=01 PP=0100 BP=0,0

EX**
RU9.

M4DM09.0 DA=070 KY=06 PP=0113,0114,077 BP=0,01

*UNEXPECTED INTERRUPT
M4DM09.0 DA=070 KY=06 PP=0113,0114,077 BP=0,01 TM=0:0:0
000001 002727 017600 000002 002734 002734 052525 125252 002246

6.8 ERROR DETECTION

During System Test operation, errors (if any) are detected by the Preliminary Loader, the EXEC, and the unit programs. Errors detected by the Preliminary Loader are discussed in Section 4. Errors detected by the individual unit programs are discussed in Sections 7 through 16. Errors detected by EXEC are discussed in this section.

Two types of errors are detected by the EXEC. They are:

- a. Errors directly related to unit programs. Error messages for such errors are identified by a leading *.
- b. Error messages not directly related to unit programs.

After the error message is output, EXEC continues normal processing. However, for some types of errors, the unit program causing the error may be aborted until its next scheduled execution.

6.8.1 MAP ERRORS

If the EXEC detects a map error while storing a unit program into its assigned map key, it generates an error message having the following format:

```
*PGM STORE ERROR IN PAGE
uuuuuu.u DA=d KY=kk PP=pppp BP=b,n TM=hh:mm:sss
```

where:

- u
Is the unit program ID. (See Section 5 for the format of the Program ID).
- d
Is the Device Address.
- k
Is the map key.
- p
Are the physical pages being used by the unit program.

b Are the buffer pages being used.

n Is the actual number of buffer pages.

hh:mm:ss
Is the time:
• hh - Hours
mm - Minutes
ss - Seconds

Example:

EX**
RUZ

```
*PGM STORE ERROR IN PAGE  
M4MT07.A DA=010 KY=01 PP=0134,0150,0151 BP=0,00134  
*PGM STORE ERROR IN PAGE  
M4MT07.A DA=010 KY=01 PP=0152,0153,0154 BP=0,00152  
*PGM STORE ERROR IN PAGE  
M4MT07.A DA=010 KY=01 PP=0155,0156,0157 BP=0,00155  
*PGM STORE ERROR IN PAGE  
M4MT07.A DA=010 KY=01 PP=0160,0161,0162 BP=0,00160
```

6.8.2 COMMAND ERRORS

Invalid entries in a command generate the following message:

??

If an invalid map key is specified (or used by default) in a memory accessing command, the following message is generated:

```
xx INVALID
```

If a unit program number that is not in the library of unit programs is specified, the following message is generated:

```
UNIT PROGRAM xx NOT FOUND
```

Example:

```
EX**  
K0  
0 INVALID  
EX**  
C3500  
(001016)
```

```
EX**  
RU3
```

```
NAME03.0 DA=0 KY=0 PP=0100 BP=0,03  
*****
```

6.8.3 EXECUTION ERRORS

6.8.3.1 Print Request Error

If the number of print requests from unit programs exceeds 16, the following message is generated:

```
*PRINT REQUEST TABLE FULL
```

Unit programs making print requests which exceed this limit are aborted.

6.8.3.2 Invalid BIC Request

An invalid BIC request generates the following message:

```
*BIC xx ADDRESS ERROR  
uuuuuu.u DA=d KY=kk PP=pppp BP=b,n TM=hh:mm:sss
```

where:

- x Is the BIC device address.
- u Is the unit program ID.
- d Is the Device Address.
- k Is the map key.
- p Are the physical pages.
- b Are the buffer pages.
- n Is the number of buffer pages being used.

hh:mm:ss

Is the time:
hh - Hours
mm - Minutes
ss - Seconds

6.8.3.3 BIC Request Error

If the number of BIC requests made by unit programs exceeds 8, the following message is generated:

*BIC xx REQUEST TABLE FULL

Unit programs making BIC requests after this limit has been reached are aborted.

6.8.3.4 Memory Protect Error

If a memory protect error occurs, and the computer is a V77-800 computer, a message with the following format is generated:

*IIA xx,MPI cc, aaaaaa(iiiii) (mmmmm) (rrrrr)
uuuuu.u DA=yy KY=kkk PP=ppp BP=bbb,nn TM=hh:mm:ss
r0 r1 r2 r3 r4 r5 r6 r7 pc

where:

xx

Is the internal interrupt address (or PE if parity error). The significance of this address is as follows:

| <u>xx</u> | <u>Significance</u> |
|-----------|------------------------|
| 20 | Non-jump (see cc code) |
| 22 | - |
| 24 | - |
| 26 | Jump |

cc

Is the memory protect interrupt code (applies only to V77-800 computer). The significance of the cc code is as follows:

| <u>cc</u> | <u>Significance</u> |
|-----------|---------------------|
| 0 | Privilege |
| 1 | Instruction fetch |
| 2 | Write Protect |
| 3 | Unassigned |

a

Is the address of the instruction causing the violation.

i

Is the instruction or ***** is unassigned address violation.

m

Is the map state register.

r

Is the map unassigned address register.

u

Is the unit program ID.

y

Is the device address.

k

Is the map key.

p

b Are the buffer pages.
n Is the number of buffer pages.

hh:mm:ss
Is the time:
hh - Hours
mm - Minutes
ss - Seconds

r0-r7
Are the pseudo registers.

pc
Is the program counter for return.

The unit program causing the violation is aborted.

If a memory parity error occurs, and the computer is a V77-600, a message with the following format is generated:

```
*IIA xx,aaaaaa(iiiii) (sssss) (wwwww)
uuuuu.u DA=yy KY=kkk PP=ppp BP=bbb,nn TM=hh:mm:ss
r0 r1 r2 r3 r4 r5 r6 r7 pc
```

where:

xx
Is the internal interrupt address (or PE is parity error). The significance of this address is as follows:

| <u>xx</u> | <u>Significance</u> |
|-----------|---------------------|
| 20 | Halt |
| 22 | I/O |
| 24 | Write Protect |
| 26 | Jump |
| 30 | Unassigned |
| 32 | Instruction Fetch |
| 34 | Data Transfer |

a
Is the address of the instruction causing the violation.

i
Is the instruction or ***** if unassigned address violation.

s
Is the processor status word.

w
Is the cache status word.

u, y, k, p, b, n, hh:mmm:sss, r0-r7, pc
Refer to the explanation of these parameters in the
discussion of the V77-600 memory protect error message.

The unit program causing the violation is aborted.

Examples:

The following are examples of memory parity error messages
(V77-600 computer):

Example 1

```
*IIA 26,001264(001010) (046044),(077777)
M4MT07.A DA=010 KY=04 PP=0105,0107,0110 BP=0,01 TM=0:0:0
177772 000224 100610 177777 001221 000000 052525 125252 001573
```

The internal interrupt address is 26. The address of the instruction causing the violation is 001264. The instruction itself is 001010. The processor status word is 046044. The cache status word is 077777. The second line of the error message gives information about the unit program causing the violation. The third line shows the contents of the pseudo registers (r0-r7) and the program counter for return (pc).

Example 2

```
*IIA 30,001214***** (046045),(003240)
M4LP08.0 DA=035 KY=05 PP=0106,0107 BP=0,0 TM=0:0:0
000026 000223 001611 000000 001221 000000 052525 125252 001214
```

This is an example of a memory parity error message which indicates (by means of the "*****") that it is an unassigned address violation.

6.8.3.5 Unexpected Interrupt

An unexpected interrupt from an I/O device generates an error message with the following format:

```
*UNEXPECTED INTERRUPT
uuuuuu.u DA=yy KY=kkk PP=ppp BP=bbb,nn TM=hh:mm:sss
r0 r1 r2 r3 r4 r5 r6 r7 pc
```

where:

- u
Is the unit program ID.
- y
Is the device address.
- k
Is the map key.
- p
Are the physical pages used by the program.
- b
Are the buffer pages.
- n
Is the number of buffer pages used by the program.
- hh:mm:sss
Is the time:
 - hh - Hours
 - mmm - Minutes
 - sss - Seconds
- r0-r7
Are the pseudo registers.
- pc
Is the program counter for return.

Example:

EX**
RU

```
M4UC01.0 DA=074 KY=01 PP=0100 BP=0,01
M4ME03.0 DA=0 KY=02 PP=0101 BP=0,02
M4DF04.A DA=016 KY=03 PP=0102,0103,0104 BP=0,04
M4DH06.A DA=014 KY=04 PP=0105,0106,0107 BP=0,04
M4MT07.A DA=010 KY=05 PP=0110,0111,0112 BP=0,03
M4LP08.0 DA=035 KY=06 PP=0113,0114 BP=0,01
M4DM09.0 DA=070 KY=07 PP=0115,0116,077 BP=0,01
*****
*UNEXPECTED INTERRUPT
M4DM09.0 DA=070 KY=07 PP=0115,0116,077 BP=0,01 TM=0:0:0
000001 002716 017600 000002 002734 002734 052525 125252 002246
```

This is an example of the type of error message generated when an unexpected interrupt is received from an I/O device. The second line of error message identifies the unit program (M4DM09.0) and gives descriptive information about it. The third line gives the contents of the pseudo registers (r0-r7) and the program counter for return.

6.8.3.6 I/O Request Error

An invalid I/O request from a unit program generates an error message with the following format:

```
*INVALID I/O INSTRUCTION xxxxxx
uuuuuu.u DA=yy KY=kkk PP=ppp BP=bbb,nn TM=hh:mm:sss
r0 r1 r2 r3 r4 r5 r6 r7 pc
```

where:

- x
Is the invalid instruction.
- u
Is the unit program ID.
- y
Is the device address.

k Is the map key.

p Are the pages used by the unit program.

b Are the buffer pages being used.

n Is the number of buffer pages.

hh:mm:ss
Is the time:
hh - Hours
mm - Minutes
ss - Seconds

r0-r7
Are the pseudo registers.

pc
Is the program counter for return.

The unit program making the invalid requests is aborted.

Example:

```

EX**
RUB.
M4LP08.0 DA=035 KY=01 PP=0102,0103 BP=0,0
*****
*INVALID I/O INSTRUCTION 001000
M4LP08.0 DA=035 KY=01 PP=0102,0103 BP=0,0 TM=0:0:0
000000 000226 001000 000000 001400 000000 052525 125252 001136

```

When unit program M4LP08.0 makes an invalid I/O request, the above error message is generated.

6.8.3.7 Unit Program Errors

Messages generated as a result of errors detected by individual unit programs are described in Sections 7 through 16.

