

PERKIN-ELMER

**OS/32
MULTI-TERMINAL MONITOR (MTM)
SYSTEM PLANNING AND OPERATOR**

Reference Manual

48-023 F00 R02

The information in this document is subject to change without notice and should not be construed as a commitment by the Perkin-Elmer Corporation. The Perkin-Elmer Corporation assumes no responsibility for any errors that may appear in this document.

The software described in this document is furnished under a license, and it can be used or copied only in a manner permitted by that license. Any copy of the described software must include the Perkin-Elmer copyright notice. Title to and ownership of the described software and any copies thereof shall remain in The Perkin-Elmer Corporation.

The Perkin-Elmer Corporation assumes no responsibility for the use or reliability of its software on equipment that is not supplied by Perkin-Elmer.

The Perkin-Elmer Corporation, Data Systems Group, 2 Crescent Place, Oceanport, New Jersey 07757

© 1977, 1981, 1983 by The Perkin-Elmer Corporation

Printed in the United States of America

TABLE OF CONTENTS

| | | | |
|----------|---|------|--|
| PREFACE | | vii | |
| CHAPTERS | | | |
| 1 | GENERAL DESCRIPTION | | |
| 1.1 | INTRODUCTION | 1-1 | |
| 1.2 | MULTI-TERMINAL MONITOR (MTM) SYSTEM REQUIREMENTS | 1-2 | |
| 1.3 | CONFIGURING A MULTI-TERMINAL MONITOR (MTM) SYSTEM | 1-2 | |
| 1.4 | OS/32 SYSTEM GENERATION (SYSGEN) OPTIONS | 1-3 | |
| 1.5 | FILE DESCRIPTORS | 1-3 | |
| 2 | MULTI-TERMINAL MONITOR (MTM) SYSTEM GENERATION (SYSGEN) | | |
| 2.1 | INTRODUCTION | 2-1 | |
| 2.2 | SELECTING MULTI-TERMINAL MONITOR (MTM) SYSTEM GENERATION (SYSGEN) OPTIONS | 2-1 | |
| 2.3 | GENERATING A MULTI-TERMINAL MONITOR (MTM) SYSTEM USING THE STANDARD COMMAND SUBSTITUTION SYSTEM (CSS) COMMAND FILE (MTMSGN.CSS) | 2-4 | |
| 2.4 | MULTI-TERMINAL MONITOR (MTM) SYSTEM GENERATION (SYSGEN) OPTIONS | 2-4 | |
| 2.4.1 | Sector Buffer Size (CDBUFSIZ) Option | 2-6 | |
| 2.4.2 | Special Console Buffer Size (CONBFSZ) Option | 2-7 | |
| 2.4.3 | Batch Job Central Processing Unit (CPU) Limit (CPULIMIT) Option | 2-8 | |
| 2.4.4 | Command Buffer Size (CSBFSZ) Option | 2-9 | |
| 2.4.5 | Input/Output (I/O) Class Table (IOTAB) Macro | 2-10 | |
| 2.4.6 | Multi-Terminal Monitor (MTM) Link (LNKOPT) Option | 2-12 | |
| 2.4.7 | Number of Special Console Buffers (NCONBFS) Option | 2-13 | |

CHAPTERS (Continued)

| | | |
|--------|---|------|
| 2.4.8 | Number of Command Buffers (NCSBFS) Option | 2-15 |
| 2.4.9 | Number of Command Substitution System (CSS) Levels (NCSSLVS) Option | 2-17 |
| 2.4.10 | Number of Message Buffers (NMSGs) Option | 2-18 |
| 2.4.11 | Number of Sector Length Buffers (NSECTS) Option | 2-19 |
| 2.4.12 | Number of Concurrent Users (NUSERS) Option | 2-20 |
| 2.4.13 | Multi-Terminal Monitor (MTM) Terminal Prompt Device Name (PRMTDEV) Macro | 2-21 |
| 2.4.14 | Multi-Terminal Monitor (MTM) Terminal Prompt Sequence (PRMTSEQ) Macro | 2-22 |
| 2.4.15 | Multi-Terminal Monitor (MTM) Task Queue Size (QUESIZE) Option | 2-24 |
| 2.4.16 | OS/32 Accounting Facility (SGN.ACT) Option | 2-25 |
| 2.4.17 | Batch Job Pause (SGN.BTCP) Option | 2-26 |
| 2.4.18 | Default User Volume (SGN.MVOL) Option | 2-27 |
| 2.4.19 | Batch Job Relative Priority (SGN.PRIO) Option | 2-28 |
| 2.4.20 | Forced Roll (SGN.ROLL) Option | 2-31 |
| 2.4.21 | Multi-Terminal Monitor (MTM) Scheduling (SGN.SCHR) Option | 2-32 |
| 2.4.22 | SPL/32 Interface (SGN.SPL) Option | 2-35 |
| 2.4.23 | Global and Local Variables (SGN.VAR) Option | 2-36 |
| 2.4.24 | Number of New Global Variables (SGN.@GLO) Option | 2-37 |
| 2.4.25 | Number of New Internal Variables (SGN.@INT) Option | 2-38 |
| 2.4.26 | HASP Interface (SGN.\$HSP) Option | 2-39 |
| 2.4.27 | Central Processing Unit (CPU) Limit Per Accounting User Class (UCPULIM) Macro | 2-40 |
| 2.4.28 | Input/Output (I/O) Count Limit Per Accounting User Class (UIOLIM) Macro | 2-42 |
| 2.4.29 | Multi-Terminal Monitor (MTM) Version Identification (VERSION) Macro | 2-45 |
| 2.5 | MEMORY REQUIREMENTS FOR SYSTEM GENERATION (SYSGEN) OPTIONS | 2-46 |

| 3 AUTHORIZED USER UTILITY

| | | |
|-------|---|------|
| 3.1 | INTRODUCTION | 3-1 |
| 3.2 | AUTHORIZED USER FILE (AUF) | 3-1 |
| 3.3 | BUILDING AND STARTING THE AUTHORIZED USER UTILITY | 3-4 |
| 3.4 | AUTHORIZED USER UTILITY COMMANDS | 3-6 |
| 3.4.1 | ACCOUNTS Command | 3-8 |
| 3.4.2 | ADD Command | 3-10 |
| 3.4.3 | CHANGE Command | 3-12 |
| 3.4.4 | DELETE Command | 3-14 |
| 3.4.5 | END Command | 3-15 |

CHAPTERS (Continued)

| | | |
|----------|---|------|
| 3.4.6 | LIST Command | 3-16 |
| 3.4.7 | MODE Command | 3-18 |
| 3.4.8 | PAUSE Command | 3-19 |
| 3.4.9 | PRIVILEGE Command | 3-20 |
| 3.4.10 | TOTAL Command | 3-26 |
| | | |
| 4 | OS/32 ACCOUNTING FACILITY | |
| | | |
| 4.1 | INTRODUCTION | 4-1 |
| | | |
| 4.2 | SYSTEM REQUIREMENTS | 4-1 |
| 4.2.1 | OS/32 System Generation (Sysgen) | 4-2 |
| 4.2.2 | Multi-Terminal Monitor (MTM) System Generation (SYSGEN) | 4-2 |
| 4.2.3 | Accounting Transaction File (ATF) | 4-2 |
| | | |
| 4.3 | LINK OPTION | 4-2 |
| | | |
| 4.4 | STARTING DATA COLLECTION | 4-3 |
| | | |
| 4.5 | DATA COLLECTION | 4-3 |
| 4.5.1 | Accounting Date and Time Formats | 4-3 |
| 4.5.2 | Job Related Accounting Data | 4-4 |
| 4.5.2.1 | Elapsed Time | 4-4 |
| 4.5.2.2 | Signon and Signoff Date and Time | 4-5 |
| 4.5.2.3 | Job Number | 4-5 |
| 4.5.2.4 | Date and Time a Job is Submitted | 4-5 |
| 4.5.2.5 | Total Number of Tasks Executed | 4-5 |
| 4.5.2.6 | Total Number of Command Substitution System (CSS) Calls Executed | 4-5 |
| 4.5.2.7 | Total Number of Commands Entered | 4-5 |
| 4.5.2.8 | Total Number of DISPLAY FILES Commands Entered | 4-6 |
| 4.5.2.9 | User Identification (Userid) | 4-6 |
| 4.5.2.10 | Renamed and Deleted Files | 4-6 |
| 4.5.2.11 | Account Number | 4-6 |
| 4.5.3 | Task Related Accounting Data | 4-6 |
| 4.5.3.1 | Central Processing Unit (CPU) Time | 4-7 |
| 4.5.3.2 | Roll Time | 4-8 |
| 4.5.3.3 | Wait Time | 4-8 |
| 4.5.3.4 | Date and Time a Task Started and Terminated | 4-8 |
| 4.5.3.5 | User Identification (Userid) | 4-8 |
| 4.5.3.6 | Memory Used by a Task | 4-8 |
| 4.5.3.7 | Count of Input/Output (I/O) Transfers to I/O Devices | 4-9 |
| 4.5.3.8 | Roll Count | 4-9 |
| 4.5.3.9 | File Descriptor (fd) from which a Task was Loaded | 4-9 |
| 4.5.3.10 | User Supplied Information | 4-10 |
| 4.5.3.11 | Account Number | 4-10 |
| 4.5.3.12 | Job Number | 4-10 |
| 4.5.3.13 | End of Task Code | 4-10 |

CHAPTERS (Continued)

| | | |
|-------|--|------|
| 4.6 | TERMINAL USER ACCESS TO ACCOUNTING INFORMATION | 4-11 |
| 4.6.1 | Task Termination | 4-11 |
| 4.6.2 | Task Control Block (TCB) Counter Overflow | 4-11 |
| 4.6.3 | Signoff Time | 4-11 |

5 MULTI-TERMINAL MONITOR (MTM) OPERATING PROCEDURES

| | | |
|-----|---|------|
| 5.1 | INTRODUCTION | 5-1 |
| 5.2 | LOADING AND ESTABLISHING THE MULTI-TERMINAL MONITOR (MTM) | 5-1 |
| 5.3 | START COMMAND | 5-2 |
| 5.4 | .MTM COMMAND | 5-4 |
| 5.5 | MULTI-TERMINAL MONITOR (MTM) END OF TASK CODES | 5-12 |
| 5.6 | MULTI-TERMINAL MONITOR (MTM) ERROR MESSAGES | 5-12 |

APPENDIXES

| | | |
|---|---|-----|
| A | MULTI-TERMINAL MONITOR (MTM) SYSTEM GENERATION (SYSGEN) OPTIONS | A-1 |
| B | AUTHORIZED USER UTILITY COMMANDS | B-1 |
| C | AUTHORIZED USER UTILITY MESSAGE SUMMARY | C-1 |
| D | ACCOUNTING TRANSACTION FILE (ATF) RECORD FORMATS | D-1 |
| E | MULTI-TERMINAL MONITOR (MTM) OPERATOR COMMAND SUMMARY | E-1 |

FIGURES

| | | |
|-----|---|-----|
| 1-1 | MTM System Environment | 1-1 |
| 3-1 | Authorized User File Record Format | 3-2 |
| 5-1 | Sample Output from the .MTM STATUS,USED Command | 5-9 |

TABLES

| | | | |
|-----|---|------|--|
| 2-1 | BUFFER TYPES USED FOR SPECIFIC I/O REQUESTS | 2-13 | |
| 2-2 | MODIFICATION OF MTM SUBTASK PRIORITIES WITH PSM ENABLED | 2-33 | |
| 2-3 | EXAMPLE I/O COUNT LIMITS SET BY UIOLIM | 2-44 | |
| 2-4 | DATA STRUCTURES BUILT BY MTM SYSGEN | 2-46 | |
| 3-1 | AUTHORIZED USER UTILITY MODES AND ALLOWABLE COMMANDS | 3-7 | |
| 3-2 | MTM ACCOUNT PRIVILEGES | 3-21 | |
| 5-1 | MTM END OF TASK CODES | 5-12 | |
| D-1 | ATF RECORD TYPES | D-1 | |

| | | | |
|-------|--|-------|--|
| INDEX | | Ind-1 | |
|-------|--|-------|--|

PREFACE

This manual is intended for system administrators and operators of an OS/32 Multi-Terminal Monitor (MTM) System. It describes step-by-step procedures for installing and operating an MTM system that is tailored to meet the needs of a particular organization.