The Executive Program (EXEC) generates a message which precedes each unit error message. This EXEC message has the following format:

uuuuuu.u DA=yy KY=kkk PP=ppp BP=bbb,nn TM=hh:mm:ss

where:

- u
Is the unit program ID.
- y
Is the device address.
- k
Is the map key.
- p
Are the pages being used by the program.
- b
Are the buffer pages.
- n
Is the number of buffer pages.

hh:mm:ss

Is the time:

hh - Hours
mmm - Minutes
sss - Seconds

Example:

M4DH06.A DA=014 KY=03 PP=0102,0103,0104 BP=0116,01 TM=0:0:02-①
DC 000014 LAST CCB EXECUTED-NONE
UNIT BUSY - ERROR
PRIMARY STATUS WORD 002400 STATUS WORD UNIT 000000
B8/10 - OFF LINE

M4LP08.0 DA=035 KY=05 PP=0110,0111 BP=0252,01 TM=0:0:010-②
LP PRINTER ERROR

① and ② are messages generated by EXEC. They precede error messages generated by individual unit programs. The EXEC message identifies the unit program generating the error message.

Section 7

V77-600 Writable Control Store (F3034) Unit Program

7.1 PURPOSE AND OPERATION

The V77-600 Writable Control Store (WCS) unit program (M4WC01) tests the operation of the WCS. This unit program executes firmware instructions written into the WCS, by moving data through a buffer. The buffer extends from the end of the unit program to the end of the map in which the program is currently executing.

The program continues in operation until the function of the program is completed. When erroneous results are obtained, or equipment malfunction occurs, error messages are generated.

7.2 WCS ERROR MESSAGES

The first line of error messages generated by the unit program identifies the unit program, and is created by the EXEC. (Refer to Section 4). The remainder of the error message is generated by the unit program itself, and consists of:

1. The identification "WCS".
2. Device Address of the unit.
3. Description of the error that has been detected.

Error messages generated by the WCS unit program are listed in this section. Only the descriptive portion of the error message is described here, since the identification and the device address portions are the same for all the error messages generated by the unit program.

<explanatory message>

or

TIMEOUT <explanatory message>

where:

<explanatory message> is one or more of the following:

WCS INITIALIZATION ERROR
WCS SELECT ERROR
WCS DATA OUT ERROR
WCS EXECUTION ERROR
MEMORY ADDRESS xxxxxx
DATA EXPECTED xxxxxx FOUND xxxxxx

7.3 EXAMPLES

Example 1

EX**

RU/D1

M4WC01.0 DA=074 KY=01 PP=0100 BP=0,0

M4WC01.0 DA=074 KY=01 PP=0100 BP=0101,01 TM=0:0:02

WCS 000074 WCS EXECUTION ERROR

MEMORY ADDRESS 002234 DATA EXPECTED 125252 FOUND 000000

*IIA 30,001123***** (046042), (025252)

M4WC01.0 DA=074 KY=02 PP=0102 BP=0103,01 TM=0:0:05

125252 125441 103175 000000 001221 000167 052525 125252 001203

EX**

Example 2

EX**

RU/D1

M4WC01.0 DA=074 KY=01 PP=0100 BP=0,01

M4WC01.0 DA=074 KY=01 PP=0100 BP=0,01 TM=0:0:01

WCS 000074 TIMEOUT WCS INITIALIZATION ERROR

*IIA 30,001123***** (046042), (025252)

M4WC01.0 DA=074 KY=02 PP=0101 BP=0102,01 TM=0:0:03

125252 125441 103175 000000 001221 000167 052525 125252 001203

Section 8

V77-600 Floating-Point Processor (F3027)

Unit Program

8.1 PURPOSE AND OPERATION

The V77-600 Floating-Point Processor (FPP) unit program (M4FP02) verifies the correct operation of the FPP. The unit program performs this by executing floating-point arithmetic instructions in the FPP. These instructions are executed to move data through a buffer area. The buffer area extends from the end of the unit program to the end of the map in which the program is currently executing. At the end of this operation, the data (result) obtained is compared with the data expected. Discrepancies result in the generation of error messages.

8.2 FPP ERROR MESSAGES

All error messages generated by this unit program are preceded by an EXEC message which identifies the unit program. (Refer to Section 4). The FPP error message consists of the identification "FPP" and descriptive information about the error.

It has the following format:

```
FPP CALCULATION DATA EXPECTED xxxxxx xxxxxx FOUND xxxxxx xxxxxx
```

8.3 EXAMPLES

EX**

RU/D2

M4FP02.0 DA=0 KY=01 PP=0100 BP=0,0

M4FP02.0 DA=0 KY=01 PP=0100 BP=0,0 TH=0:0:01

FPP CALCULATION DATA EXPECTED 040526 077024 FOUND 123456 054321

Section 9

V77-800 Writable Control Store (F3057) and Floating-Point Processor (F3053) Unit Program

9.1 PURPOSE AND OPERATION

The V77-800 Writable Control Store (WCS) unit program (M4WC15) verifies the correct operation of the WCS. Firmware instructions written into the WCS are executed. Each set of instructions executed tests various functions and capabilities of the firmware, and is loaded as 24 separate tests. One of these tests verifies the current operation of the V77-800 Floating-Point Processor (FPP).

9.2 WCS and FPP (V77-800) ERROR MESSAGES

Any errors detected by the unit program cause error messages to be generated. The error messages are preceded by the standard EXEC message which identifies the unit program. (See Section 4). Error messages related to operation of the WCS contain the following:

1. The identification "WCS".
2. The Device Address of the unit.
3. Descriptive information about the error.

Error messages related to the operation of the FPP contain the identification "FPP" and descriptive information about the error.

The following is a list of WCS error messages:

WCS READ ERROR
WCS WRITE ERROR
TEST NUMBER xx
REGISTER FOUND EXPECTED

R0 xxxxxx xxxxxx
R1 xxxxxx xxxxxx
R2 xxxxxx xxxxxx

R3 xxxxxx xxxxxx
R4 xxxxxx xxxxxx
R5 xxxxxx xxxxxx
R6 xxxxxx xxxxxx
R7 xxxxxx xxxxxx

9.3 EXAMPLES

WCS 074 TEST NUMBER 4

| REGISTER | FOUND | EXPECTED |
|----------|--------|----------|
| R0 | 001115 | 000000 |
| R1 | 006020 | 000001 |
| R2 | 000010 | 000001 |
| R3 | 101146 | 177777 |
| R4 | 001221 | 177777 |
| R5 | 000167 | 000001 |
| R6 | 052525 | 000000 |
| R7 | 125252 | 000000 |

Section 10

Memory Unit Program

10.1 PURPOSE AND OPERATION

The Memory unit program (M4ME03) tests the operation of memory in the V77-600 and V77-800 computers. The program ascertains the operational status of the computer memory and assists in locating and correcting faults.

The program operates as follows:

1. It writes data into memory.
2. It reads the data back.
3. It executes the instructions written into memory using all available memory from the end of the unit program to the end of its assigned map.
4. Errors detected cause error messages to be generated.

10.2 MEMORY ERROR MESSAGES

All the error messages are preceded by the EXEC message which identifies the unit program generating the error message. The error message itself consists of:

1. The identification "MEM".
2. Descriptive error information.

The following error messages are output by the unit program:

MEM DATA COMPARE ERROR

BUFFER ADDRESS xxxxxx SIZE (OCTAL) xxxxxx
DATA EXPECTED xxxxxx FOUND xxxxxx LOCATION xxxxxx

MEM INSTRUCTION EXECUTION ERROR

BUFFER ADDRESS xxxxxx SIZE (OCTAL) xxxxxx
DATA EXPECTED xxxxxx FOUND xxxxxx LOCATION xxxxxx

10.3 EXAMPLES

EX**
RU/D3

M4NE03.0 DA=0 KY=01 PP=0100 BP=0,0

M4NE03.0 DA=0 KY=01 PP=0100 BP=0101,02 TM=0:0:01
MEM DATA COMPARE ERROR
BUFFER ADDRESS 002000 SIZE (OCTAL) 001700
DATA EXPECTED 125252 FOUND 000000 LOCATION 002000

Section 11

Cartridge Disk Unit Program

11.1 PURPOSE AND OPERATION

The Cartridge Disk unit program (M4DF04) verifies the correct operation of the F3094, F3096, F3310, F3016, and Models 70-7600 and 70-7610 (types DB and DF) cartridge disks and controllers. (Refer to Section 2 for a list of the Software Model codes corresponding to these disks).

The unit program operates as follows:

1. It writes records from memory onto cartridge disk units on a controller, alternating between units.
2. It then reads the records from the disk units back into memory.
3. The data read is compared with the data written to ensure a valid write/read operation.
4. If errors are detected, error messages are generated.

11.2 ERROR MESSAGES

All error messages output by this unit program are preceded by the EXEC message which identifies the unit program generating the error message. The error message itself consists of the following:

1. Identification:

DC - For Disk Controller

BIC - For Buffer Interlace Controller.

2. Device Address of the unit.
3. Descriptive error information.

The following BIC and DC error messages are output by this unit program:

BIC Error Messages:

TIMEOUT <explanatory message>

or

<explanatory message>

where:

<explanatory message> is one or more of the following:

BUSY
SETUP ERROR
ABNORMAL BIC DEVICE STOP ERROR
BIC STOPPED - MAP MEMORY ERROR
FINAL ADDRESS WAS xxxxxx SHOULD BE xxxxxx
NOT AVAILABLE ERROR

DC Error Messages:

TIMEOUT <explanatory message>

or

<explanatory message>

where:

<explanatory message> is one or more of the following:

BUSY
INITIALIZE CONTROLLER ERROR
WRITE ERROR
READ ERROR
OUTPUT TRACK ADDRESS ERROR
OUTPUT SECTOR ADDRESS ERROR
SELECT SEEK MODE ERROR
SELECT SECTOR MODE ERROR
SEEK COMPLETE ERROR
RECALIBRATION ERROR
STATUS WORD <bit identification>
Bxx - <descriptive message>

(For correlation of xx and
<descriptive message> see
Table 11-1)

WRITE/READ DATA COMPARE ERROR
 BUFFER ADDRESS xxxxxx SIZE (OCTAL) xxxxxx
 DATA EXPECTED xxxxxx FOUND xxxxxx LOCATION xxxxxx