Chapter 1 describes MTM and lists the system requirements and basic steps in configuring an MTM system. It also lists the recommended OS/32 system generation (sysgen) options that improve MTM efficiency. Chapter 2 describes the MTM sysgen procedure and defines the MTM sysgen procedure options and has been restructured. Chapter 3 describes building, loading, and starting the authorized user utility. It also lists the authorized user utility commands. Chapter 4 discusses the OS/32 accounting facility option. Chapter 5 describes the operating procedures needed to load, start, and control MTM. Appendix A lists the MTM sysgen options summary. Appendix B summarizes the authorized user utility commands. Appendix C is an authorized user utility message summary. Appendix D contains the accounting transaction file (ATF) record formats. Appendix E lists the MTM operator commands.

Revision 02 of this manual describes the use of the following additional or enhanced MTM sysgen options:

- batch job pause
- default system volume
- MTM subtask scheduling
- SPL/32 interface
- number of new global variables
- number of internal variables
- HASP interface
- CPU limit for accounting user classes
- input/output (I/O) count limit for accounting user classes

Also described are the account privileges that can be added to or changed through the authorized user utility and the USED parameter to the .MTM command.

This manual is intended for use with the OS/32 R06.2 software release and higher.

For further information on the contents of all Perkin-Elmer manuals, see the 32-Bit Systems User Documentation Summary.

CHAPTER 1
GENERAL DESCRIPTION

1.1 INTRODUCTION

The OS/32 Multi-Terminal Monitor (MTM) is an interactive multi-terminal, time-sharing system that executes as an executive task (e-task) in an OS/32 environment. MTM can be loaded, started, and cancelled without disrupting the execution of other tasks running concurrently under OS/32. All tasks loaded and started from a user terminal execute as subtasks of MTM. Figure 1-1 shows the relationship of application tasks to OS/32 and MTM in a typical MTM system environment.

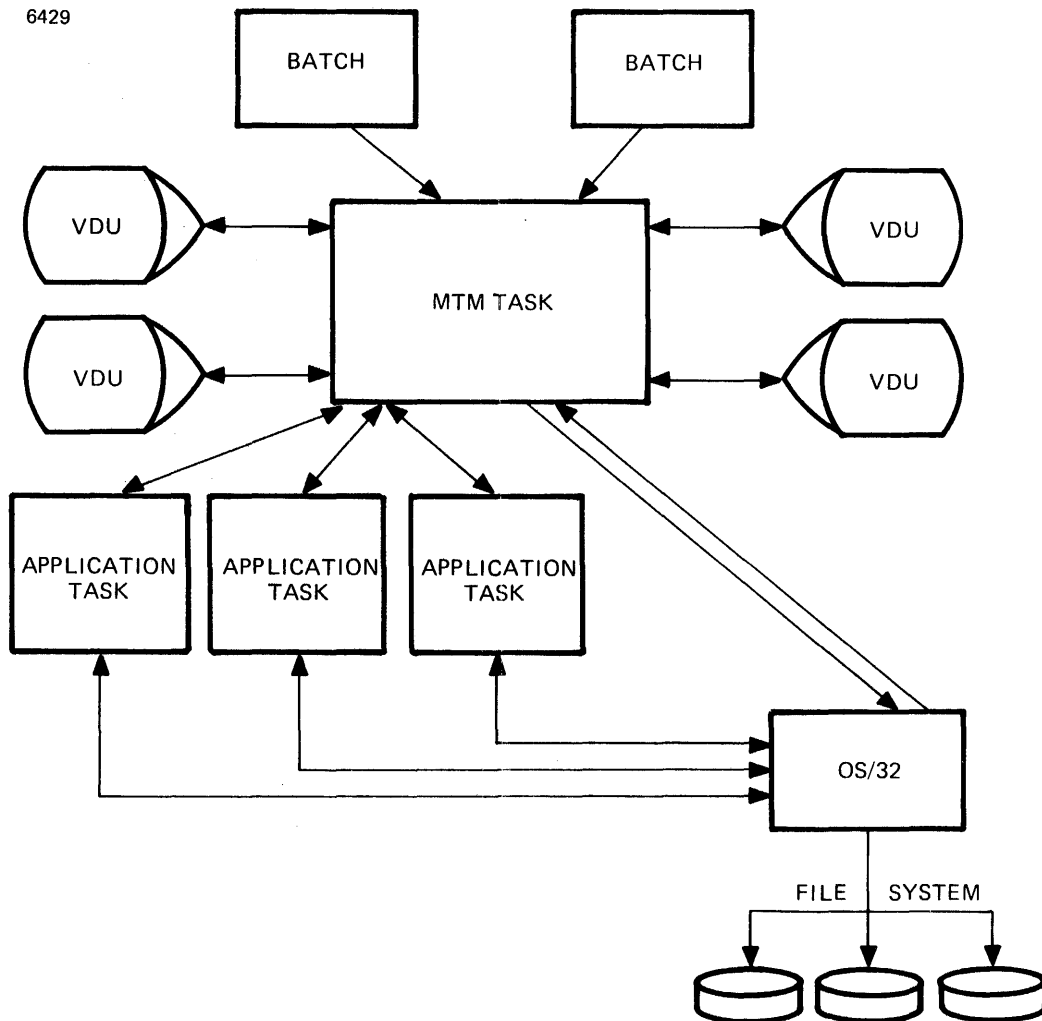


Figure 1-1 MTM System Environment

1.2 MULTI-TERMINAL MONITOR (MTM) SYSTEM REQUIREMENTS

The minimum hardware and software requirements for configuring an MTM system are:

- OS/32 R06.2
- Perkin-Elmer processor
- Memory access controller (MAC) or memory address translator (MAT) (as appropriate)
- Universal clock module
- Power fail/auto restart option
- Console device
- Relocating (REL) loader, bootloader, or loader storage unit (LSU) to load the operating system
- 256kb of memory (288kb or greater is recommended)
- 10Mb disk device
- ASCII terminal device

1.3 CONFIGURING A MULTI-TERMINAL MONITOR (MTM) SYSTEM

The following procedure is used to configure an MTM system.

- | 1. Perform an OS/32 system generation (sysgen). See the OS/32
| System Generation/32 (Sysgen/32) Reference Manual.
2. Unpackage the MTM software. See the MTM packaging document
for instructions.
- | 3. Perform an MTM sysgen, modifying the MTM sysgen options to
| reflect the requirements of your installation. See Chapter
| 2.
4. Build an authorized user file (AUF), using the authorized
user utility described in Chapter 3.
5. Load and start MTM. See Chapter 5.

1.4 OS/32 SYSTEM GENERATION (SYSGEN) OPTIONS

When generating an operating system for an MTM system, select the following sysgen options to improve the overall efficiency and effectiveness of the system:

- Secondary directory support - this option adds a memory-resident directory to the system to reduce file search time.
- Segmentation support - this option allows the task image to be segmented into a private image area containing impure code and a shared image area containing pure code. During multiple simultaneous executions of the task, a copy of the private image is loaded into memory for each user, but only one copy of the shared image is loaded.
- Roll support - this option ensures that MTM can service a number of tasks in a limited amount of memory space.
- Spool support - this option provides input and output spooling capabilities to the MTM user.
- Accounting support - this option is required if the OS/32 accounting facility is to be used with MTM. See Chapter 4.

1.5 FILE DESCRIPTORS

Many of the command formats in this manual require the user to enter a file descriptor (fd). File descriptors are entered in the following format.

Format:

$$\left[\left\{ \begin{array}{l} \text{voln:} \\ \text{dev:} \end{array} \right\} \right] [\text{filename} \ . [\text{ext}]] \left[/ \left\{ \begin{array}{l} \text{actno} \\ \text{file class} \end{array} \right\} \right]$$

Parameters:

voln: is the name of a disk volume. It may be from 1- to 4-characters. The first character must be alphabetic and the remaining, alphanumeric. If voln: is not specified, the default volume specified by the VOLUME command is used.

dev: is a 1- to 4-character device name. The first character must be alphabetic and the remaining, alphanumeric.

filename is the name of a file and is from 1- to 8-characters. The first character must be alphabetic and the remaining, alphanumeric. If a filename is specified when a device mnemonic is specified as voln:, the filename is ignored.

.ext is the name of the extension and consists of from 1- to 3-alphanumeric characters. For some commands, extensions are appended by default. If the period is omitted, the default is blanks.

actno is a decimal number ranging from 0 through 65,535 specifying the account number associated with the file. Account numbers 1 through 65,535 are used by MTM. Account number 255 is a restricted account. MTM uses account 255 for the authorized user file (AUF). Account number 0 is used for system files and is the default for all operator commands entered from the system console. Tasks must have account privileges to specify an account number in this field.

file class is the class name of the file and consists of one character. The class names are:

- P for private file
- G for group file
- S for system file

NOTE

The file class can be specified by a terminal user or the system operator. If the system operator specifies P, G, or S, the operation is performed to account 0 only.

|
|

CHAPTER 2
MULTI-TERMINAL MONITOR (MTM) SYSTEM GENERATION (SYSGEN)

2.1 INTRODUCTION

This chapter describes MTM sysgen procedures and options available for configuring an MTM system.

The MTM task consists of two modules:

- MTMMAIN
- MTMSGN

MTMMAIN is a fixed length object file containing the executable code for MTM.

MTMSGN is a source file containing the MTM sysgen options. These options can be modified to meet the requirements of your installation.

The procedure for performing an MTM sysgen is as follows:

- Select the MTM sysgen options that pertain to your installation.
- Use the command substitution system (CSS) command file (MTMSGN.CSS) to build the MTM task.

The following sections describe the MTM sysgen procedure in detail.

2.2 SELECTING MULTI-TERMINAL MONITOR (MTM) SYSTEM GENERATION (SYSGEN) OPTIONS

MTMSGN is a source module that contains information necessary to configure an MTM system that meets the requirements of a particular installation. For example, one data field in this file tells MTM how many users can be signed on the system concurrently. Another data field tells MTM how many command buffers to reserve in memory. See Section 2.4 for a complete description of the sysgen options contained in this file.

Perkin-Elmer supplies a standard MTM sysgen parameter file (MTMPARMS.MAC) with the MTM package. This file contains the following code:

```

**MTM.SGN
        SPACE 2
        VERSION
        SPACE 2
SGN.ACT EQU 0          NO ACCOUNTING SUPPORT
SGN.SCHR EQU 1        0=NONE / 1=DYNAMIC / 2=PSM
SGN.ROLL EQU 1        FORCE TASK ROLLABLE
SGN.PRIO EQU 6        BATCH PRIORITY
SGN.BTCP EQU 0        CANCEL BATCH JOB ON PAUSE
SGN.SPL EQU 0         NO SPL/32 SUPPORT
SGN.$HSP EQU 0        NO $HASP INTERFACE SUPPORT
SGN.MVOL EQU C'      ' NO DEFAULT USER VOLUME
SGN.VAR EQU 28        NUMBER OF VARIABLES
SGN.@GLO EQU 18       NUMBER OF GLOBAL VARIABLES
SGN.@INT EQU 8        NUMBER OF INTERNAL VARIABLES
NUSERS EQU 4          NUMBER OF USERS
NCSSLVS EQU 8         NUMBER OF CSS LEVELS
NCSBFS EQU 10        NUMBER OF EXTRA CSS BUFFERS
NCONBFS EQU 0         NUMBER OF CONSOLE BUFFERS
NSECTS EQU NUSERS/2  NUMBER OF SECTOR BUFFERS
NMSGs EQU NUSERS     NUMBER OF MESSAGE BUFFERS
QUESIZE EQU NUSERS   SIZE OF TASK QUEUE
CSBFSZ EQU 80        SIZE OF CSS BUFFERS
CONBFSZ EQU 0        SIZE OF CONSOLE BUFFERS
CDBUFSIZ EQU 3       SIZE OF SECTOR BUFFERS
CPULIMIT EQU 0       NO CPU TIME LIMIT
        SPACE 2
        LINKOPT 4,30          LINK OPTIONS      IOB,LU

```

In addition, MTMSGN defines values for the prompt sequence, the input/output (I/O) class table, user I/O limits, and user central processing unit (CPU) limits. See the IOTAB, UIOLIM, UCPULIM, PRMTSEQ, and PRMTDEV macro instructions in Section 2.4 for more information on the default values for these sysgen options.

MTM sysgen options that differ from those defined by the standard MTM sysgen parameter file (MTMPARMS.MAC) can be selected by editing MTMPARMS.MAC via Edit/32.

In the following example, MTMPARMS.MAC is edited to permit five accounting classes, SPL/32 and HASP support, assignment of a user default volume, 25 concurrent users, larger CSS buffers, and an increased number of I/O blocks and logical units. In addition, several macro calls have been appended to MTMPARMS.MAC. Discussion of these macro calls is deferred until their respective sections in this chapter. Identification name '3230-B System' is assigned to this MTM system.


```

**MTM.SGN
      SPACE 2
      VERSION '3230-B SYSTEM'
      SPACE 2
SGN.ACT EQU 5          5 ACCOUNTING CLASSES
SGN.SCHR EQU 1        0=NONE / 1=DYNAMIC / 2=PSM
SGN.ROI.L EQU 1      FORCE TASK ROI.LABLE
SGN.PRIO EQU 6       BATCH PRIORITY
SGN.BTCP EQU 0       CANCEL BATCH JOB ON PAUSE
SGN.SPL EQU 25      SPL/32 SUPPORT - 25 BUFFERS
SGN.$HSP EQU 1      1 $HASP INTERFACE
SGN.MVOL EQU C'MTM ' DEFAULT USER VOLUME - MTM
SGN.VAR EQU 28      NUMBER OF VARIABLES
SGN.@GLO EQU 15     NUMBER OF GLOBAL VARIABLES
SGN.@INT EQU 8      NUMBER OF INTERNAL VARIABLES
NUSERS EQU 25       NUMBER OF USERS
NCSSLVS EQU 8       NUMBER OF CSS LEVELS
NCSBFS EQU 10      NUMBER OF EXTRA CSS BUFFERS
NCONBFS EQU 0       NUMBER OF CONSOLE BUFFERS
NSECTS EQU NUSERS/2 NUMBER OF SECTOR BUFFERS
NMSGs EQU NUSERS   NUMBER OF MESSAGE BUFFERS
QUESIZE EQU NUSERS SIZE OF TASK QUEUE
CSBFSZ EQU 120     SIZE OF CSS BUFFERS
CONBFSZ EQU 0       SIZE OF CONSOLE BUFFERS
CDBUFSIZ EQU 3     SIZE OF SECTOR BUFFERS
CPULIMIT EQU 0     NO CPU TIME LIMIT
      SPACE 2
      LINKOPT 25,80      LINK OPTIONS      IOB,LU
      SPACE 2
      IOTAB (0,INDX),(1,CNTG),(2,DEV),(3,SPL),(4,TAPE)
      SPACE 2
      UCPULIM 30,30,,,40,3600,80,10800
      SPACE 2
      UIOLIM (00,01,02,03,04),(100,101,102,103,104),      U CLS 0      *
              (10,11,12,13,14),,                          U CLS 1      *
              (20,21,22,23,24),(120,121,122,123,124),    U CLS 2      *
              (30,31,32,33,34),(130,131,132,133,134)      U CLS 3
      SPACE 2
      PRMTDEV (FOX,CT30,CT32,CT34),                          *
              (SIGMA,SM50)
      SPACE 2
      PRMTSEQ (FOX,(*,08FF),(-,08FF),(>,FF),(BUILD>,FF)), *
              (SIGMA)

```

It is recommended that the MTM sysgen parameter file that results from editing MTMPARMS.MAC be saved with a different extension. For example, MTMPARMS.MAC can be edited as above, with the new file saved as MTMPARMS.xxx where xxx is a 1- to 3-character alphanumeric extension. For a further explanation of this parameter, see Section 2.3.

See the OS/32 Edit User Guide for information on editing files.

2.3 GENERATING A MULTI-TERMINAL MONITOR (MTM) SYSTEM USING THE STANDARD COMMAND SUBSTITUTION SYSTEM (CSS) COMMAND FILE (MTMSGN.CSS)

Perkin-Elmer supplies a CSS command file (MTMSGN.CSS) with the MTM package. MTMSGN.CSS assigns the SYSSTRUC.MLB macro library. Therefore, MTMSGN.CSS may have to be edited via Edit/32 to reflect the location of this file for a particular installation. For example, a volume and/or account classification, i.e., S, G, P, may have to be specified for SYSSTRUC.MLB.

To call the standard CSS command file, enter the command as follows:

Format:

MTMSGN. { xxx }
 { MAC }

Parameter:

xxx is a 1- to 3-character alphanumeric field which must be preceded by a space.

Functional Details:

The parameter specified in the MTMSGN command must match the extension specified for the MTM sysgen parameter file (MTMPARMS). If the edited MTM sysgen parameter file was not saved with an extension different from that of the standard MTM sysgen parameter file, or the standard file is being used, MAC must be specified.

Successful execution of MTMSGN.CSS returns condition code = 0 upon completion of the MACRO/32 and CAL/32 processing phases. Upon completion of the Link phase of MTMSGN.CSS, condition code = 1 is returned with the following message:

*** WARNING: ABSOLUTE SPACE LESS THAN 100 ***

2.4 MULTI-TERMINAL MONITOR (MTM) SYSTEM GENERATION (SYSGEN) OPTIONS

The MTM sysgen options are:

- CDBUFSIZ - Sector buffer size
- CONBFSZ - Special console buffer size

- CPULIMIT - Batch job CPU limit
- CSBFSZ - Command buffer size
- IOTAB - I/O class table
- LINKOPT - Number of I/O blocks and logical units for the LINK
OPTION command
- NCONBFS - Number of special console buffers
- NCSBFS - Number of command buffers
- NCSLVS - Number of CSS levels
- NMSGs - Number of message buffers
- NSECTS - Number of sector length buffers
- NUSERS - Number of concurrent users
- PRMTDEV - MTM terminal prompt device name
- PRMTSEQ - MTM terminal prompt sequence
- QUESIZE - MTM task queue size
- SGN.ACT - Number of accounting classes
- SGN.BTCP - Batch job pause
- SGN.MVOL - Default user volume
- SGN.PRIO - Batch job relative priority
- SGN.ROLL - Forced roll
- SGN.SCHR - Scheduling mechanism
- SGN.SPL - Number of SPL/32 send data buffers
- SGN.VAR - Number of global and local variables
- SGN.@GLO - Number of new global variables
- SGN.@INT - Number of internal variables
- SGN.\$HSP - HASP interface
- UCPULIM - CPU limit for each accounting user class
- UIOLIM - I/O count limit for each accounting user class
- VERSION - MTM version identification

2.4.1 Sector Buffer Size (CDBUFSIZ) Option

The sector buffer size (CDBUFSIZ) option specifies the number of sectors in a sector length buffer. For example, using this option to increase the length of the buffers for the DISPLAY FILES and variable-related commands causes a corresponding increase in the speed of MTM. The standard MTMPARMS file specifies a value of 3 for this option. To modify the standard value, edit this option to the following:

Format:

CDBUFSIZ EQU n

Parameters:

n is a decimal number from 0 through 6.

Functional Details:

The size of each sector buffer is set according to this formula:

$$(CDBUFSIZ*256)+256$$

The use of this option to increase the speed of the DISPLAY FILES command pertains only to files on disks that have been marked on with a secondary directory. The number of sectors calculated by the CDBUFSIZ option are read from the SYSTEM.DIR file each time a DISPLAY FILES command is executed. If the disk is marked on with a secondary directory that can be contained in memory, MTM scans the memory buffer to search the directory. If CDBUFSIZ is set to 0, MTM does not scan the secondary directory to get the desired files. Instead, the primary directory is scanned.

When variable support is included in an MTM system, the size and number of sector buffers allocated determine the number of variables that can be contained in memory. Sometimes, a user will attempt to define more variables than can fit into the user's allocated sector buffers. When this occurs, one of the allocated sector buffers is paged to disk, and a freed sector buffer is made available to the user. This allows the user to define more variables.

2.4.2 Special Console Buffer Size (CONBFSZ) Option

The CONBFSZ option specifies the maximum length of the special console buffers generated by the NCONBFS option. Since no special console buffers are reserved by the standard MTMPARMS file, the standard value of this option is 0. To specify the maximum length of each special console buffer reserved by MTM, edit this option to the following:

Format:

CONBFSZ EQU n

Parameters:

n is 0 or a decimal number from 72 through 32,767 specifying the length of each special console buffer in the system. If the standard value of this option is changed, the value of n should be greater than the number specified in the command buffer length (CSBFSZ) option. See Section 2.4.4.

2.4.3 Batch Job Central Processing Unit (CPU) Limit (CPULIMIT) Option

The batch job CPU limit (CPULIMIT) option defines the maximum CPU time (in seconds) for a batch job running under MTM. No CPU limit is set by the standard MTMPARMS file. To set the CPU time limit, edit this option to the following:

Format:

CPULIMIT EQU n

Parameters:

n is a decimal number from 0 through 86,400 (24 hours) specifying the maximum CPU time (in seconds) to which each batch job is limited. If 0 is specified, no limit applies.

2.4.4 Command Buffer Size (CSBFSZ) Option

The command buffer length (CSBFSZ) option sets the maximum length of the MTM system command buffers, including the buffers used for command substitution system (CSS) commands. The standard value for this option is 80. To modify the standard value, edit this option to the following:

Format:

```
CSBFSZ EQU n
```

Parameter:

n is a decimal number from 72 through 1,024.

Functional details:

The user should define a command buffer length one greater than needed. If a number less than 72 is specified for n, a message is output and MTM goes to end of task code 7 when the .MTM START command is entered.

2.4.5 Input/Output (I/O) Class Table (IOTAB) Macro

The I/O class table (IOTAB) macro defines values for the I/O class table. To change the default values, enter the IOTAB macro instruction as follows:

Format:

```
[label] IOTAB (classno,classid [,iocount] )0  
          '...'   
          (classno,classid [,iocount] )n
```

Operand:

| | |
|---------|--|
| classno | is a decimal number from 0 through 31 specifying a device or file class number. |
| classid | is a user-specified 4-character alphanumeric class identification associated with each device or file class. The classid is installation dependent. |
| iocount | is a decimal number from 0 through 1,000,000 specifying the maximum number of I/O transfers to which each batch job associated with a particular device or file class is limited. If iocount is omitted, the default is 0 (no limit on I/O transfers). |

Functional Details:

MTMSGN defines default values for the first four classes of the I/O class table as follows:

```
IOTAB (0,INDX,0),(1,CNTG,0),(2,DEV,0),(3,SPL,0)
```

These default values may be modified by appending the IOTAB macro call, with the desired parameters, to the MTMPARMS.xxx file, as demonstrated in Section 2.3.

2.4.6 Multi-Terminal Monitor (MTM) Link (LNKOPT) Option

The MTM link (LNKOPT) option sets the maximum number of I/O blocks and logical units available to MTM. The standard MTMPARMS file permits four I/O control blocks and 30 logical units. To modify the standard values of LNKOPT, edit this option to the following:

Format:

LNKOPT n,m

Parameters:

n is the maximum number of I/O control blocks available to a task.

m is the maximum number of logical units available to a task.

Functional Details:

The number of I/O control blocks should equal the number of concurrent users (NUSERS) set by MTMPARMS.

The minimum number of logical units that should be specified can be determined by the following formula:

$lu = 6 + (NCSSLVS + 1) * NUSERS$ or 254, whichever is smaller.

MTM dynamically allocates logical units to subtasks as needed. The actual number of logical units available to MTM for dynamic allocation can be calculated as follows:

available logical units = (number of logical units specified at MTM sysgen (LNKOPT)) - NUSERS - 6.