Table 11-1. Cartridge Disk Status Word Significance

| Bit | <descriptive message> |
|-----|-------------------------------|
| B15 | SYNC BYTE NOT FOUND |
| B14 | FORMAT ERROR |
| B13 | HEADER COMPARE AREA |
| B12 | UNIT NOT READY |
| B11 | WRITE PROTECT ERROR |
| B10 | END OF TRACK ERROR |
| B9 | READ CRC ERROR |
| B8 | CRC SEARCH ERROR |
| B7 | SELECTED UNIT TIMING ERROR |
| B6 | SELECTED UNIT MALFUNCTION |
| B5 | SELECTED UNIT ILLEGAL ADDRESS |
| B4 | SELECTED UNIT ILLEGAL SECTOR |
| B3 | UNIT 3 SEEK COMPLETE |
| B2 | UNIT 2 SEEK COMPLETE |
| B1 | UNIT 1 SEEK COMPLETE |
| B0 | UNIT 0 SEEK COMPLETE |

11.3 EXAMPLES

Example 1:

EX**

RU/D4

M4DF04.A DA=016 KY=01 PP=0100,0101,0102 BP=0,04

M4DF04.A DA=016 KY=01 PP=0100,0101,0102 BP=0103,01 TM=0:0:02
BIC 000020 ABNORMAL BIC DEVICE STOP READ ERROR
INITIAL ADDRESS 004000
FINAL ADDRESS WAS 004730 SHOULD BE 004740

Example 2:

EX**

RU/D4

M4DF04.A DA=016 KY=01 PP=0100,0101,0102 BP=0,0

M4DF04.A DA=016 KY=02 PP=0104,0105,0106 BP=0107,01 TM=0:0:05
BIC 000020 TIMEOUT NOT AVAILABLE ERROR

Example 3:

M4DF04.A DA=016 KY=015 PP=01025,01026,01027 BP=0703,03 TM=0:01:023
DC 001016 WRITE ERROR
STATUS WORD 000217
B7 - SELECTED UNIT TIMING ERROR
B3 - UNIT 3 SEEK COMPLETE
B2 - UNIT 2 SEEK COMPLETE
B1 - UNIT 1 SEEK COMPLETE
B0 - UNIT 0 SEEK COMPLETE

Example 4:

M4DF04.A DA=016 KY=01 PP=0762,0763,0764 BP=01042,02 TM=0:02:010
DC 001016 READ ERROR
STATUS WORD 000217
B7 - SELECTED UNIT TIMING ERROR
B3 - UNIT 3 SEEK COMPLETE
B2 - UNIT 2 SEEK COMPLETE
B1 - UNIT 1 SEEK COMPLETE
B0 - UNIT 0 SEEK COMPLETE

Example 5:

M4DF04.A DA=016 KY=05 PP=0525,0526,0527 BP=01523,04 TM=0:03:047

DC 000016 RECALIBRATION ERROR

STATUS WORD 004017

B11 - UNIT WRITE PROTECTED

B3 - UNIT 3 SEEK COMPLETE

B2 - UNIT 2 SEEK COMPLETE

B1 - UNIT 1 SEEK COMPLETE

B0 - UNIT 0 SEEK COMPLETE

Section 12

Disk Memory and Controller Unit Program

12.1 PURPOSE AND OPERATION

The Disk Memory and controller unit program (M4DC05) verifies the correct operation of 2822, 2823, 2824, F3092, and Model 70-7510 (types DC, DD, and DE) disks. Refer to Section 2 for the Software Model code corresponding to each disk.

The unit program operates as follows:

1. Data is written from memory onto the disk, alternating between the units on the controller.
2. This data is read back into memory.
3. Data written is compared with data read, to ensure a valid write/read operation.
4. Errors result in the generation of error messages.

12.2 ERROR MESSAGES

Error messages generated by the unit program are preceded by the EXEC message which identifies the unit program which is generating the messages. The error message consists of:

1. Identification
 - DC - For Disk controller
 - BTC - For Block Transfer Controller.
2. Device address of the unit.
3. Descriptive error information.

The following is a list of the descriptive error information generated by the unit program:

BTC Error Messages:

TIMEOUT <explanatory message>

or

<explanatory message>

where:

<explanatory message> is one or more of the following:

BUSY ERROR
SETUP ERROR
ABNORMAL BTC DEVICE STOP ERROR
BTC STOPPED - MAP MEMORY ERROR
FINAL ADDRESS xxxxxx

DC Error Messages:

TIMEOUT <explanatory message>

or

<explanatory message>

where:

<explanatory message> is one or more of the following:

BUSY ERROR
SELECT CYLINDER ERROR
RESTORE CYLINDER REGISTER ERROR
STOP TRANSFER AND INITIALIZE ERROR
OUTPUT SUW ERROR
WRITE ERROR
READ ERROR
WRITE/READ DATA COMPARE ERROR
BUFFER ADDRESS xxxxxx SIZE (OCTAL) xxxxxx
DATA EXPECTED xxxxxx FOUND xxxxxx LOCATION xxxxxx
STATUS WORD xxxxxx
Bxx - <descriptive message>

(For correlation of xx and <descriptive message>, refer to Table 12-1).

Table 12-1. Disk Memory Status Word Significance

| Bit | <descriptive message> |
|-----|---------------------------------|
| B15 | NOT USED |
| B14 | NOT USED |
| B13 | NOT USED |
| B12 | BAD TRACK FLAG SET |
| B11 | HEAD SELECT ERROR |
| B10 | UNIT NOT SELECTED |
| B9 | WRITE PROTECT ERROR |
| B8 | SEEK ERROR (INCOMPLETE) |
| B7 | DISK PACK UNSAFE |
| B6 | UNIT NOT ON LINE |
| B5 | END OF FILE (DATA LENGTH=0) |
| B4 | DATA ERROR (BAD CYCLIC CHECK) |
| B3 | SEARCH ERROR (BAD CYCLIC CHECK) |
| B2 | FORMAT ERROR (RECORD TOO LONG) |
| B1 | TRACK STOP (RECORD NOT FOUND) |
| B0 | TIMING ERROR |

12.3 EXAMPLES

EX**
RUS

M4DC05.A DA=015 KY=01 PP=0100,0101,0102 BP=0,0

M4DC05.A DA=015 KY=01 PP=0100,0101,0102 BP=0,0 TM=0:0:02
DC 000015 BUSY ERROR

M4DC05.A DA=015 KY=02 PP=0103,0104,0105 BP=0,0 TM=0:0:05
DC 000015 BUSY ERROR

Section 13

Direct Memory Access Disk Unit Program

13.1 PURPOSE AND OPERATION

The Direct Memory Access Disk unit program (M4DH06) verifies the correct operation of the 2825, 2826, 2842, 2843, and 8433 (Type DH) disks and associated controllers.

Operation is identical to that of the other disk unit programs. (Refer to Sections 11 and 12).

13.2 ERROR MESSAGES

All error messages output by this unit program are preceded by the EXEC message which identifies the unit program generating the message. The error message itself consists of the following:

1. Identification "DC" for disk controller.
2. Device address of the unit.
3. Descriptive error information.

The following descriptive error information is output by this unit program:

TIMEOUT <explanatory message>

or

<explanatory message>

where:

<explanatory message> is one or more of the following:

WAITING FOR DISK
LAST CCB EXECUTED-NONE
LAST CCB EXECUTED-POSITION
LAST CCB EXECUTED-WRITE
LAST CCB EXECUTED-READ
COMMAND STATUS NOT AVAILABLE ERROR
OUTPUT UAW ADDRESS ERROR

UNIT BUSY-ERROR
 SECONDARY STATUS INPUT ERROR
 WRITE/READ ERROR
 WRITE/READ DATA COMPARE ERROR
 WRONG UNIT STATUS ERROR
 DATA COMPARE ERROR
 PURGE CACHE ERROR
 PRIMARY STATUS WORD xxxxxx
 Bxx - <descriptive message>

(For correlation of xx and <descriptive message>, refer to Table 13-1).

Table 13-1. Direct Memory Access Disk Status Word Significance

| Status Word | <descriptive message> |
|-------------|--------------------------------------|
| B15 | UNIT BUSY |
| B14 | SECONDARY STATUS |
| B13 | END OF UNIT (NON-EXISTENT CYLINDER) |
| B12 | END OF FILE (HEADER STATUS BYTE SET) |
| B9/11 | MEMORY DATA BUS VERIFICATION ERROR |
| B8/11 | RATE ERROR |
| B11 | COMMAND REJECT |
| B8-10 | MEMORY TIMEOUT |
| B9-10 | CHAINING INTERRUPT (NOT USED) |
| B8/10 | OFF LINE |
| B10 | ATTENTION |
| B8-9 | UNIT DATA ERROR |
| B9 | MAP ERROR |
| B8 | MEMORY ERROR |

CYLINDER xxxxxx TRACK/SECTOR xxxxxx SECONDARY STATUS WORD xxxxxx

(Refer to Table 13-2 for significance of secondary status word).

BUFFER ADDRESS xxxxxx SIZE (OCTAL) xxxxxx
DATA EXPECTED xxxxxx FOUND xxxxxx LOCATION xxxxxx

Table 13-2. Direct Memory Access Disk Secondary Status Word Significance

| Secondary Status Word | <descriptive message> |
|-----------------------|----------------------------------|
| B9 | CORRECTABLE ECC ERROR (NOT USED) |
| B8 | WRITE PROTECT ERROR |
| B7 | SECTOR FLAGGED BAD |
| B6 | DATA SYNC FAIL |
| B5 | HEADER SYNC FAIL |
| B4 | SECTOR SEARCH ERROR |
| B3 | TRACK SELECT ERROR |
| B2 | CYLINDER SEEK ERROR |
| B1 | DATA CHECK ERROR |
| B0 | HEADER CRC ERROR |

13.3 EXAMPLES

Example 1:

EX**

RU

```
M4WC01.0 DA=074 KY=01 PP=0100 BP=0,02
M4ME03.0 DA=0 KY=02 PP=0101 BP=0,03
M4DH06.A DA=014 KY=03 PP=0102,0103,0104 BP=0,01
M4MT07.A DA=010 KY=04 PP=0105,0106,0107 BP=0,01
M4LP08.0 DA=035 KY=05 PP=0110,0111 BP=0,0
M4DM09.0 DA=070 KY=06 PP=0112,0113,077 BP=0,01
*****
M4DH06.A DA=014 KY=03 PP=0102,0103,0104 BP=0116,01 TM=0:0:02
DC 000014 LAST CCB EXECUTED-NONE
UNIT BUSY - ERROR
PRIMARY STATUS WORD 002400 STATUS WORD UNIT 000000
B0/10 - OFF LINE
```

Example 2:

```
M4DH06.A DA=014 KY=016 PP=0377,0400,0401 BP=0405,01 TM=0:03:034
DC 000014 LAST CCB EXECUTED-WRITE
SECONDARY STATUS INPUT ERROR
PRIMARY STATUS WORD 040000 STATUS WORD UNIT 000000
B14 - SECONDARY STATUS
CYLINDER 000002 TRACK/SECTOR 001000 SECONDARY STATUS WORD 000400
B0 - WRITE PROTECT ERROR
```

Section 14

Magnetic Tape Unit Program

14.1 PURPOSE AND OPERATION

The Magnetic Tape unit program (M4MT07) verifies the correct operation of F3088, F3089, F3093, and 0870 (Type MTA) magnetic tape units and associated controllers. Operation of the program is similar to that of the other disk unit programs. (Refer to Sections 11 and 12).