To avoid a decrease in response time, the recommended number of logical units for NUSERS greater than 5 is:

$6 + NUSERS * 2 + (NCSSLVS * 2)$

To determine if the number of logical units specified by LNKOPT is sufficient, check the number of times MTM goes into a logical unit (lu) wait state by using the .MTM STATUS,WAITS command. This command returns a table similar to the one shown below.

| | .MTM ST,W | | | | | |
|----------|-------------|------|---------|-------|-------------|--|
| 13:14:32 | .MTM: TOTAL | PEAK | CURRENT | WAITS | | |
| 13:14:34 | .MTM: 42 | 34 | 11 | 0 | CMD BUFFERS | |
| 13:14:37 | .MTM: 0 | 0 | 0 | 0 | CON BUFFERS | |
| 13:14:40 | .MTM: 7 | 4 | 0 | 0 | SEC BUFFERS | |
| 13:14:42 | .MTM: 50 | 24 | 19 | 0 | LU | |

A decision to increase or decrease the number of logical units specified by LNKOPT can be based on information returned by .MTM STATUS,WAITS.

The MTMAIN object module has the following LINK OPTION commands imbedded:

```
OPTION COMMUNICATION,CONTROL,XSVCL,ETASK,INTERCEPT,ABSOLUTE=0
```

These options can be overridden by using the NDCMD command to suppress imbedded LINK options.

The LINK OPTIONS IOBLOCKS and LU are contained in the MTMSGN source module. These options are selected as described above.

For more information on LINK commands and OPTIONS, see the OS/32 Link Reference Manual.

2.4.7 Number of Special Console Buffers (NCONBFS) Option

The NCONBFS option specifies the maximum number of special console buffers that can be used concurrently during MTM execution. No special console buffers are specified in the standard MTM sysgen parameter file. To specify the number of special console buffers to be reserved, edit this option to the following:

Format:

NCONBFS EQU n

Parameters:

n is a decimal number from 0 through 64.

Functional Details:

The maximum number of special console buffers that can be used at one time is equal to the number of users (NUSERS). These special console buffers are used only when an I/O request is made to a user terminal that requires an I/O buffer larger than the size specified in the command buffer length (CSBFSZ) option. An example of this is the use of graphic buffers.

Table 2-1 shows the types of buffers used in different situations.

TABLE 2-1 BUFFER TYPES USED FOR SPECIFIC I/O REQUESTS

| I/O REQUEST SIZE | BUFFER TYPE | |
|--------------------------------|---------------------------|--------|
| | READ | WRITE |
| < CSBFSZ | TUB | TUB |
| > CSBFSZ and < CONBFSZ | CON | CON |
| > CSBFSZ and CONBFSZ = 0 | TRUNCATE to CSBFSZ | DIRECT |
| > CONBFSZ | TRUNCATE to CONBFSZ | DIRECT |

TUB (terminal buffer) Dedicated buffer of size CSBFSZ for each user.

CON (console) Special console buffer of size CONBFSZ.

TRUNCATE Read request will be truncated.

DIRECT Write request will be performed directly from the task. The task is made nonrollable for the duration of the I/O call.

The decision to buffer large writes through the special console buffers, or to perform the write directly from the task, should be based on:

- the speed of the terminal,
- the size of the write, and
- the impact on other system users caused by the user task (u-task) being nonrollable for the duration of the write request.

If an insufficient number of console buffers is allocated, a deadlock between two tasks might occur; e.g., both tasks enter into a console buffer wait state. If this happens, one of the two tasks involved should be cancelled. This deadlock situation is likely to occur if the special console buffers are used for read purposes and there is an insufficient amount of task space causing tasks to be rolled. To monitor the number of console buffer waits, enter the .MTM STATUS parameter with a WAITS option.

2.4.8 Number of Command Buffers (NCSBFS) Option

The MTM program requires a minimum number of command buffers as follows:

- One buffer for each user (terminal or batch job)
- One buffer for each active CSS level
- One buffer for each MESSAGE command
- One buffer for each BROADCAST message
- One buffer for each new global or new internal variable

MTM terminal response time depends on the number of available command buffers. To avoid poor performance, an adequate number of buffers must be allowed for MTM use. During MTM sysgen, three command buffers are reserved for each user. However, additional buffers can be reserved according to the number of CSS levels that can be active under MTM. The NCSBFS option tells MTM how many additional command buffers are to be reserved. The length of these command buffers is specified by the CSBFSZ option. The standard value for the NCSBFS option is 10.

To modify the standard value, edit this option to the following:

Format:

NCSBFS EQU n

Parameters:

n is a decimal number specifying the number of additional command buffers that can be used by the MTM system. Use the following formula to calculate n for your installation:

$$n = (\text{number of users}) \times (\text{number of CSS levels} + 1) / 3.$$

The minimum number for n is 0.

Functional Details:

Response time may be adversely affected if the value specified for n is too small. Conversely, memory space is wasted if n is set above the maximum buffers needed.

The program development feature of MTM uses new global and new internal variables extensively. A user who plans to employ the program development feature, or expects to use a large number of new global and new internal variables, may require a large number of command buffers. The recommended minimum number of extra command buffers that should be specified by users of the program development feature is 15.

The following methods can be used to judge whether the value specified for n is too small or too large for your system:

- Terminate MTM with the .MTM QUIESCE command. At end of task, MTM outputs the number of times it had to wait to get a command buffer.
- Enter .MTM STATUS,WAITS at the system console. This command displays the total number of buffers available, peak usage, current usage, and the number of times MTM had to wait for system resources on the user's behalf.
- Enter .MTM STATUS, USED at the system console. This command displays in histogram form the number of times command buffers were used during MTM execution.

2.4.9 Number of Command Substitution System (CSS) Levels (NCSSLVS) Option

The number of CSS levels (NCSSLVS) option sets the maximum nesting depth of CSS routines. If one CSS routine tries to call another and exceeds the designated depth, the system responds with an error message and CSS processing ceases. To modify the standard value, edit this option to the following:

Format:

```
NCSSLVS EQU n
```

Parameters:

n is a decimal number specifying the maximum number of CSS levels. The minimum value for n is one; the maximum value is 1,024.

2.4.10 Number of Message Buffers (NMSGs) Option

The number of message buffers (NMSGs) option defines the maximum number of message buffers that MTM can reserve for operator commands and accounting messages from the file manager. The standard value for this option is NUSERS.

To modify the standard value, edit this option to the following:

Format:

NMSGs EQU n

Parameters:

n is a decimal number from 1 through 1,024. A recommended value for n is NUSERS.

Functional Details:

MTM reserves 76 bytes per buffer. If these buffers overflow, the following message is displayed by the operating system:

SVC6-ERR TYPE=NMSG POS=

2.4.11 Number of Sector Length Buffers (NSECTS) Option

The number of sector length buffers (NSECTS) option specifies the maximum number of sector length buffers that can be used by MTM for commands that require this buffer type. These commands are DISPLAY FILES, \$GLOBAL, \$LOCAL, \$SET, and \$FREE. The number of sector length buffers specified by the standard MTMPARMS file is NUSERS/2. To modify the standard value, edit this option to the following:

Format:

NSECTS EQU n

Parameters:

n is a decimal number specifying the maximum number of 256-byte sector buffers. The minimum value for n is 1. A recommended value for n is NUSERS/2.

Functional Details:

When variable support is included in an MTM system through the SGN.VAR option, a sufficient amount of sector buffers should be reserved to hold MTM variables. A 256-byte (one sector) buffer can contain a maximum of seven variables. The amount of sector buffers needed depends on the number of users, the number of variables being used, and the size of the sector buffers. Ideally, two 256-byte sector buffers per user should be allocated; a maximum of four sector buffers per user can be allocated to a sector buffer pool. Very large sector buffers can also be allocated using the sector buffer size (CDBUFSIZ) option. If an insufficient number of sector buffers is allocated, the buffer pool can run empty, causing the user to go into a sector buffer wait state.

When variable information is not required, sector buffers are paged to disk. The resulting freed buffers are returned to the sector buffer pool. Because of this paging to and from disk, the size of each sector buffer is an important factor in processing variables.

2.4.12 Number of Concurrent Users (NUSERS) Option

The number of concurrent users (NUSERS) option defines the maximum number of terminal users and batch jobs that can be executed concurrently. The value specified for this option in the standard MTMPARMS file is 4. To modify the standard value, edit this option to the following:

Format:

NUSERS EQU n

Parameter:

n is a decimal number from 1 through 64.

2.4.13 Multi-Terminal Monitor (MTM) Terminal Prompt Device Name (PRMTDEV) Macro

The PRMTDEV macro defines the devices to be used with the prompt sequences specified in the PRMTSEQ macro. Any terminal device not defined by the PRMTDEV macro will use the standard prompt sequences. The PRMTDEV macro is used as follows:

Format:

```
[label] PRMTDEV (pname1, dev1 [, ..., devn])1  
                [, ..., (pnamen, dev1 [, ..., devn])n]
```

```
[label] PRMTDEV (SIGMA, dev1 [, ..., devn])
```

Operands:

pname is the 1- to 8-character prompt sequence name specified in the PRMTSEQ option. See Section 2.4.14.

dev is the device name of the terminal.

SIGMA is the predefined prompt sequence for the Perkin-Elmer SIGMA 10 terminal.

Example:

```
PRMTDEV (FOX, CT30, CT32, CT34), (SIGMA, SM50)
```

2.4.14 Multi-Terminal Monitor (MTM) Terminal Prompt Sequence (PRMTSEQ) Macro

The PRMTSEQ macro defines prompt sequences for terminals requiring special prompt sequences. This macro is used in conjunction with the MTM terminal prompt device name (PRMTDEV) macro. This macro is only required if the user wishes to use a terminal prompt sequence other than the standard sequence.

Format:

[label] PRMTSEQ (SIGMA)

[label] PRMTSEQ (pname₁, (com-req, x), (task-exec, x),
(task-read, x), (build-read, x))₁
[, ...,
(pname_n, (com-req, x), (task-exec, x),
(task-read, x), (build-read, x))_n]

[label] PRMTSEQ (STANDARD, (com-req, x), (task-exec, x),
(task-read, x), (build-read, x))₁

Parameters:

| | |
|------------|--|
| SIGMA | requests the predefined prompt sequence for the Perkin-Elmer SIGMA 10 terminal. |
| pname | is a 1- to 8-character alphanumeric prompt sequence name beginning with an alphanumeric character. This parameter corresponds to that specified for the PRMTDEV macro. |
| com req | is 0 to 8 ASCII characters defining the prompt sequence for a command request. |
| task exec | is 0 to 8 ASCII characters defining the prompt sequence for task or CSS execution. |
| utask read | is 0 to 8 ASCII characters defining the prompt sequence for a user task read. |
| build read | is 0 to 8 ASCII characters defining the prompt sequence for a build read. |

STANDARD redefines the standard prompt sequence which is as follows:

- command read (*,FFFFFF)
- task/CSS executing (-,FFFFFF)
- user task read (>,FFFFFF)
- build read (B>,FFFFFF)

x is a hexadecimal pad sequence from 0 to 8 bytes.

Functional Details:

The total length of the ASCII character sequence and the hexadecimal pad sequence cannot exceed 8 bytes.

When specifying more than one set of prompt sequences, a comma must be placed immediately following the last parameter specified on a line, and an asterisk must appear in column 72 to designate continuation of the operands.

Example:

```
PRMTSEQ (BS,(*,08FF),(-,08FF),(>,FF),(BUILD>,FF)), *  
(SIGMA)
```

2.4.15 Multi-Terminal Monitor (MTM) Task Queue Size (QUESIZE) Option

The task queue size (QUESIZE) option specifies the maximum size of the task queue used by MTM. The standard MTMPARMS file specifies NUSERS for this option. To modify the standard value, edit this option to the following:

Format:

QUESIZE EQU n

Parameter:

n is a decimal number specifying the maximum number of slots for the MTM task queue. The maximum value for this option is 200. A recommended value for n is NUSERS.

2.4.16 OS/32 Accounting Facility (SGN.ACT) Option

The accounting facility (SGN.ACT) option enables support for the OS/32 accounting facility under MTM. The standard MTMPARMS file does not support the accounting facility. To enable support for the accounting facility and specify the maximum number of device and file classes that can be used, edit this option to the following:

Format:

SGN.ACT EQU n

Parameters:

n is a decimal number from 2 through 32 specifying the maximum number of device and file classes to be used by the accounting facility. A 0 indicates no accounting support.

Functional Details:

The value of n should be greater than, or equal to, the number specified in the OS/32 accounting configuration statement. A minimum value of 2 is required to support indexed and contiguous files.

2.4.17 Batch Job Pause (SGN.BTCP) Option

The batch job pause (SGN.BTCP) option allows a batch job running under MTM to pause without being cancelled. The standard value of this option causes a batch job running under MTM to be cancelled whenever execution of that job is suspended. To enable the batch job pause option, edit this option to the following:

Format:

```
SGN.BTCP EQU 1
```

Functional Details:

The batch job pause option is enabled by equating SGN.BTCP to 1. If this option is enabled, a message is output to the system console when a batch job pauses. The system operator may then take appropriate action.

2.4.18 Default User Volume (SGN.MVOL) Option

The default user volume (SGN.MVOL) option defines the default volume for an MTM user. The standard MTMPARMS file defines the system volume as the default user volume. To specify a default volume, edit this option to the following:

Format:

```
SGN.MVOL EQU C'name'
```

Parameters:

name is the name of a disk volume to be used as the default user volume for all users at sign-on time. This name must be four characters in length. The first character must be alphabetic, and the remaining alphanumeric. If less than four characters are used, the remaining characters must be blanks, e.g., C'MTM'. Four blank characters indicate that the default user volume is the system volume.

2.4.19 Batch Job Relative Priority (SGN.PRIO) Option

The batch job relative priority (SGN.PRIO) option defines the relative default priority of a batch job running under MTM. To modify the standard value, edit this option to the following:

Format:

SGN.PRIO EQU n

Parameter:

n is a decimal number from 0 through 20 that is added to the MTM default priority (MTMdp) of an interactive job to determine the absolute priority at which an MTM user's batch job will run. For example, the standard value of 6 indicates that a batch job will run at six priorities lower than an interactive job unless otherwise specified. If 0 is specified, 0 is added to the MTMdp of an interactive job. This indicates that a batch job will run at the same priority as an interactive job unless otherwise specified.

Functional Details:

Default priorities for interactive and batch jobs are calculated as follows:

| TYPE OF JOB | PRIORITY |
|-------------|---|
| MTM | MTM link priority |
| Interactive | MTMdp = MTM link priority + 12 |
| Batch | MTMdp + SGN.PRIO = MTM link priority + 12 + n |

Once these default priorities are established, any priorities set by an MTM user in the SUBMIT command are checked against the default priorities. A batch job is always run at the lowest possible valid priority and is never run at the same priority as MTM. Default priorities for interactive jobs are not affected by this option.

Example:

If:

```
MTM link priority = 128
SGN.PRIO          = 10
```

the following become the MTM default priorities of interactive and batch jobs:

```
Interactive job = 140
Batch job      = 150
```

As shown in the following examples, priorities specified by an MTM user in the SUBMIT command are checked against the default priorities established at MTM sysgen.

Example:

Assume the following SUBMIT commands are entered by an MTM user (SGN.PRIO=10):

```
SUBMIT TEST.JOB,PR=138
SUBMIT TEST1.JOB,PR=150
```

The priority for TEST.JOB is invalid because the user specified that the submitted job be run at a higher priority than the MTMdp of 140. The job will be run at 150.

The priority for TEST1.JOB is valid because it is lower than the MTMdp of 139.

The batch job TEST1.JOB will be run at the specified priority of 150.

Example:

Assume the following SUBMIT commands are entered by an MTM user. The jobs will be run at either the MTMdp for batch jobs or the Link priorities, whichever are lower. (SGN.PRIO=6)

| JOB | LINK PRIORITY |
|---------------------------|------------------|
| SUBMIT TEST.JOB | 128 |
| SUBMIT TEST1.JOB | 200 |
| SUBMIT TEST2.JOB,PRIO=160 | 180 |

TEST.JOB will be run at the batch priority of 140, which is lower than the link priority of 128.

TEST1.JOB will be run at the link priority of 200, which is lower than the batch priority of 146.

TEST2.JOB will be run at the link priority of 180, which is lower than both the submit priority of 160 and the batch priority of 146.

2.4.20 Forced Roll (SGN.ROLL) Option

In most operating system environments, it is desirable to make every task running under MTM rollable. With the forced roll option (SGN.ROLL) enabled, every task, except executive tasks (e-tasks) loaded by MTM, is made rollable at load time. The standard MTMPARMS file enables forced roll. To disable forced roll, edit this option to the following:

Format:

```
SGN.ROLL EQU 0
```

Functional Details:

This option applies only to the private image segment of the task. Shared segments are not rolled. See the OS/32 Application Level Manual for a description of the task image segments established by Link.

2.4.21 Multi-Terminal Monitor (MTM) Scheduling (SGN.SCHR) Option

The MTM scheduling option (SGN.SCHR) allows the user to select one of three scheduling techniques. The scheduling techniques available through SGN.SCHR are OS/32 priority scheduling, dynamic priority scheduling, and the MTM priority scheduling mechanism (PSM). Dynamic priority scheduling is specified by the standard MTMPARMS file. To modify the standard value, edit this option to the following:

Format:

SGN.SCHR EQU n

Parameter:

n may be 0, 1, or 2.

Functional Details:

When 0 is selected for n, no special task scheduling is done for MTM subtasks. The standard task scheduling of OS/32 is used; i.e., tasks are run according to their priority and the timeslice value from the SET SLICE command. Tasks of equal priority are scheduled on a first-in/first-out (FIFO) basis. See the OS/32 Application Level Programmer Reference Manual for more information on OS/32 priority scheduling.

When 1 is selected for n, dynamic priority and timeslice scheduling is activated for MTM subtasks. Through dynamic scheduling a task's priority is adjusted through a range of 13 priorities, while a task's timeslice is adjusted through a range from 20 milliseconds to 2 minutes. Computation intensive tasks (i.e., tasks that consume their timeslice without an interrupt) are adjusted to the next lower priority and their timeslice is doubled. I/O intensive tasks (i.e., tasks that are interrupted before they consume their timeslice) are adjusted to the next higher priority and their timeslice is halved. Dynamic scheduling of MTM subtasks by OS/32 ensures that tasks which require interactive terminal I/O are adjusted to the highest allowable priority.

When 2 is selected for n, the MTM priority scheduling mechanism (PSM) is activated for MTM subtasks. PSM increases system throughput by adjusting the priorities of MTM subtasks according to their run-time priorities. PSM performs a periodic check on the run-time behavior of all MTM subtasks. Based on this behavioral information, PSM adjusts the priorities of subtasks whose current priorities are in the range of *+12 to *+128, where * equals the task priority for .MTM.

All current subtask priorities outside this range and those priorities set by the PRIOR command or the SUBMIT command at load time are fixed. Fixed priorities cannot be modified by PSM. In addition, PSM cannot alter the maximum batch priority set by the SGN.PRIO option. Table 2-1 summarizes task priority modification when PSM is enabled.

PSM lowers the task priorities of subtasks requiring a relatively large amount of CPU time, while PSM raises the priorities of subtasks that perform a large number of I/O operations. PSM will not raise a task's priority above the priority set at link time (except when the task is rolled).

All priorities assigned by PSM (except for rolled-out tasks) must be within the *+12 to *+128 range. To increase the chances of entry into the system, PSM gradually raises the priority of a rolled-out subtask to *+1. However, once the task is rolled-in, PSM adjusts its priority within the *+12 to *+128 range.

TABLE 2-2 MODIFICATION OF MTM SUBTASK PRIORITIES WITH PSM ENABLED

| PRIORITY | DEFAULT TASKS | ALLOWABLE PRIORITY MODIFICATION |
|------------------------------|--------------------------|--|
| * | .MTM .MTMASST .SPL | Fixed priority - cannot be modified by PSM. |
| * +1 . . . * +11 | None | Fixed priority range - can be modified by PRIOR command or a SUBMIT command issued by a user with the privilege to alter batch priorities. |

TABLE 2-2 MODIFICATION OF MTM SUBTASK PRIORITIES WITH PSM ENABLED (Continued)

| PRIORITY | DEFAULT TASKS | ALLOWABLE PRIORITY MODIFICATION |
|--------------------------------------|----------------------------|--|
| * +12 | All MTM inter-active tasks | Subtasks whose priorities lie within this range can be modified by PSM according to their run-time behavior priorities for all tasks (except rolled-out tasks) adjusted by PSM must remain within this range. PSM raises or lowers task priorities for batch jobs as it does for inter-active tasks except PSM cannot raise a batch job priority higher than the maximum set at MTM sysgen. (See SGN.PRIO). NOTE Priorities in this range that have been set by the PRIOR or SUBMIT commands are fixed. These priorities cannot be adjusted by PSM. |
| * +13 | All batch MTM jobs | |
| . | | |
| . | | |
| . | | |
| . | | |
| . | | |
| . | | |
| . | | |
| . | | |
| . | | |
| . | | |
| . | | |
| . | | |
| . | | |
| * +128 or 248 (whichever is smaller) | | |
| * +129 or 249 (whichever is smaller) | None | Fixed priority range - can be modified by PRIOR command only. |
| . | | |
| . | | |
| 255 | | |

See the Multi-Terminal Monitor (MTM) Reference Manual for more information on the PRIOR command.

2.4.22 SPL/32 Interface (SGN.SPL) Option

The SPL/32 interface (SGN.SPL) option gives MTM users access to the SPL/32 software. SPL/32 must be configured into the system during OS/32 sysgen; otherwise, this option has no affect on the system. The standard MTMPARMS file does not enable the SPL/32 interface. To enable SPL/32 support, and specify the desired number of send data buffers, edit this option to the following:

Format:

```
SGN.SPL EQU n
```

Parameter

n is the number of send data buffers.

Functional Details:

When 0 is selected for n, SPL/32 support is disabled. If SPL/32 support is desired, a recommended value for n is NUSERS.

2.4.23 Global and Local Variables (SGN.VAR) Option

The global and local variables (SGN.VAR) option specifies the maximum number of global and local variables that an MTM user can define. The maximum number specified by the standard MTMPARMS file is 28. To modify the standard value, edit this option to the following:

Format:

SGN.VAR EQU n

Parameter:

n is a decimal number from 28 through 1,024 specifying the combined maximum number of global and local variables a user can define. If n equals 0, variable support is not included in MTM.

2.4.24 Number of New Global Variables (SGN.@GLO)

The number of new global variables (SGN.@GLO) option specifies the maximum number of new global variables that can be defined by an MTM user via the \$DEFINE command. The maximum number of new global variables specified by the standard MTMPARMS file is 15. To modify the standard value, edit this option to the following:

Format:

```
SGN.@GLO EQU n
```

Parameter:

n is a decimal number from 0 through 99 specifying the maximum number of new global variables allowed.

Functional Details:

Each new global variable defined by a user occupies one command buffer. Therefore the number specified for this option should be considered when specifying the NCSBFS option.

2.6.25 Number of New Internal Variables (SGN.@INT) Option

The number of new internal variables (SGN.@INT) option specifies the maximum number of new internal variables that can be defined in an MTM user's CSS command file via the \$DEFINE command. The maximum number of new internal variables specified by the standard MTMPARMS file is 8. To modify the standard value, edit this option to the following:

Format:

```
SGN.@INT EQU n
```

Parameter:

n is a decimal number from 0 through 99 specifying the maximum number of new internal variables allowed in a CSS command file.

Functional Details:

Each new internal variable defined by a user occupies one command buffer. Therefore the number specified for this option should be considered when specifying the NCSBFS option.

2.6.26 HASP Interface (SGN.\$HSP) Option

The HASP interface (SGN.\$HSP) option allows terminal users to switch from an MTM time-sharing environment to a HASP environment. The standard MTMPARMS file does not enable the HASP interface. To enable the HASP interface, edit the SGN.\$HSP option to the following:

Format:

```
SGN.$HSP EQU n
```

Parameters:

n is a decimal number from 1 through 99 specifying the maximum number of tasks that can be concurrently interfaced with HASP by MTM users. If n equals 0, the HASP interface feature is disabled.