14.2 ERROR MESSAGES

All error messages generated by this unit program are preceded by the EXEC message which identifies the unit program generating the error message. The error message itself consists of the following:

1. Identification:

MT - For Magnetic Tape Controller.

BIC - For Buffer Interlace Controller.

2. Device Address of the unit.

3. Descriptive error information.

The following BIC error messages and MT error messages are output by this unit program:

BIC Error Messages:

TIMEOUT <explanatory message>

or

<explanatory message>

where:

<explanatory message> is one or more of the following:

BUSY ERROR
SETUP ERROR
ABNORMAL BIC DEVICE STOP ON WRITE ERROR
ABNORMAL BIC DEVICE STOP ON READ ERROR
BIC STOPPED - MAP MEMORY ERROR
INITIAL ADDRESS xxxxxx
FINAL ADDRESS WAS xxxxxx SHOULD BE xxxxxx
NOT AVAILABLE ERROR

MT Error messages:

TIMEOUT <explanatory message>

or

<explanatory message>

where:

<explanatory message> is one or more of the following:

REWINDING ERROR
NOT READY ERROR
REWIND ERROR
TRANSPORT SELECT ERROR
WRITE ERROR
WRITE PARITY ERROR
BACKSPACE ERROR
READY BINARY RECORD ERROR
READ FILE MARK INSTEAD OF DATA ERROR
READ PARITY ERROR
WRITE/READ DATA COMPARE ERROR
BUFFER ADDRESS xxxxxx SIZE (OCTAL) xxxxxx
DATA EXPECTED xxxxxx FOUND xxxxxx LOCATION xxxxxx
WRITE FILE MARK ERROR
SENSE END OF TAPE ERROR
STATUS WORD xxxxxx
Bxx - <descriptive message>

(For the correlation of xx and
<descriptive message>, refer
to Table 14-1).

Table 14-1. Magnetic Tape Status Word Significance

| Status Word | <descriptive message> |
|-------------|--------------------------------------|
| B15 | RATE ERROR |
| B14 | LRC ERROR/MULTIPLE DEAD TRACKS (PE) |
| B13 | DEAD TRACK ERROR |
| B12 | FORMATTER FATAL ERROR |
| B11 | CRC ERROR/FALSE POSTAMBLE ERROR (PE) |
| B10 | CHARACTER PARITY ERROR |
| B9 | HIGH DENSITY |
| B8 | NOT USED |
| B7 | REWIND |
| B6 | BOT |
| B5 | EOT |
| B4 | ODD LENGTH |
| B3 | FILE MARK |
| B2 | TRANSPORT NOT READY |
| B1 | WRITE ENABLE |
| B0 | TAPE ERROR |

14.3 EXAMPLES

M4MT07.A DA=010 KY=01 PP=0516,0517,0520 BP=01426,05 TM=0:01:035
BIC 000022 ABNORMAL BIC DEVICE STOP ON WRITE ERROR
INITIAL ADDRESS 061545
FINAL ADDRESS 065275 SHOULD BE 065345

M4MT07.A DA=010 KY=017 PP=0350,0351,0352 BP=0207,03 TM=0:03:029
NT 000010 NOT READY ERROR

Section 15

Line Printer Unit Program

15.1 PURPOSE AND OPERATION

The Line Printer unit program (M4LP08) verifies the correct operation of 2819, 2820, and 0786 line printers and associated controllers including the F3361 controller. The program is designed to operate one line printer.

This unit program tests the line printer by causing a particular print pattern to be output by the printer. All transfers of data from the computer to the printer buffer are accomplished by means of the Buffer Interlace Controller (BIC). The program outputs a "spiral" pattern on the printer. In this "spiral" pattern, each line begins with the second character in the preceding line. Each line has 96 characters from a 64-character set. These characters are output from successive pages of memory, until one of the following happens:

- 61 pages have been accessed
- An error occurs
- The program is aborted.

Only one type of interrupt, namely, "BIC COMPLETE" is processed by the program. Errors detected by the program result in the generation of error messages.

15.2 ERROR MESSAGES

All error messages generated by this unit program are preceded by the EXEC message which identifies the unit program generating the error message. The error message itself consists of the following:

1. Identification:
 - BIC - For the Buffer Interlace Controller.
 - LP - For the Line Printer.
2. Device address of the unit.
3. Descriptive error information.

The following descriptive error information is output by this unit program:

BIC Error Message:

```
BIC BUSY PRIOR TO ACTIVATION ERROR
ABNORMAL BIC DEVICE STOP ERROR
BIC STOPPED - MEMORY MAP ERROR
BIC HALT ADDRESS xxxxxx
BUFFER START ADDRESS xxxxxx
END ADDRESS xxxxxx
TIMEOUT AFTER BIC REQUEST
```

LP Error Messages:

```
PRINTER NOT READY ERROR
BUFFER NOT READY ERROR
PRINTER ERROR
TIMEOUT AFTER BIC CONNECT ERROR
```

15.3 EXAMPLES

Example 1:

EX**

RU

M4LP00.0 DA=035 KY=016 PP=0155,0156 BP=0,01

M4LP00.0 DA=035 KY=016 PP=0155,0156 BP=0,01 TM=0:0:02

LP PRINTER ERROR

Example 2:

M4LP00.0 DA=035 KY=05 PP=0110,0111 BP=0,0 TM=0:0:010

LP PRINTER NOT READY

Section 16

Data Communications Multiplexer Unit Program

16.1 PURPOSE AND OPERATION

The Data Communications Multiplexer (DCM) unit program (M4DM09) verifies the correct operation of the DCM (F3000) and its associated Line Adapters (LADs). The following LAD types are currently supported:

F3001-00 Asynchronous LAD RS232 Modem
F3001-01 Asynchronous LAD RS232
F3001-02 Asynchronous LAD, Current Loop
F3001-03 Asynchronous LAD, Relay I/O
F3001-04 Synchronous LAD RS232 Modem
F3001-05 Binary Synchronous Communication LAD
F3006-00 BSC LAD with Wide Band option.

The unit program runs from 1 to 4 lines consecutively. These lines are specified during the System Test Generation phase. Test connectors must be installed on the LADs to place them in a back-to-back configuration. The jumper configuration is described later in this section. It is identical to the jumper configuration for the MAINTAIN III diagnostic program. All of the lines must be of the same type. (Refer to Table 16-1 for a cross-reference of type and feature numbers of LADs.)

Table 16-1. LAD Type and Feature Number Cross-Reference

| Type | Feature Number |
|------|------------------------------|
| 0 | F3001-00 |
| 1 | F3001-01, F3001-02, F3001-03 |
| 2 | F3001-04 |
| 3 | F3001-05, F3006-00 |

For LAD types 0, 1, and 2, the program uses an ascending binary pattern of 256 bytes for test data. For the type 3 LAD, a pattern with zeros and ones alternating is used. The lines are run consecutively, and all errors are reported to the operator, via error messages.

16.2 ERROR MESSAGES

The following is a list of error messages generated by this unit program:

```
DCM ERROR ON LINE <octal line number>
LINE ERROR INTERRUPT
FRAMING ERROR
PARITY ERROR
STATUS CHANGE INTERRUPT
STATUS WORD IS xxxxxx
OUTPUT UNDERFLOW
CONTROL-LINE-IN
RING INDICATOR
CLEAR TO SEND
CARRIER ON
INTERLOCK ON
FORMAT ERROR
STATUS-LINE-IN
CONTROL CHARACTER DETECTED
OUTPUT COMPLETE
NO CONTROL INTERRUPT
DCM TIMEOUT ERROR ON LINE <octal line number>
DCM DATA COMPARE ERROR ON LINE <octal line number>
TOTAL NUMBER OF ERRORS IS <octal line number>
FIRST ERROR AT <octal address>
DCM SELECTION TIMEOUT ERROR
```

16.3 EXAMPLES

RU9

```
M4DM09.0 DA=070 KY=01 PP=0100,0101,077 BP=0,01
```

```
*****
```

```
M4DM09.0 DA=070 KY=02 PP=0102,0103,077 BP=0,01 TM=0:0:032
```

```
DCM SELECTION TIMEOUT ERROR
```

```
M4DM09.0 DA=070 KY=03 PP=0104,0105,077 BP=0,01 TM=0:01:05
```

```
DCM SELECTION TIMEOUT ERROR
```

```
EX**
```

SYSTEM TEST CONFIGURATION GUIDE

SALES ORDER NUMBER _____

DATE PREPARED _____

V70-SERIES SYSTEM TEST GENERATION

SELECT CONFIGURATION QUESTIONS

NOTE:ACCEPTABLE REPLIES ARE IN ()
DEFAULT=0 OR VALUE AFTER ;
END EACH LINE WITH CARRIAGE RETURN OR PERIOD
IF REPLY IS OMITTED,DEFAULT VALUE IS USED
A=ALL,H=HELP,N=NO OR NONE,Y=YES
ENTER SPACE IF OUTPUT PAUSES
* INDICATES PRESELECTED SYSTEM QUESTION
LEADING 0=OCTAL,MAY BE OMITTED

CNFG?(H,A,1,2,...;A) H

- *1.SYS TEST CPU,SYS GEN CPU?(H,0-4,0-4)
- *2.START,END ADR(K) OF EACH MEM BLOCK?(S1,E1 S2,E2 ...;0,256)
- *3.MEMORY PARITY INTERRUPT ADR?(010-077;060)
- *4.SYSTEM TEST OPERATOR'S CONSOLE
- *5.SYSTEM TEST LIST DEVICE?(0=TTY/CRT,1=LPR,2=BOTH;2)
- *6.SYSTEM GEN INPUT DEVICE?(0=PT,1=MT,2=DS,3=CR;1)
- *7.SYSTEM GEN OUTPUT DEVICE?(0=PT,1=MT,2=DS,3=CP;1)
- 8.WCS-UP1,15?(Y;N)
- 9.FLOATING POINT PROCESSOR-UP2,15?(Y;N)
- 10.MEMORY TEST-UP3?(Y;N)
- 11.HOW MANY DISC CONTROLLERS-UP4,5,6,16?(0-4)
- 12.HOW MANY MAG TAPE CONTROLLERS-UP7?(0-4)
- 13.HOW MANY LINE PRINTERS-UP8?(0-4)
- 14.HOW MANY DCM'S-UP9?(0-4)
- 15.HOW MANY TELETYPES-UP10?(0-7)
- 16.HOW MANY CRT'S-UP11?(0-7)
- 17.CARD READER-UP12?(Y;N)
- 18.CARD PUNCH-UP13?(Y;N)
- 19.PAPER TAPE-UP14?(Y;N)

CNFG?(H,A,1,2,...;A) A

*1.SYS TEST CPU,SYS GEN CPU?(H,0-4,0-4) 1,1. ①

- (0=V77-800)
- (1=V77-600)
- (2=V77-400)
- (3=V76)
- (4=V75)

*2.START,END ADR(K) OF EACH MEM BLOCK?(S1,E1 S2,E2 ...;0,256) 0,256 ②

*3.MEMORY PARITY INTERRUPT ADR?(010-077;060) 060. ③

Appendix A

System Test Generation Tutorial Guide

A.1 PURPOSE

The purpose of this section is to guide the user in the preparation of system configuration dialog responses.