2.4.27 Central Processing Unit (CPU) Limit Per Accounting User Class (UCPULIM) Macro

The CPU limit (UCPULIM) per accounting user class macro defines the maximum CPU time allowed for each accounting user class. There is a maximum of 16 accounting user classes (0-15) established in the authorized user file (AUF). Each account is assigned to one of the established accounting user classes. A terminal session or batch job is limited to the CPU time designated for the class to which the account belongs. To define CPU limits for each user class, first enable the OS/32 accounting facility using the SGN.ACT option. Then, enter the UCPULIM macro as follows:

Format:

```
[label] UCPULIM limit0t,limit0b,limit1t,limit1b,  
        .....  
        limitnt,limitnb
```

Operands:

limit_nt is a decimal number ranging from 0 through 86,400 (one day) indicating the maximum number of seconds of CPU time allowed any terminal session for a user signed on to an account in user class n. If limit_nt equals 0, no CPU limit applies to terminal sessions under class n.

limit_nb is a decimal number ranging from 0 through 86,400 (one day) indicating the maximum number of seconds of CPU time allowed any batch job running under an account in user class n. If limit_nb equals 0, the default CPU time limit is the limit set by the CPULIMIT option at MTM sysgen.

Functional Details:

If execution of a task under any account exceeds the CPU limit set for the accounting user class to which the account belongs, the task is terminated (end of task code 1200). After termination, no other tasks can be loaded under that account, and all batch jobs under that account are aborted.

Example:

UCPULIM 30,30,,,40,3600,80,10800

In the above example, the CPU limit for both batch jobs and terminal users in accounting user class 0 is 30 seconds. Accounting user class 1 has no CPU limit for terminal users; batch jobs are restricted by the limit set in CPULIMIT. For user class 2, the CPU limit for terminal users is 40 seconds, while batch jobs are limited to 1 hour. Class 3 users are limited to 80 seconds for terminal users and 3 hours for batch jobs. For the remaining classes, no CPU limits are set for terminal sessions; batch-jobs running under these classes are limited to the value set by the CPULIMIT option.

2.4.28 Input/Output (I/O) Count Limit Per Accounting User Class (UIOLIM) Macro

The I/O count limit per accounting user class (UIOLIM) macro defines the maximum number of I/O transfers, within each I/O class, allowed for each task run either from a terminal or batch stream for a given user class (0-15). To define I/O count limits for each user class, first enable the OS/32 accounting facility using the SGN.ACT option, and establish I/O classes for MTM using the IOTAB macro. Then, enter UIOLIM as follows:

Format:

```
[label] UIOLIM (u0io0t, ..., u0iont), (u0io0b, ..., u0ionb)  
          .....  
          (unio0t, ..., uniont), (unio0b, ..., unionb)
```

Operands:

u_{niont} is a decimal number ranging from 0 through 1,000,000 indicating the maximum number of I/O transfers to the device assigned to I/O class n for all interactive tasks run from a terminal for user class n. If u_{niont} for a given user class is omitted, or equals 0, the I/O count limits specified for each I/O class by the IOTAB macro are in effect for that class.

u_{nionb} is a decimal number ranging from 0 through 1,000,000 indicating the maximum number of I/O transfers to the device assigned to I/O class n for all batch jobs for user class n. If u_{nionb} for a given user class is omitted, or equals 0, the I/O count limits specified for each I/O class by the IOTAB macro are in effect for that class.

Functional Details:

If a task attempts to exceed an I/O count limit set for the user class to which its account belongs, task execution is terminated with an end of task code 1210. All outstanding jobs corresponding to the I/O count limit exceeded are aborted.

Because the IOTAB macro defines the I/O classes for MTM, the IOTAB macro must precede UIOLIM in the MTM parameter file.

Interactive I/O counts for all I/O classes must be enclosed in parentheses; a comma is then used to separate the interactive I/O counts from the batch I/O counts (also enclosed in parentheses) for the same user class. Similarly, a comma following the batch I/O counts for all I/O classes is used to separate the parameters for one user class from those of the next user class.

The following example is designed to shed some light on the format of the UIOLIM macro and its relationship to the IOTAB macro.

```

IOTAB (0,INDX),(1,CNTG),(2,DEV),(3,SPL),(4,TAPE)

UIOLIM (00,01,02,03,04),(100,101,102,103,104), U CLS 0 *
        (10,11,12,13,14),, U CLS 1 *
        (20,21,22,23,24),(120,121,122,123,124), U CLS 2 *
        (30,31,32,33,34),(130,131,132,133,134) U CLS 3 *

      I C D S T      I C D S T
      N N E P A      N N E P A
      D T V L P      D T V L P
      X G           X G           E
    └──────────┘    └──────────┘
      TERMINAL      BATCH
  
```

Table 2-3 illustrates the I/O count limits set for batch and terminal jobs by the above example of UIOLIM.

TABLE 2-3 EXAMPLE I/O COUNT LIMITS SET BY UIOLIM

| ACCOUNTING USER CLASS | I/O CLASS | | | | | | | | | |
|--------------------------|-----------|-------|----------|-------|---------|-------|---------|-------|----------|-------|
| | 0 (INDX) | | 1 (CNTG) | | 2 (DEV) | | 3 (SPL) | | 4 (TAPE) | |
| | TERM. | BATCH | TERM. | BATCH | TERM. | BATCH | TERM. | BATCH | TERM. | BATCH |
| 0 | * | 100 | 1 | 101 | 2 | 102 | 3 | 103 | 4 | 104 |
| 1 | 10 | ** | 11 | ** | 12 | ** | 13 | ** | 14 | ** |
| 2 | 20 | 120 | 21 | 121 | 22 | 122 | 23 | 123 | 24 | 124 |
| 3 | 30 | 130 | 31 | 131 | 32 | 132 | 33 | 133 | 34 | 134 |
| 4 | * | ** | * | ** | * | ** | * | ** | * | ** |

LEGEND

- * no I/O count limit.
- ** I/O count limit set by IOTAB macro.

NOTE

The actual I/O count limits used in the previous example are for illustrative purposes only and should not be construed as recommended parameters to the UIOLIM macro.

2.4.29 Multi-Terminal Monitor (MTM) Version Identification (VERSION) Macro

The MTM version identification (VERSION) macro assigns an identification name or number to the current revision of the MTM system. To specify an identification name for an MTMPARMS file, edit this option to the following:

Format:

```
[label] VERSION 'id'
```

Parameter:

id is a 16-character alphanumeric string identifying the current revision of the MTM system. Special characters are allowed. If blanks are used, identification name must be enclosed in quotes.

Example:

```
VERSION '3230-B'
```

2.5 MEMORY REQUIREMENTS FOR SYSTEM GENERATION (SYSGEN) OPTIONS

The MTM sysgen options reside in data structures built within the MTM task address space. Table 2-4 lists each data structure and its size.

TABLE 2-4 DATA STRUCTURES BUILT BY MTM SYSGEN

| DATA STRUCTURE | MEANING | SIZE (IN BYTES) |
|---------------------------|---|---|
| Sysgen parameter area | Contains the sysgen parameter values to be used by the MTMMAIN module | 128 |
| Terminal user block (TUB) | One is built for each concurrent MTM user defined by NUSERS option | $584 + (NCSLVS + 1) * 8 + CSBFSZ + 4 + SGN.ACT * 4 + 4$ |
| Command buffer area | Contains the command buffers | $(3 * NUSERS + NCSBFS) * (CSBFSZ + 4)$ |
| Special console buffers | Contains the pool of special console buffers | $NCONBFS * (CONBFSZ + 4)$ |
| Message buffer area | Contains the message buffers | $76 * NMSGs$ |
| Sector buffer area | Contains the sector buffers | $(256 * CDBUFSIZ + 256) * NSECTS$ |
| Circular list area 1 | Contains the MTM task queue | $4 * QUESIZE + 8$ |
| Circular list area 2 | Command buffer wait queue | $4 * NUSERS + 8$ |
| Circular list area 3 | Console buffer wait queue | $4 * NUSERS + 8$ |
| Circular list area 4 | Sector buffer wait queue | $4 * NUSERS + 8$ |

TABLE 2-4 DATA STRUCTURES BUILT BY MTM SYSGEN (Continued)

| DATA STRUCTURE | MEANING | SIZE (IN BYTES) |
|--------------------------|--|---|
| Circular list area 5 | lu wait queue | 4*NUSERS+8 |
| Circular list area 6 | Message buffer wait queue | 4*NMSGs+8 |
| Accounting write buffer | Used by MTM to write accounting information to the accounting transaction file (ATF) | 128 or 8*classes+84 (whichever is larger) |
| Command buffer usage | Used to accumulate command buffer usage for .MTM STATUS,USED command | (3*NUSERS+NCSBFS)*4 |
| I/O class table | Contains the I/O accounting class name and limits. This table is built by the IOTAB macro | 8*SGN.ACT |
| CPU limit table | Contains the CPU limits for terminals and batch jobs for each user class. Built by the UCPULIM macro | 128 |
| I/O limit table | Contains the I/O limits for terminals and batch jobs for each I/O class by user class. Built by the UIOLIM macro | 128*SGN.ACT |
| SPL/32 send data buffers | Contains the mnemonic table, send data buffer queue, and, send data buffers used by MTM and SPL/32 | 260*SGN.SPL+532 |

CHAPTER 3 AUTHORIZED USER UTILITY

3.1 INTRODUCTION

Prior to loading and starting the multi-terminal monitor (MTM), an installation must establish valid account numbers and passwords for its authorized users. This is accomplished through the authorized user utility supplied with the MTM software package. The authorized user utility maintains a file containing information on each account number. This file is called the authorized user file (AUF).

3.2 AUTHORIZED USER FILE (AUF)

The AUF contains one record for each account. An account number, group number, and password are required for each account. In addition, a user identification, limits for signon and CPU time, and privileges for each account may be optionally specified in the AUF. Defaults are set by the utility for all optional information not specified by the installation. The record format for the AUF is shown in Figure 3-1.

| | | | | |
|---------|-------------------------------|--|------|----------------------|
| 0(0) | ACCOUNT NUMBER | | 2(2) | GROUP ACCOUNT NUMBER |
| 4(4) | | | | |
| 8(8) | PASSWORD (12 characters) | | | |
| 12(C) | | | | |
| 16(10) | | | | |
| 20(14) | | | | |
| 24(18) | USERID (20 characters) | | | |
| 28(1C) | | | | |
| 32(20) | | | | |
| 36(24) | SIGNON TIME SINCE LAST REPORT | | | |
| 40(28) | RESERVED | | | |
| 44(2C) | | | | |
| 48(30) | DATE OF LAST REPORT | | | |
| 52(34) | TOTAL CPU TIME | | | |
| 56(38) | SIGNON TIME LEFT (seconds) | | | |
| 60(3C) | CPU TIME LEFT (milliseconds) | | | |
| 64(40) | PRIVILEGES | | | |
| 68(44) | | | | |
| 72(48) | RESERVED | | | |
| 76(4C) | | | | |
| 80(50) | | | | |
| 84(54) | | | | |
| 88(58) | FILLER | | | |
| | | | | |
| | | | | |
| 120(78) | | | | |
| 124(7C) | FILLER | | | |

Figure 3-1 Authorized User File Record Format

The authorized user utility is used to create, update, and report on the AUF.

Initially, the utility should be run from the system console to create an AUF. In order to enable access under MTM, the AUF must be renamed to account 255.

Example:

```
RENAME USERS.AUF,USERS.AUF/255
```

Once the AUF is renamed to account 255, it can be updated while MTM is running by signing on to account 255 and running the utility.

When the AUF is assigned to account number 255, it is protected in two ways:

- only users signed-on to account number 255 can access the AUF, and
- the authorized user utility does not display the password to account 255 in a listing.

When the LIST command of the authorized user utility is entered, the account number, group account number, user identifier (userid), signon time remaining, CPU time remaining, account privileges, and date are displayed. Once added to an account record, the password cannot be retrieved from the AUF, but can be changed by the CHANGE command.

Accounting limits and privileges can be specified for each account number through the following AUF commands:

- ADD command (create or update mode) when adding a new account to the AUF, or
- CHANGE command (update mode) when making changes to an already existing account.

Two resources can be limited by account number:

- Signon time
- CPU time

Signon time can be limited for each account by specifying the maximum time the total number of terminal users assigned to that account can be signed on to the system. Whenever a terminal user signs off the system, these limits are checked. If the terminal user has used all of the remaining signon time specified for that account, or tries to sign on to an account with no signon time left, the signon request is rejected and this message is displayed:

NO TIME LEFT

CPU time can be limited for each account by specifying the maximum time the total number of terminal users assigned to that account can use the CPU. Whenever a terminal user's task terminates, these limits are checked. If the terminal user has used all of the remaining CPU time specified for that account, this message is displayed:

YOUR CPU TIME HAS EXPIRED

If a terminal user tries to load a task assigned to an account with no CPU time left, this message is displayed:

TIME-ERR

3.3 BUILDING AND STARTING THE AUTHORIZED USER UTILITY

The authorized user utility supplied with the package has been linked according to the sequence of Link commands shown below. User involvement is only necessary when additional Link options are required. In that case, the user must specify the necessary option and re-link the task. See the OS/32 Link Reference Manual.

```
L .BG,LINK
T .BG
ST
TITLE ACTUTY
ESTABLISH TASK
MAP ,ADDR
INCL ACTUTY.OBJ
BUILD ACTUTY
END
```

The following starts the authorized user utility.

Format:

$$\text{START, } \left\{ \begin{array}{l} \text{COMMAND} \\ \text{CMD} \end{array} \right\} = \text{fd}_1, \text{LIST} = \text{fd}_2, \left\{ \begin{array}{l} \text{CREATE} \\ \text{UPDATE} \\ \text{REPORT} \end{array} \right\} = \text{fd}_3 \text{ [, ACCOUNTS} = \text{n}]$$

Parameters:

CMD= fd₁ indicates the device or file from which command input will come. The authorized user utility assigns the command input device to logical unit 1 (lu1).

LIST= fd₂ indicates the device or file to which the output is sent. The authorized user utility assigns the list device to lu3.

CREATE
UPDATE=
REPORT specifies a mode of operation for the utility. The AUF to be created, updated, or about which a report is needed, is specified by fd₃. The authorized user utility assigns the specified AUF to lu3.

ACCOUNTS=n n is a decimal number (0-65,535) indicating the maximum number of accounts allowed in the AUF file specified by fd₃. In the CREATE mode, n indicates the size of the new AUF. In the UPDATE mode, n indicates the size to which the AUF is to be extended. If the ACCOUNTS parameter is not specified, the maximum number of accounts in the AUF defaults to 255 or the maximum set by a previous ACCOUNT command. This parameter is ignored when AUF is operating in the REPORT mode.

Functional Details:

When the CREATE mode is selected, the authorized user utility allocates, assigns, and initializes the new file specified by fd₃. If the file specified by fd₃ already exists, an error message is generated and the authorized user utility goes to end of task. If the UPDATE mode is selected, an update to the file specified by fd₃ is requested. If the REPORT mode is selected, a report of the file specified by fd₃ is requested. If the specified file does not exist for the UPDATE and REPORT modes, an error message is generated, and the utility goes to end of task.

The authorized user utility assigns all logical units based on the parameters specified in the START command. These starting parameters can be entered in any order. The user cannot assign specific logical units.

Examples:

The following example creates an AUF named NEWFILE. The system console (CON:) is the command input device and the line printer (PR:) is the list device. Until the START command is issued, the file (NEWFILE) does not exist. The authorized user utility allocates, assigns, and initializes the new file for accounts 1 to 511.

```
START,CMD=CON:,LIST=PR:,CR=NEWFILE,AC=511  
ST B, CR=XX(NDEXT) CMD=CON:, LI=CON:, AC=XX
```

The following example specifies the console (CON:) as both the command input device and the list device. The mode of operation is UPDATE an existing file called AUFFILE.

```
START,UP=AUFFILE,CMD=CON:,LIST=CON:
```

3.4 AUTHORIZED USER UTILITY COMMANDS

The authorized user utility commands are:

- ACCOUNTS
- ADD
- CHANGE
- DELETE
- END
- LIST
- MODE
- PAUSE
- PRIVILEGE
- TOTAL

The utility has three modes of operation:

- CREATE
- REPORT
- UPDATE

Each mode allows certain authorized user utility commands to be executed. See Table 3-1.

The create mode is used to establish a new AUF.

The report mode allows an account record to be listed or an accounting report for each active account (an account record already assigned whether used or not) to be generated.

The update mode allows account records to be added, deleted, modified, or listed.

TABLE 3-1 AUTHORIZED USER UTILITY
MODES AND ALLOWANCE
COMMANDS

| COMMANDS | MODES | | |
|-----------|--------|--------|--------|
| | CREATE | REPORT | UPDATE |
| ACCOUNTS | * | ** | * |
| ADD | * | | * |
| CHANGE | * | | * |
| DELETE | * | | * |
| END | * | * | * |
| LIST | * | * | * |
| MODE | * | * | * |
| PAUSE | * | * | * |
| PRIVILEGE | * | ** | * |
| TOTAL | | * | |

LEGEND

* Command can be executed in specified mode

** Display (no operand) form only

| ACCOUNTS |

3.4.1 ACCOUNTS Command

The ACCOUNTS command raises or lowers the account number limit for the AUF file. This command also displays the current limit set for the file.

Format:

ACCOUNTS n

Parameters:

n is a decimal number from 1 to 65,535 indicating the account number limit of the AUF. If n is omitted, the current account number limit is displayed. If the value specified for n is illegal, the ACCOUNTS command is ignored.

Functional Details:

In the REPORT mode, this command can be used in display form only. The authorized user utility sets the account limit in blocks of 256 records - one account per record. If n does not specify the last record in a 256 record block (e.g., 255, 1023, etc.), the value of n is raised to the next multiple of 256. For example, if n=100, the account limit is set to 255.

After the ACCOUNTS command is executed, the following message is sent to the console:

ACCOUNT LIMIT = n

The variable n is the current maximum account number to which the AUF has been raised or lowered. If the ACCOUNTS command is entered without a value specified for n, this message will display the current account number limit in the AUF.

Examples:

ACCOUNT
ACCOUNT LIMIT = 2047

ACC 1000
ACCOUNT LIMIT = 1023

| ADD |

3.4.2 ADD Command

The ADD command establishes an account to the AUF.

Format:

ADD actno,groupno,password, [userid] [,{ signontime }] [,{ CPUtime }]
 [,privileges]

Parameters:

| | |
|------------|---|
| actno | is a decimal number from 1 through the current account limit specifying the account number to be established. |
| groupno | is a decimal number from 1 through the current limit (except 255) specifying a group account number to which the private account number is to be assigned. Group number 255 is valid only as a group number for account 255. |
| password | is a 1- to 12-character alphanumeric string specifying the password to be established for the account. |
| userid | is a 1- to 20-character alphanumeric string specifying the account user's identification. |
| signontime | is a decimal number specifying the number of seconds of signon time to which the account is limited. If an asterisk (*) is specified, no limit is applied. For convenience, this parameter may be entered in any of the following formats: hh:mm:ss, mm:ss, or seconds; for example, 1:30:10, 90:10, and 5410 all represent the same limit. |
| CPUtime | is a decimal number specifying the CPU time to which the account user is limited. If this parameter is omitted, the default established at MTM system generation (sysgen) is used. If an asterisk (*) is specified, no limit is applied. This parameter may be entered in the same formats as signontime. |

privileges is a hexadecimal number specifying the privileges to be allowed for users signed on under this account. If this parameter is not specified, the default privileges 0000A800 (RENAME, ENVIRONMENT AT SIGNON, and PRIVATE CSS EXECUTION) are assigned to the account. See Section 3.4.9 for information on account privileges and their corresponding hexadecimal codes.

Functional Details:

This command cannot be used in REPORT mode. The account added to the AUF must be inactive. If the account number is active, an error message is generated and the existing record is not changed. The utility mode must be CREATE or UPDATE. A copy of the record is written to the log device.

Account 255 must be added when the AUF is first created. A warning results if account number 255 does not exist when the utility is terminated.

The privileges for account 255 are set to FFFFBFFF regardless of the privileges specified when account 255 is established.

Example:

The following ADD command adds account number 1000; the group number is 250; the password is RAY; the userid is DEPT 2; the signon time limit is 9999:59:59; no CPU time limit is set; the privileges in effect are SVC6 COMMUNICATION, CHANGE PASSWORD, PRIVATE CSS EXECUTION, TEMPORARY and SPOOL FILES, ENVIRONMENT AT SIGNON, and RENAME.

```
ADD 1000,250,RAY,DEPT 2,9999:59:59,*,8000F800
1000 250 DEPT 2 9999:59:59 ** 8000F800 03/03/83
```

| CHANGE |

3.4.3 CHANGE Command

The CHANGE command changes fields in the AUF record of an active account.

Format:

CHANGE actno, [groupno], [password], [userid] [, { signontime }]
[, { CPUtime }] [, privileges]

Parameters:

- actno is a decimal number from 1 through the current account limit specifying the account number to be changed.
- groupno is a decimal number from 1 through the current account limit (except 255) specifying the group number to which the account is to be assigned. Group number 255 is valid as a group number for account 255.
- password is a 1- to 12-character alphanumeric string specifying the password for the account.
- userid is a 1- to 20-character alphanumeric string specifying the user's identification for the account.
- signontime is a decimal number specifying the additional signon time to which a terminal user is limited. If an asterisk (*) is specified, the currently established limit is changed and no limit is applied. For convenience, this parameter may be entered in any of the following formats: hh:mm:ss, mm:ss, or seconds; for example, 1:30:10, 90:10, and 5410 all represent the same limit.

CPUtime is a decimal number specifying the additional CPU time to which the terminal user is limited. If an asterisk (*) is specified, the currently established limit is changed and no limit is applied. This parameter may be entered in the same format as signontime.

privileges is a hexadecimal number specifying the privileges to be allowed for users signed on under this account. See Section 3.4.9 for information on the account privileges and their corresponding hexadecimal codes. The privileges for account 255 default to FFFF BFFF.

Functional Details:

This command cannot be used in REPORT mode. As each record is changed, a copy of the original and resultant record is sent to the list device.

The values of signontime and CPU time are added to the current signontime and CPU time limits in the AUF to increase the time allocations. To decrease the time allocation, precede signontime or CPU time with a minus sign (-). An asterisk (*) causes the limit to be disabled.

Example:

The following CHANGE command changes the CPU time limit for account number 1000 from unlimited CPU time to a limit of 1:30:10.

```
CH 1000,,,,,5410
  1000 250 DEPT 2          9999:59:59  **      8000F800 03/03/83
  1000 250 DEPT 2          9999:59:59  1:30:10  8000F800 03/03/83
```

```
-----  
|  DELETE  |  
-----
```

3.4.4 DELETE Command

The DELETE command makes an active account inactive.

Format:

```
DELETE actno
```

Parameter:

actno is a decimal number from 1 through the current account limit (except 255) specifying the account number to be made inactive.

Functional Details:

This command cannot be used in REPORT mode. A copy of the deleted record is written to the list device.

Examples:

The following DELETE command deletes account number 207.

```
DEL 207  
207 205 308503154TEMP SHOBER** ** A800 02/18/83
```

The following command attempts to delete an inactive file.

```
DEL 207  
*** ACCOUNT NUMBER 207 WAS INACTIVE
```

| END |

3.4.5 END Command

The END command closes all logical units and terminates the utility.

Format:

END

Functional Details:

If the utility is ended and account 255 does not exist, the utility issues a warning that the user has created an AUF that cannot be accessed later.

Example:

```
END
** WARNING ACCOUNT 255 DOES NOT EXIST
LAURIE -END OF TASK CODE= 0 CPUTIME=1.463/2.450
```

| LIST |

3.4.6 LIST Command

The LIST command displays active account records to the list device.

Format:

LIST [actno₁ [, actno₂]]

Parameters:

actno₁ is a decimal number from 1 through the current account limit specifying the account numbers to be displayed to the list device. If specified with actno₂, it specifies the beginning of a range of account numbers to be displayed to the list device.

actno₂ is a decimal number from 1 through the current account limit specifying the end of a range of account numbers to be displayed to the list device.