All input responses should be prepared prior to performing the system test configuration. The System Memo shipped with the system is the primary source of information required by the operator to accomplish this. In this section, a "System Test Configuration Guide" filled in with values applicable to a given system configuration is used for explanatory purposes. This sample system configuration is as follows:

- V77-600 CPU
- 256K words of memory
- Keyboard-display terminal (CRT): System List device
- Teletypewriter: System List device
- Central complex options:
 - Writable Control Store (WCS)
 - Floating-Point Processor (FPP)
- Peripheral device options:
 - Two disk controllers
 - Disk Type DC
 - Disk Type DF
 - Three magnetic tape controllers (Model 0)
 - Line Printer
 - Data Communications Multiplexer (DCM)
 - Buffer Interlace Controller (BIC)
 - Block Transfer Controller (BTC)

The System Test Configuration Guide is followed by a discussion which is keyed to the Guide. The discussion presents pointers to sections in the System Memo where the operator can obtain information to correctly respond to the configuration questions. Sections of the Guide which are not applicable to the sample system configuration are framed. Excerpts from the System Memo associated with the sample system are included.

A listing obtained during an actual System Test Generation on the sample system configuration follows the System Test Configuration Guide. The detailed discussion of configuration questions in Section 4 is keyed to this listing.

NOTES

The number keys in this listing do not correspond with the number keys in the System Test Configuration Guide.

A.2 SYSTEM TEST CONFIGURATION GUIDE

The System Test Configuration Guide duplicates the sequence and content of all the configuration questions output during System Test Generation. Since all the configuration questions are numbered, corresponding sections in the System Test Generation listing can be easily identified.

NOTES

Question 11 in the System Test Configuration Guide does not list those questions that are asked depending upon the type of disk specified by the operator. For information about these questions, refer to Section 4. Determine the model type of the disk in your system; determine the questions you will be asked (by referring to Section 4), and prepare your responses accordingly.

16.4 JUMPER CONFIGURATIONS

Test connectors (Part Number 57 A 0036-000) are required for testing. The jumper configuration for each type of LAD is described in this section.

LAD type F3001-00

Pin 29 to Pin 35
Pin 35 to Pin 33
Pin 29 to Pin 25
Pin 27 to Pin 37
Pin 31 to Pin 23
Pin 21 to Pin 18
Pin 1 to Pin 13
Pin 13 to Pin 11
Pin 1 to Pin 7
Pin 3 to Pin 19
Pin 5 to Pin 9
Pin 6 to Pin 8

LAD type F3001-01

Pin 2 to Pin 22
Pin 4 to Pin 24

LAD type F3001-02

Pin 3 to Pin 11
Pin 29 to Pin 39
Pin 37 to Pin 40
Pin 27 to Pin 34 to Pin 6
Pin 12 to Pin 7

NOTE

R27, R30, R39, R42, R51, R54, R63, and R66 must be shorted across when using this test connector.

LAD type F3001-03

Pin 1 to Pin 25 to Pin 44
Pin 10 to Pin 5
Pin 31 to Pin 38
Pin 7 to Pin 27 to Pin 37

NOTE

R33, R29, R45, R41, R53, R57, R65, and R69 must be shorted across, and jumpers placed where R22 and R46 are mounted in order to use this test connector.

LAD type F3001-04

Same jumper configuration as for the type F3001-00 with the following additions:

Pin 38 to Pin 39
Pin 39 to Pin 41
Pin 38 to Pin 17
Pin 17 to Pin 15

LAD type F3001-05

Pin 29 to Pin 33 to Pin 25
Pin 27 to Pin 37
Pin 31 to Pin 23
Pin 21 to Pin 18
Pin 17 to Pin 35
Pin 38 to Pin 39
Pin 10 to Pin 41

LAD type F3006-00

Pin 5 to Pin 23
Pin 3 to Pin 37
Pin 7 to Pin 18 to Pin 33 to Pin 25
Pin 17 to Pin 35
Pin 44 to Pin 39 to Pin 41

*4. SYSTEM TEST OPERATOR'S CONSOLE

DEVICE ADDRESS?(0-076; 01) 01.

READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) 0154.

WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) 0156.

*5. SYSTEM TEST LIST DEVICE?(0=TTY/CRT,1=LPR,2=BOTH;2) 2.

PRINTER X:

DEVICE ADDRESS?(0-076; 35) 035.

TTY OR CRT X:

DEVICE ADDRESS?(0-076; 01) 01

(note: if DA=DA of *4, next two questions are skipped)

READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) 0154.

WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) 0156.

*6. SYSTEM GEN INPUT DEVICE?(0=PT,1=MT,2=DS,3=CR;1) 1.

(default device addresses are: PT=37,MT=10,DS=16,CR=30)

DEVICE ADDRESS?(0-076; xx) 010.

(if PT the next two questions are skipped)

BIC/BTC EVEN DEVICE ADDRESS?(0-076) 024.

UNIT NO?(0-3) 0.

*7. SYSTEM GEN OUTPUT DEVICE?(0=PT,1=MT,2=DS,3=CP;1) 1.

(default device addresses are: PT=37,MT=10,DS=16,CP=31)

DEVICE ADDRESS?(0-076; xx) 011.

(if PT the next two questions are skipped)

BIC/BTC EVEN DEVICE ADDRESS?(0-076) 022.

UNIT NO?(0-3) 0.

8. WCS-UP1,15?(Y;N) Y.

EVEN DEVICE ADDRESS?(0-076; 74) 074.

9.FLOATING POINT PROCESSOR-UP2,15?(Y;N) Y.

10.MEMORY TEST-UP3?(Y;N) Y.

11.NOW MANY DISC CONTROLLERS-UP4,5,6,16?(0-4) 2.

DISC n: (for models 0 and 4)

MODEL?(H,0-6;5) 4.

- (0=TYPE DB)
- (1=TYPE DC)
- (2=TYPE DD)
- (3=TYPE DE)
- (4=TYPE DF)
- (5=TYPE DH)
- (6=TYPE DJ)

DEVICE ADDRESS?(0-076; 16) 013.

BIC/BTC EVEN DEVICE ADDRESS?(0-076) 020.

BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) 0100.

EACH SEEK COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) 0124.

WHICH TRACK(T=0-0312/0625),SECTOR(S=0-2)?(T,S) 0,0.

WHICH PLATTERS(P=0-1) & UNITS(U=0-3)?(PU,...) 00

DISC n: (for models 1, 2, and 3)

MODEL?(H,0-6;5) 1

DEVICE ADDRESS?(0-076; 15) 015.

BIC/BTC EVEN DEVICE ADDRESS?(0-076) 070.

BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) 0110.

SEEK COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) 0116.

WHICH HEAD?(0-023) 0.

WHICH UNITS?(0-3/7) 0.

DISC n: (for model 5)

MODEL?(H,0-6;5) _____

EVEN DEVICE ADDRESS?(0-076; 14) _____

COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

WHICH CYLINDER(C=0-0632/01466), HEAD(H=0-4/022)?(C,H) _____

WHICH UNITS?(0-3/7) _____

DISC n: _____ (for model 6)

MODEL?(H,0-6;5) _____

DEVICE ADDRESS?(0-076; 12) _____

BIC/BTC EVEN DEVICE ADDRESS?(0-076) _____

BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

CONTROLLER COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

WHICH TRACK(T=0-0114), SECTOR(S=0-033)?(T,S) _____

WHICH UNITS?(0-3/7) _____

12. HOW MANY MAG TAPE CONTROLLERS-UP??(0-4) 3.

MAG TAPE 1:
 MODEL?(H,0,1) 0.

(0=NO STATUS WORD)
 (1=USES STATUS WORD)

DEVICE ADDRESS?(0-076; 10) 010.

BIC/BTC EVEN DEVICE ADDRESS?(0-076) 024.

BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) 0104.

MOTION COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) 0132.

WHICH UNITS?(0-3/7) 0.

MAG TAPE 2:
 MODEL?(H,0,1) 0.

DEVICE ADDRESS?(0-076; 11) 011.

BIC/BTC EVEN DEVICE ADDRESS?(0-076) 022.

BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) 0102.
MOTION COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) 0130.
WHICH UNITS?(0-3/7) 0.

MAG TAPE 3:
MODEL?(H,0,1) 0.
DEVICE ADDRESS?(0-076; 12) 012.
BIC/BTC EVEN DEVICE ADDRESS?(0-076) 026.
BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) 0106.
MOTION COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) 0134.
WHICH UNITS?(0-3/7) 0.

MAG TAPE 4:
MODEL?(H,0,1) _____
DEVICE ADDRESS?(0-076; 13) _____
BIC/BTC EVEN DEVICE ADDRESS?(0-076) _____
BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____
MOTION COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____
WHICH UNITS?(0-3/7) _____

13. HOW MANY LINE PRINTERS-UP8?(0-4) 1.

PRINTER 1:
DEVICE ADDRESS?(0-076; 35) 035.
BIC/BTC EVEN DEVICE ADDRESS?(0-076) 026.
BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) 0106.

PRINTER 2:
DEVICE ADDRESS?(0-076; 35) _____
BIC/BTC EVEN DEVICE ADDRESS?(0-076) _____
BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

PRINTER 3:
DEVICE ADDRESS?(0-076; 35) _____
BIC/BTC EVEN DEVICE ADDRESS?(0-076) _____
BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____
PRINTER 4:
DEVICE ADDRESS?(0-076; 35) _____
BIC/BTC EVEN DEVICE ADDRESS?(0-076) _____
BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

14. HOW MANY DCM'S-UP9?(0-4) 1. — (7)

DCM 1:
LINE ADAPTER TYPE?(H,0-3) 0.

(0=ASYNCHRONOUS DIRECT CONNECT)
(1= " " DATA SET)
(2=SYNCHRONOUS)
(3=BI-SYNCHRONOUS)

DEVICE ADDRESS?(0-076; 70) 066.
DCM INTERRUPT ADDRESS ORIGIN?(0100-0260; 0260) 0260.
BITS PER BYTE?(5-010; 010) 010.
LCB MEMORY PAGE?(070-077; 075) 075.
WHICH LINES (MAX OF 4)?(00-077) 0,1.

DCM 2:
LINE ADAPTER TYPE?(H,0-3) _____
DEVICE ADDRESS?(0-076; 71) _____
DCM INTERRUPT ADDRESS ORIGIN?(0100-0260; 0260) _____
BITS PER BYTE?(5-010; 010) _____
LCB MEMORY PAGE?(070-077; 075) _____
WHICH LINES (MAX OF 4)?(00-077) _____
DCM 3:

LINE ADAPTER TYPE?(H,0-3) _____
 DEVICE ADDRESS?(0-076; 72) _____
 DCM INTERRUPT ADDRESS ORIGIN?(0100-0260;0260) _____
 BITS PER BYTE?(5-010;010) _____
 LCB MEMORY PAGE?(070-077;075) _____
 WHICH LINES(MAX OF 4)?(00-077) _____
 DCM 4:
 LINE ADAPTER TYPE?(H,0-3) _____
 DEVICE ADDRESS?(0-076; 73) _____
 DCM INTERRUPT ADDRESS ORIGIN?(0100-0260;0260) _____
 BITS PER BYTE?(5-010;010) _____
 LCB MEMORY PAGE?(070-077;075) _____
 WHICH LINES(MAX OF 4)?(00-077) _____

15. HOW MANY TELETYPES-UP10?(0-7) 1.

TTY 1:
 DEVICE ADDRESS?(0-076; 1) 01.

(note: if DA=DA of *4, next two questions are skipped)

READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____
 WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

TTY 2:
 DEVICE ADDRESS?(0-076; 2) _____
 READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____
 WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____
 TTY 3:
 DEVICE ADDRESS?(0-076; 3) _____
 READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

TTY 4:
 DEVICE ADDRESS?(0-076; 4) _____

READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

TTY 5:
 DEVICE ADDRESS?(0-076; 5) _____

READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

TTY 6:
 DEVICE ADDRESS?(0-076; 6) _____

READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

TTY 7:
 DEVICE ADDRESS?(0-076; 7) _____

READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

16. HOW MANY CRT'S-UP11?(0-7) 2.

CRT 1:
 DEVICE ADDRESS?(0-076; 1) 01.

(note: if DA=DA of *4, next two questions are skipped)

READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

CRT 2:
 DEVICE ADDRESS?(0-076; 2) 02.

READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) 0140.

WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) 0142.

CRT 3:
 DEVICE ADDRESS?(0-076; 3) _____
 READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____
 WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

CRT 4:
 DEVICE ADDRESS?(0-076; 4) _____
 READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____
 WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

CRT 5:
 DEVICE ADDRESS?(0-076; 5) _____
 READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____
 WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

CRT 6:
 DEVICE ADDRESS?(0-076; 6) _____
 READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____
 WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

CRT 7:
 DEVICE ADDRESS?(0-076; 7) _____
 READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____
 WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

17. CARD READER-UP12?(Y;N) Y.
 DEVICE ADDRESS?(0-076; 30) 030.
 BIC/BTC EVEN DEVICE ADDRESS?(0-076) 022.
 BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) 0102.

18. CARD PUNCH-UP13?(Y;N) Y.
 DEVICE ADDRESS?(0-076; 31) 031.
 BIC/BTC EVEN DEVICE ADDRESS?(0-076) 022.

BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS? (0100-0276) 0102.

19. PAPER TAPE-UP14? (Y;N) Y.

DEVICE ADDRESS? (0-076; 37) 037.

CHAR READY COMPLETE EVEN INTERRUPT ADDRESS? (0100-0276) 0144.

- ① Information for configuration question number 1 can be obtained from the EQUIPMENT SUMMARY section of the System Memo.
- ② Refer to the SYSTEM BLOCK DIAGRAM section of the System Memo to determine start and end of addresses of each memory block.
- ③ Refer to the OPTION ASSIGNMENTS section (5.2) of the System Memo to determine memory parity interrupt address.
- ④ Information for questions 4, 5, and 6 can be obtained from the PIM/BIC ASSIGNMENTS section (4.1) and the DEVICE ADDRESS ASSIGNMENTS section (4.2) in the System Memo. Figure A-1 consists of three pages from the System Memo associated with the specific system being discussed here. These three pages show the PIM/BIC and device assignments for this particular system. From page 3 of the figure, it can be seen that the device address of the operator's console is 01, and that of the printer (one of the list devices) is 035. From page 1 of the figure it can be seen that the operator's console (TYOB) is associated with PIM #3. Note that the interrupt addresses for PIM #3 start from 140. To calculate the READ READY COMPLETE EVEN INTERRUPT ADDRESS associated with the operator's console, apply the following formula using octal (base 8) arithmetic:

Starting Interrupt Address + (2 x priority level)

For the operator's console, the starting interrupt address is 140. The priority level (for READ) is 6. Therefore, the READ READY COMPLETE EVEN INTERRUPT ADDRESS is:

$$(140)_8 + (2 \times 6)_8 = 154.$$

Using the same formula, the WRITE READY COMPLETE EVEN INTERRUPT ADDRESS for the operator's console can be calculated to be: 156.

- ⑦ For configuration question number 14, obtain information from the DCM section of the System Memo.

For configuration question number 5, if the device address specified for the TTY or CRT List device is the same as the System Test operator's console device address, the operator will not be asked for the READ and WRITE READY COMPLETE EVEN INTERRUPT ADDRESSES.

The operator needs to refer to only these three pages of the System Memo (Figure A-1), to obtain most of the information required to answer the rest of the configuration questions. Correlation of the sample System Test Configuration Guide and Figure A-1 will itself clearly illustrate how this is done.

4.1 PIM/BIC Assignments

4.1.1 PIM Priority Assignments

PIM #1

Device Address 40
Interrupt Address 100-117
Priority Levels:

| | |
|----|-------------|
| 0. | BIC #1 Comp |
| 1. | BIC #2 |
| 2. | BIC #3 |
| 3. | BIC #4 |
| 4. | BIC #1 |
| 5. | |
| 6. | BIC #5 Comp |
| 7. | 7500 Disk |

PIM #2

Device Address 41
Interrupt Address 120-137
Priority Levels:

| | |
|----|----------------------|
| 0. | |
| 1. | |
| 2. | HAWK Seek Complete |
| 3. | |
| 4. | MT1A Motion Complete |
| 5. | MTOA Motion Complete |
| 6. | MT2A Motion Complete |
| 7. | PT CHAR READY |

PIM #3

Device Address 42
Interrupt Address 140-157
Priority Levels:

| | |
|----|------------|
| 0. | |
| 1. | |
| 2. | |
| 3. | |
| 4. | CTOB READ |
| 5. | CTOB WRITE |
| 6. | TYOB READ |
| 7. | TYOB WRITE |

PIM #4

Device Address _____
Interrupt Address _____
Priority Levels:

| | |
|----|--|
| 0. | |
| 1. | |
| 2. | |
| 3. | |
| 4. | |
| 5. | |
| 6. | |
| 7. | |

PIM #5

Device Address _____
Interrupt Address _____
Priority Levels:

| | |
|----|--|
| 0. | |
| 1. | |
| 2. | |
| 3. | |
| 4. | |
| 5. | |
| 6. | |
| 7. | |

PIM #6

Device Address _____
Interrupt Address _____
Priority Levels:

| | |
|----|--|
| 0. | |
| 1. | |
| 2. | |
| 3. | |
| 4. | |
| 5. | |
| 6. | |
| 7. | |

Figure A-1. Excerpts From A System Memo (Part 1 of 3)

4.1.2

BIC Assignments

BIC #1

Device Address 20, 21
Controller Assignments:

- 1. HAWK DA=13
- 2. _____
- 3. _____
- 4. _____
- 5. _____

BIC #2

Device Address 22, 23
Controller Assignments:

- 1. MT1A DA=11
- 2. CARD READER DA=30
- 3. _____
- 4. _____
- 5. _____

BIC #3

Device Address 24, 25
Controller Assignments:

- 1. MTOA DA=10
- 2. PTS DA=37
- 3. _____
- 4. _____
- 5. _____

BIC #4

Device Address 26, 27
Controller Assignments:

- 1. MT2A DA=12
- 2. LPOB DA=35
- 3. STATOS DA=36
- 4. _____
- 5. _____

BIC #5

Device Address 50
Controller Assignments:

- 1. CPOA DA=31
- 2. _____
- 3. _____
- 4. _____
- 5. _____

BIC #6

Device Address 70, 71
Controller Assignments:

- 1. 7500 Disk DA=15
- 2. _____
- 3. _____
- 4. _____
- 5. _____

Figure A-1. Excerpts From A System Memo (Part 2 of 3)

4.2 Device Address Assignments

| DA | DEVICE | DA | DEVICE |
|-------|-------------------------|-------|----------------------|
| * 01 | Operator Console TYOB | * 040 | PIM #1 IA = 100-117 |
| 02 | CTOB | * 041 | PIM #2 IA = 120-137 |
| 03 | | * 042 | PIM #3 IA = 140-157 |
| 04 | | * 043 | PIM #4 IA = 160-177 |
| 05 | | * 044 | All PIMs, ECC Memory |
| 06 | | * 045 | MP |
| 07 | | * 046 | MAP |
| * 010 | Mag Tape | * 047 | RTC |
| 011 | Mag Tape | 050 | BIC #5 |
| 012 | Mag Tape | 051 | |
| 013 | Hawk Disc | 052 | |
| * 014 | Direct Memory Disc | 053 | |
| * 015 | Direct Memory Disc | 054 | |
| * 016 | Cartridge Disc | 055 | |
| 017 | | 056 | |
| * 020 | BIC #1 | 057 | |
| 021 | | 060 | |
| * 022 | BIC #2 | 061 | |
| 023 | | 062 | |
| * 024 | BIC #3 | 063 | |
| 025 | | 064 | |
| * 026 | BIC #4 | 065 | |
| 027 | | 066 | DCM |
| * 030 | Card Reader | 067 | |
| 031 | Card Punch | * 070 | BTC #1 |
| 032 | | 071 | |
| 033 | | 072 | |
| 034 | | 073 | |
| * 035 | Line Printer LPOB | * 074 | WCS |
| 036 | STATOS | * 075 | |
| * 037 | Paper Tape Reader/Punch | 076 | |
| | | * 077 | CONSOLE |

***Standard Device Addresses**

Figure A-1. Excerpts From A System Memo (Part 3 of 3)

A.3 SAMPLE SYSTEM TEST GENERATION LISTING

V70-SERIES SYSTEM TEST GENERATION

SELECT CONFIGURATION QUESTIONS

NOTE: ACCEPTABLE REPLIES ARE IN ()

DEFAULT=0 OR VALUE AFTER ;

END EACH LINE WITH CARRIAGE RETURN OR PERIOD

IF REPLY IS OMITTED, DEFAULT VALUE IS USED

A=ALL, H=HELP, N=NO OR NONE, Y=YES

ENTER SPACE IF OUTPUT PAUSES

* INDICATES PRESELECTED SYSTEM QUESTION

LEADING 0=OCTAL, MAY BE OMITTED

CNFG?(H,A,1,2,...;A)H

- *1.SYS TEST CPU,SYS GEN CPU?(H,0-4,0-4)
- *2.START,END ADR(K) OF EACH MEM BLOCK?(S1,E1 S2,E2 ...;0,256)
- *3.MEMORY PARITY INTERRUPT ADR?(010-077;060)
- *4.SYSTEM TEST OPERATOR'S CONSOLE
- *5.SYSTEM TEST LIST DEVICE?(0=TTY/CRT,1=LPR,2=BOTH;2)
- *6.SYSTEM GEN INPUT DEVICE?(0=PT,1=MT,2=DS,3=CR;1)
- *7.SYSTEM GEN OUTPUT DEVICE?(0=PT,1=MT,2=DS,3=CP;1)
- 8.UCS-UP1,15?(Y;N)
- 9.FLOATING POINT PROCESSOR-UP2,15?(Y;N)
- 10.MEMORY TEST-UP3?(Y;N)
- 11.HOW MANY DISC CONTROLLERS-UP4,5,6,16?(0-4)
- 12.HOW MANY MAG TAPE CONTROLLERS-UP7?(0-4)
- 13.HOW MANY LINE PRINTERS-UP8?(0-4)
- 14.HOW MANY DCM'S-UP9?(0-4)
- 15.HOW MANY TELETYPES-UP10?(0-7)
- 16.HOW MANY CRT'S-UP11?(0-7)
- 17.CARD READER-UP12?(Y;N)
- 18.CARD PUNCH-UP13?(Y;N)
- 19.PAPER TAPE-UP14?(Y;N)

CNFG?(H,A,1,2,...;A)A

- *1.SYS TEST CPU,SYS GEN CPU?(H,0-4,0-4) 1,1
- *2.START,END ADR(K) OF EACH MEM BLOCK?(S1,E1 S2,E2 ...;0,256)
- *3.MEMORY PARITY INTERRUPT ADR?(010-077;060)
- *4.SYSTEM TEST OPERATOR'S CONSOLE
DEVICE ADDRESS?(0-076; 01)
READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)154
WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)156
- *5.SYSTEM TEST LIST DEVICE?(0=TTY/CRT,1=LPR,2=BOTH;2)
PRINTER X:
DEVICE ADDRESS?(0-076; 35)
TTY OR CRT X:
DEVICE ADDRESS?(0-076; 01)
- *6.SYSTEM GEN INPUT DEVICE?(0=PT,1=MT,2=DS,3=CR;1)
DEVICE ADDRESS?(0-076; 10)
BIC/BTC EVEN DEVICE ADDRESS?(0-076)24
UNIT NO?(0-3)

```

*7.SYSTEM GEN OUTPUT DEVICE?(0=PT,1=MT,2=DS,3=CF;1) .
  DEVICE ADDRESS?(0-076; 10) 11.
  BIC/BTC EVEN DEVICE ADDRESS?(0-076)22.
  UNIT NO?(0-3).
  8.WCS-UP1,15?(Y;N) Y. 10
  EVEN DEVICE ADDRESS?(0-076; 74)
  9.FLOATING POINT PROCESSOR-UP2,15?(Y;N)Y. 11
10.MEMORY TEST-UP3?(Y;N)Y. 12
11.HOW MANY DISC CONTROLLERS-UP4,5,6,16?(0-4) 2.
DISC 1:
  MODEL?(H,0-6;5)4.
  DEVICE ADDRESS?(0-076; 16) 13.
  BIC/BTC EVEN DEVICE ADDRESS?(0-076)20.
  BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)100.
  EACH SEEK COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)124.
  WHICH TRACK(T=0-0312/0625),SECTOR(S=0-2)?(T,S)
  WHICH PLATTERS(P=0-1) & UNITS(U=0-3)?(PU,...)
DISC 2:
  MODEL?(H,0-6;5)1.
  DEVICE ADDRESS?(0-076; 15)
  BIC/BTC EVEN DEVICE ADDRESS?(0-076)70.
  BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)110.
  SEEK COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)116.
  WHICH HEAD?(0-023)
  WHICH UNITS?(0-3/7)
12.HOW MANY MAG TAPE CONTROLLERS-UP7?(0-4)3.
MAG TAPE 1:
  MODEL?(H,0,1)
  DEVICE ADDRESS?(0-076; 10)
  BIC/BTC EVEN DEVICE ADDRESS?(0-076)24.
  BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)104.
  MOTION COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)132.
  WHICH UNITS?(0-3/7)
MAG TAPE 2:
  MODEL?(H,0,1)
  DEVICE ADDRESS?(0-076; 11)
  BIC/BTC EVEN DEVICE ADDRESS?(0-076)22.
  BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)102.
  MOTION COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)130.
  WHICH UNITS?(0-3/7)
MAG TAPE 3:
  MODEL?(H,0,1)
  DEVICE ADDRESS?(0-076; 12)
  BIC/BTC EVEN DEVICE ADDRESS?(0-076)26.
  BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)106.
  MOTION COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)134.
  WHICH UNITS?(0-3/7)
13.HOW MANY LINE PRINTERS-UP8?(0-4) 1.
  PRINTER 1:
  DEVICE ADDRESS?(0-076; 35)
  BIC/BTC EVEN DEVICE ADDRESS?(0-076)26.
  BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)106.

```

```

14.HOW MANY DCM'S-UP9?(0-4) 1
  DCM 1:
  LINE ADAPTER TYPE?(H,0-3)
  DEVICE ADDRESS?(0-076; 70) 66
  DCM INTERRUPT ADDRESS ORIGIN?(0100-0260;0260)
  BITS PER BYTE?(5-010;010)
  LCB MEMORY PAGE?(070-077;075)
  WHICH LINES(MAX OF 4)?(00-077) 0,1
15.HOW MANY TELETYPES-UP10?(0-7)1
  TTY 1:
  DEVICE ADDRESS?(0-076; 01)
16.HOW MANY CRT'S-UP11?(0-7)2
  CRT 1:
  DEVICE ADDRESS?(0-076; 01)
  CRT 2:
  DEVICE ADDRESS?(0-076; 01) 2
  READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)140.
  WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)142.
17.CARD READER-UP12?(Y;N) Y.
  DEVICE ADDRESS?(0-076; 30)
  BIC/BTC EVEN DEVICE ADDRESS?(0-076)22.
  BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)102.
18.CARD PUNCH-UP13?(Y;N)Y.
  DEVICE ADDRESS?(0-076; 31)
  BIC/BTC EVEN DEVICE ADDRESS?(0-076)22.
  BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)102.
19.PAPER TAPE-UP14?(Y;N)Y.
  DEVICE ADDRESS?(0-076; 37)
  CHAR READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276)144.
END OF LIBRARY
UNIT PROGRAM 10 NOT FOUND
UNIT PROGRAM 11 NOT FOUND
UNIT PROGRAM 11 NOT FOUND
UNIT PROGRAM 12 NOT FOUND
UNIT PROGRAM 13 NOT FOUND
SYSTEM CONFIGURATION DONE
PREPARE OUTPUT DEVICE-ENTER SPACE WHEN READY

```

16

18

19

20

21

22

23

A.4 BLANK SYSTEM TEST CONFIGURATION GUIDE

Use the blank System Test Configuration Guide provided in this section to prepare system configuration dialog responses.

SYSTEM TEST CONFIGURATION GUIDE

SALES ORDER NUMBER _____

DATE PREPARED _____

V70-SERIES SYSTEM TEST GENERATION

SELECT CONFIGURATION QUESTIONS

NOTE:ACCEPTABLE REPLIES ARE IN ()

DEFAULT=0 OR VALUE AFTER ;

END EACH LINE WITH CARRIAGE RETURN OR PERIOD

IF REPLY IS OMITTED,DEFAULT VALUE IS USED

A=ALL,H=HELP,N=NO OR NONE,Y=YES

ENTER SPACE IF OUTPUT PAUSES

* INDICATES PRESELECTED SYSTEM QUESTION

LEADING 0=OCTAL,MAY BE OMITTED

CNFG?(H,A,1,2,...;A) _____

- *1.SYS TEST CPU,SYS GEN CPU?(H,0-4,0-4)
- *2.START,END ADR(K) OF EACH MEM BLOCK?(S1,E1 S2,E2 ...;0,256)
- *3.MEMORY PARITY INTERRUPT ADR?(010-077;060)
- *4.SYSTEM TEST OPERATOR'S CONSOLE
- *5.SYSTEM TEST LIST DEVICE?(0=TTY/CRT,1=LPR,2=BOTH;2)
- *6.SYSTEM GEN INPUT DEVICE?(0=PT,1=MT,2=DS,3=CR;1)
- *7.SYSTEM GEN OUTPUT DEVICE?(0=PT,1=MT,2=DS,3=CP;1)
- 8.WCS-UP1,15?(Y;N)
- 9.FLOATING POINT PROCESSOR-UP2,15?(Y;N)
- 10.MEMORY TEST-UP3?(Y;N)
- 11.HOW MANY DISC CONTROLLERS-UP4,5,6,16?(0-4)
- 12.HOW MANY MAG TAPE CONTROLLERS-UP7?(0-4)
- 13.HOW MANY LINE PRINTERS-UP8?(0-4)
- 14.HOW MANY DCM'S-UP9?(0-4)
- 15.HOW MANY TELETYPES-UP10?(0-7)
- 16.HOW MANY CRT'S-UP11?(0-7)
- 17.CARD READER-UP12?(Y;N)
- 18.CARD PUNCH-UP13?(Y;N)
- 19.PAPER TAPE-UP14?(Y;N)

CNFG?(H,A,1,2,...;A) _____

*1.SYS TEST CPU,SYS GEN CPU?(H,0-4,0-4) _____

- (0=V77-800)
- (1=V77-600)
- (2=V77-400)
- (3=V76)
- (4=V75)

*2.START,END ADR(K) OF EACH MEM BLOCK?(S1,E1 S2,E2 ...;0,256) _____

*3.MEMORY PARITY INTERRUPT ADR?(010-077;060) _____

*4. SYSTEM TEST OPERATOR'S CONSOLE

DEVICE ADDRESS?(0-076; 01) _____

READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

*5. SYSTEM TEST LIST DEVICE?(0=TTY/CRT,1=LPR,2=BOTH;2) _____

PRINTER X:

DEVICE ADDRESS?(0-076; 35) _____

TTY OR CRT X:

DEVICE ADDRESS?(0-076; 01) _____

(note: if DA=DA of *4, next two questions are skipped)

READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

*6. SYSTEM GEN INPUT DEVICE?(0=PT,1=MT,2=DS,3=CR;1) _____

(default device addresses are: PT=37,MT=10,DS=16,CR=30)

DEVICE ADDRESS?(0-076; xx) _____

(if PT the next two questions are skipped)

BIC/BTC EVEN DEVICE ADDRESS?(0-076) _____

UNIT NO?(0-3) _____

*7. SYSTEM GEN OUTPUT DEVICE?(0=PT,1=MT,2=DS,3=CP;1) _____

(default device addresses are: PT=37,MT=10,DS=16,CP=31)

DEVICE ADDRESS?(0-076; xx) _____

(if PT the next two questions are skipped)

BIC/BTC EVEN DEVICE ADDRESS?(0-076) _____

UNIT NO?(0-3) _____

8. WCS-UP1,15?(Y;N) _____

EVEN DEVICE ADDRESS?(0-076; 74) _____

9.FLOATING POINT PROCESSOR-UP2,15?(Y;N) _____

10.MEMORY TEST-UP3?(Y;N) _____

11.NOW MANY DISC CONTROLLERS-UP4,5,6,16?(0-4) _____

DISC n: (for models 0 and 4)

MODEL?(H,0-6;5) _____

(0=TYPE DB)

(1=TYPE DC)

(2=TYPE DD)

(3=TYPE DE)

(4=TYPE DF)

(5=TYPE DH)

(6=TYPE DJ)

DEVICE ADDRESS?(0-076; 16) _____

BIC/BTC EVEN DEVICE ADDRESS?(0-076) _____

BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

EACH SEEK COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

WHICH TRACK(T=0-0312/0625),SECTOR(S=0-2)?(T,S) _____

WHICH PLATTERS(P=0-1) & UNITS(U=0-3)?(PU,...) _____

DISC n: (for models 1, 2, and 3)

MODEL?(H,0-6;5) _____

DEVICE ADDRESS?(0-076; 15) _____

BIC/BTC EVEN DEVICE ADDRESS?(0-076) _____

BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

SEEK COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

WHICH HEAD?(0-023) _____

WHICH UNITS?(0-3/7) _____

DISC n: (for model 5)

MODEL?(H,0-6;5) _____

EVEN DEVICE ADDRESS?(0-076; 14) _____
 COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____
 WHICH CYLINDER(C=0-0632/01466), HEAD(H=0-4/022)?(C,H) _____
 WHICH UNITS?(0-3/7) _____
 DISC n: _____ (for model 6)
 MODEL?(H,0-6; 5) _____
 DEVICE ADDRESS?(0-076; 12) _____
 BIC/BTC EVEN DEVICE ADDRESS?(0-076) _____
 BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____
 CONTROLLER COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____
 WHICH TRACK(T=0-0114), SECTOR(S=0-033)?(T,S) _____
 WHICH UNITS?(0-3/7) _____
 12. HOW MANY MAG TAPE CONTROLLERS-UP7?(0-4) _____
 MAG TAPE 1:
 MODEL?(H,0,1) _____
 (0=NO STATUS WORD)
 (1=USES STATUS WORD)
 DEVICE ADDRESS?(0-076; 10) _____
 BIC/BTC EVEN DEVICE ADDRESS?(0-076) _____
 BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____
 MOTION COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____
 WHICH UNITS?(0-3/7) _____
 MAG TAPE 2:
 MODEL?(H,0,1) _____
 DEVICE ADDRESS?(0-076; 11) _____
 BIC/BTC EVEN DEVICE ADDRESS?(0-076) _____

BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____
MOTION COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____
WHICH UNITS?(0-3/7) _____

MAG TAPE 3:

MODEL?(H,0,1) _____
DEVICE ADDRESS?(0-076; 12) _____
BIC/BTC EVEN DEVICE ADDRESS?(0-076) _____
BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____
MOTION COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____
WHICH UNITS?(0-3/7) _____

MAG TAPE 4:

MODEL?(H,0,1) _____
DEVICE ADDRESS?(0-076; 13) _____
BIC/BTC EVEN DEVICE ADDRESS?(0-076) _____
BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____
MOTION COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____
WHICH UNITS?(0-3/7) _____

13. HOW MANY LINE PRINTERS-UP8?(0-4) _____

PRINTER 1:

DEVICE ADDRESS?(0-076; 35) _____
BIC/BTC EVEN DEVICE ADDRESS?(0-076) _____
BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

PRINTER 2:

DEVICE ADDRESS?(0-076; 35) _____
BIC/BTC EVEN DEVICE ADDRESS?(0-076) _____
BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

PRINTER 3:
DEVICE ADDRESS?(0-076; 35) _____
BIC/BTC EVEN DEVICE ADDRESS?(0-076) _____
BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____
PRINTER 4:
DEVICE ADDRESS?(0-076; 35) _____
BIC/BTC EVEN DEVICE ADDRESS?(0-076) _____
BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

14. HOW MANY DCM'S-UP9?(0-4) _____

DCM 1:
LINE ADAPTER TYPE?(H,0-3) _____
(0=ASYNCHRONOUS DIRECT CONNECT)
(1= " DATA SET)
(2=SYNCHRONOUS)
(3=BI-SYNCHRONOUS)

DEVICE ADDRESS?(0-076; 70) _____
DCM INTERRUPT ADDRESS ORIGIN?(0100-0260;0260) _____
BITS PER BYTE?(5-010;010) _____
LCB MEMORY PAGE?(070-077;075) _____
WHICH LINES(MAX OF 4)?(00-077) _____

DCM 2:
LINE ADAPTER TYPE?(H,0-3) _____
DEVICE ADDRESS?(0-076; 71) _____
DCM INTERRUPT ADDRESS ORIGIN?(0100-0260;0260) _____
BITS PER BYTE?(5-010;010) _____
LCB MEMORY PAGE?(070-077;075) _____
WHICH LINES(MAX OF 4)?(00-077) _____

DCM 3:

LINE ADAPTER TYPE? (H, 0-3) _____
 DEVICE ADDRESS? (0-076; 72) _____
 DCM INTERRUPT ADDRESS ORIGIN? (0100-0260; 0260) _____
 BITS PER BYTE? (5-010; 010) _____
 LCB MEMORY PAGE? (070-077; 075) _____
 WHICH LINES (MAX OF 4)? (00-077) _____
 DCM 4:
 LINE ADAPTER TYPE? (H, 0-3) _____
 DEVICE ADDRESS? (0-076; 73) _____
 DCM INTERRUPT ADDRESS ORIGIN? (0100-0260; 0260) _____
 BITS PER BYTE? (5-010; 010) _____
 LCB MEMORY PAGE? (070-077; 075) _____
 WHICH LINES (MAX OF 4)? (00-077) _____
 15. HOW MANY TELETYPES-UP10? (0-7) _____
 TTY 1:
 DEVICE ADDRESS? (0-076; 1) _____
 (note: if DA=DA of *4, next two questions are skipped)
 READ READY COMPLETE EVEN INTERRUPT ADDRESS? (0100-0276) _____
 WRITE READY COMPLETE EVEN INTERRUPT ADDRESS? (0100-0276) _____
 TTY 2:
 DEVICE ADDRESS? (0-076; 2) _____
 READ READY COMPLETE EVEN INTERRUPT ADDRESS? (0100-0276) _____
 WRITE READY COMPLETE EVEN INTERRUPT ADDRESS? (0100-0276) _____
 TTY 3:
 DEVICE ADDRESS? (0-076; 3) _____
 READ READY COMPLETE EVEN INTERRUPT ADDRESS? (0100-0276) _____

WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

TTY 4:
DEVICE ADDRESS?(0-076; 4) _____

READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

TTY 5:
DEVICE ADDRESS?(0-076; 5) _____

READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

TTY 6:
DEVICE ADDRESS?(0-076; 6) _____

READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

TTY 7:
DEVICE ADDRESS?(0-076; 7) _____

READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

16. HOW MANY CRT'S-UP11?(0-7) _____

CRT 1:
DEVICE ADDRESS?(0-076; 1) _____

(note: if DA=DA of *4, next two questions are skipped)

READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

CRT 2:
DEVICE ADDRESS?(0-076; 2) _____

READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

CRT 3:
DEVICE ADDRESS?(0-076; 3) _____

READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

CRT 4:
DEVICE ADDRESS?(0-076; 4) _____

READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

CRT 5:
DEVICE ADDRESS?(0-076; 5) _____

READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

CRT 6:
DEVICE ADDRESS?(0-076; 6) _____

READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

CRT 7:
DEVICE ADDRESS?(0-076; 7) _____

READ READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

WRITE READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

17. CARD READER-UP12?(Y;N) _____

DEVICE ADDRESS?(0-076; 30) _____

BIC/BTC EVEN DEVICE ADDRESS?(0-076) _____

BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

18. CARD PUNCH-UP13?(Y;N) _____

DEVICE ADDRESS?(0-076; 31) _____

BIC/BTC EVEN DEVICE ADDRESS?(0-076) _____

BIC/BTC COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____

19.PAPER TAPE-UP14?(Y;N) _____

DEVICE ADDRESS?(0-076; 37) _____

CHAR READY COMPLETE EVEN INTERRUPT ADDRESS?(0100-0276) _____