Functional Details:

When no account number is specified, all active accounts are listed. When only actno₁ is specified, only actno₁ is listed. When two account numbers are specified, all account numbers from the first account number to the second account number are listed.

Examples:

In the following example, a single account is listed.

```
LIST 1000
  ACT  GRP NAME          TIME LEFT    CPU LEFT    PRIV    DATE
  1000 250 DEPT 2      9999:59:59  **         8000F800 03/03/83
```

In the following example a range of accounts is listed.

LIST 200,1000

| ACT | GRP | NAME | TIME LEFT | CPU LEFT | PRIV | DATE |
|------|-----|----------------------|------------|----------|----------|----------|
| 201 | 205 | 308503154TEMP SHOBER | ** | ** | 0000A800 | 02/18/83 |
| 207 | 205 | 308503154TEMP SHOBER | ** | ** | 0000A800 | 02/18/83 |
| 211 | 205 | 308503154TEMP SHOBER | ** | ** | 0000A800 | 02/18/83 |
| 230 | 230 | 035713087CAROL CARCH | ** | ** | 0000A800 | 02/18/83 |
| 1000 | 250 | DEPT 2 | 9999:59:59 | 1:30:10 | 8000F800 | 03/03/83 |

| MODE |

3.4.7 MODE Command

The MODE command displays the current mode, changes the current mode of the utility, or reassigns or creates an AUF.

Format:

$$\text{MODE} \left[\begin{array}{c} \text{(CREATE)} \\ \text{UPDATE} \\ \text{(REPORT)} \end{array} \right] [=fd_3]$$

Functional Details:

The MODE command, with no parameters, displays the current mode. When a mode is specified, the mode is reassigned but the AUF is not changed. When fd_3 is specified, the mode is changed and the AUF is reassigned.

When the create mode is specified, an error message is generated if fd_3 exists. The authorized user utility allocates the new AUF and initializes it for the number of account records currently in effect.

Examples:

To place the utility in the REPORT mode:

```
>MODE REPORT
```

To display the current mode:

```
>M  
***     REPORT IS THE CURRENT MODE
```


3.4.8 PAUSE Command

The PAUSE command pauses the utility and transfers control to the operating system. If the utility is running under MTM, control is transferred to MTM. The CONTINUE command returns control to the utility.

Format:

PAUSE

| PRIVILEGE |

3.4.9 PRIVILEGE Command

The PRIVILEGE command can be used to change or display the privileges for an account or range of accounts.

Format:

PRIVILEGE [actno₁ [actno₂] [{ (privilege) } *]]

Parameters:

- actno₁ is a decimal number from 1 through the current account limit, specifying the account number, or the start of a range of account numbers, of which the current privileges are to be changed or displayed. If this parameter is omitted, the default is all currently active accounts.
- actno₂ is a decimal number, from (actno₁ + 1) through the current account limit, specifying the final account number of the range to be changed or displayed.
- privilege is the hexadecimal mask of the privileges to be assigned to the specified account or range of accounts. If * is specified, the default privilege 0000A800 is used. If this parameter is omitted, the current privileges of the specified account or range of accounts are displayed.

TABLE 3-2 MTM ACCOUNT PRIVILEGES

| HEX MASK | PRIVILEGE AND DESCRIPTION | BIT |
|-----------|---|-----|
| 8000 0000 | SVC6 COMMUNICATION allows the user to load a task with task communication enabled. If a user without this privilege loads a task with the SVC6 communication option, MTM disables the option when the task is loaded. | 0 |
| 0800 0000 | ETASK allows the user to load executive tasks (e-tasks). If a user without this privilege attempts to load an e-task, an error message is generated and the task is not loaded. | 4 |
| 0400 0000 | SVC6 CONTROL allows a user to load tasks with SVC6 control enabled. At load time, MTM checks the task's requested options against the user's privileges. If MTM discovers that the user does not have the SVC6 control privilege requested by the task, MTM disables the option at load time. | 5 |
| 0080 0000 | EXTENDED ACCOUNT ACCESS allows the user to enter an account number, in addition to a class, in an fd. This extended access is allowed for the commands RENAME, ASSIGN, and LOAD, and for CSS calls. File access is limited to SRO for accounts other than the user's private account. Without this privilege, a user may only enter file classes in an fd; i.e., P, G, S. | 8 |
| 0040 0000 | SET GROUP allows the user to change the group account number of his account via the SET GROUP command. If a user without this privilege attempts to use the SET GROUP command, an error message is generated. The user's new group account number is retained until signoff or until changed by another SET GROUP command. The user's group number is not changed in the AUF. No user may set his group account to 255. | 9 |
| 0008 0000 | TASK ACCOUNT PRIVILEGE allows the user to load tasks with the account privilege task option enabled. If a user without this privilege attempts to load a task requesting this option, MTM disables the option. | 12 |

TABLE 3-2 MTM ACCOUNT PRIVILEGES (Continued)

| HEX MASK | PRIVILEGE AND DESCRIPTION | BIT |
|-----------|--|-----|
| 0004 0000 | BARE DISK ACCESS allows the user to access bare disks for tasks which have this option enabled. User's requesting the bare disk option without this privilege will have the option disabled by MTM. | 13 |
| 0002 0000 | UNIVERSAL TASKS allows the user to run tasks as universal tasks. If a user without this privilege attempts to run a task with the universal option enabled, MTM disables the option and th task is run as a user-task (u-task). | 14 |
| 0000 8000 | ENVIRONMENT AT SIGNON permits the use of the ENV= parameter of the signon command. If a user without this privilege attempts to use the ENV= parameter, MTM ignores the environment request and forces the USERINIT.CSS to be executed. | 16 |
| 0000 4000 | TEMPORARY and SPOOL FILES permit a user to display the filenames of temporary and spool files. If a user without this privilege attempts to display temporary or spool file fds, MTM returns *TEMPFILE* and *SPOOLFILE*, respectively. | 17 |
| 0000 2000 | PRIVATE CSS EXECUTION permits a user to execute private CSS files. A user without this privilege is limited to system CSS calls. | 18 |
| 0000 1000 | CHANGE PASSWORD permits a user to select a new password via the PASSWORD command. The new password is entered in the AUF and remains in effect until changed by another PASSWORD command or the authorized user utility. An error message is generated for user's without this privilege attempting to use the PASSWORD command. | 19 |
| 0000 0800 | RENAME permits a user to rename private files to group or system files; i.e., G, S. If a user also has the EXTENDED ACCOUNT ACCESS privilege, private files may be renamed to any account. Users without this privilege may rename files only within their private accounts. | 20 |

TABLE 3-2 MTM ACCOUNT PRIVILEGES (Continued)

| HEX MASK | PRIVILEGE AND DESCRIPTION | BIT |
|-----------|---|-----|
| 0000 0400 | PRIORITIES 1 to 11 permit a user to employ the PRIOR command, in a private CSS, to set a task's priority within the range of .MTM+1 .MTM+11. The maximum priority available to users without this privilege is .MTM+12. Priorities .MTM+1 to .MTM+11 are always available to system CSS calls regardless of a user's privileges. | 21 |
| 0000 0200 | BATCH PRIORITY allows a user to raise the priority of a batch job above the default priority. The default priority for batch tasks is .MTM+2+SGN.BPR1. Batch priorities can be raised through the use of the SUBMIT and PRIOR commands. The priority of a batch task can never be raised above the priority of interactive tasks (.MTM+12). Users may lower batch priorities regardless of their privileges. | 22 |
| 0000 0100 | <p>PRIVILEGED USER (UNLIMITED FILE ACCESS) allows a user unlimited access to all accounts (except 255). Users with this privilege may enter the SET PRIVATE command to change their account numbers. This allows users to access the files of any account as if they were the users' private files; knowledge of the account's password is unnecessary. This privilege also allows users to change their group account numbers. The user retains all privileges when moving among accounts; the privileges of the account to which a privileged user changes are ignored.</p> <p>This privilege also provides the user with the ability to display all batch jobs within an account through the INQUIRE command. Through the DISPLAY FILES command entered with an * in the account field, a user can display account numbers rather than account classes in fds.</p> | 23 |

TABLE 3-2 MTM ACCOUNT PRIVILEGES (Continued)

| HEX MASK | PRIVILEGE AND DESCRIPTION | BIT |
|-----------|---|----------|
| 0000 0080 | BACKGROUND INTERFACE allows the use of the \$BACKGROUND interface. If a user without this privilege attempts to connect to a background task, an error message is generated. | 24 |
| 0000 0040 | HASP INTERFACE allows the user to interface with HASP tasks in the system. SGN.\$HSP must be enabled at MTM sysgen to permit this priority. A user without this privilege who attempts to connect with a HASP task receives an error message. | 25 |
| 0000 000F | ACCOUNTING USER CLASSES is used to specify an accounting user's class for each account. There are fifteen (O-F) possible accounting user classes. Each account must be assigned to an accounting user class. Each accounting class corresponds to a set of CPU and I/O limits as specified by the UCPULIM and UIOLIM macros at MTM sysgen. If limits for each class are not established through UCPULIM and UIOLIM, the limits set by CPULIMIT and IOTAB apply, and all accounts effectively belong to one accounting user class. | 28 to 31 |

Examples:

In the following example, the privileges of a single account are listed.

```

PRIV 1000
  ACT  GRP      NAME      PRIV
 1000  250    DEPT 2    8000F800
    
```

In the following example, the privileges of a range of accounts are listed:

PRIV 200-1000,*

| ACT | GRP | NAME | PRIV |
|------|-----|----------------------|------|
| 201 | 205 | 308503154TEMP SHOBER | A800 |
| 211 | 205 | 308503154TEMP SHOBER | A800 |
| 230 | 230 | 035713087CAROL CARCH | A800 |
| 1000 | 250 | DEPT 2 | A800 |

In the following example, the privileges of an account are changed.

PRIV 1000,FFFFFFFF

| ACT | NAME | PRIV |
|------|--------|----------|
| 1000 | DEPT 2 | FFFFFFFF |

| TOTAL |

3.4.10 TOTAL Command

The TOTAL command generates a report to the list device or resets the values in the signontime used and CPU time used fields to 0.

Format:

TOTAL[C]

Parameter:

C is an alphabetic character specifying that the signontime used and CPU time used fields are to be cleared or reset to 0 and no report is to be generated to the list device. If this parameter is omitted, a report is generated to the list device and no values are cleared or reset to 0.

Functional Details:

The generated report lists each active account. The following details concerning each account are displayed: account number, group number, signon time left, CPU time left, and the time used since the last report was generated.

Example:

| TOTAL ACT | GRP | NAME | TIME LEFT | CPU LEFT | PRIV | TIME USED | CPU USED |
|--------------|-------------|-------------|-----------|----------|----------|-----------|----------|
| 1 | 1035213065 | DCF GROUP | ** | ** | 0000A800 | 0:00:00 | 0:00:00 |
| 10 | 18030503089 | DC CONSULTA | ** | ** | 0000A800 | 0:00:00 | 0:00:00 |
| 13 | 1035213065 | CHUCK PECK | ** | ** | 8000FCOF | 0:00:33 | 0:00:00 |
| 16 | 18031183089 | BOB DOMITZ | ** | ** | 8000FCOF | 5:49:15 | 3:03:59 |
| 18 | 18031183089 | WILL ETZ | ** | ** | 8000FCOF | 0:01:47 | 0:00:00 |
| 20 | 20035703085 | S32MT GROU | ** | ** | 8C0EFE0F | 0:00:00 | 0:00:00 |
| 23 | 1035213065 | CHRIS ARCHE | ** | ** | 8000FCOF | 0:00:00 | 0:00:00 |
| 29 | 18036243087 | S53 | ** | ** | FFFFFFFF | 0:00:00 | 0:00:00 |
| 33 | 18030493086 | N SHKOLNIKO | ** | ** | 8000FCOF | 0:26:51 | 0:00:00 |
| 34 | 20032453088 | ADMINISTRAT | ** | ** | 8C0EFE0F | 2:22:33 | 1:23:49 |
| 35 | 18032373086 | RANDEE HAFT | ** | ** | 8000FCOF | 0:00:00 | 0:00:00 |
| 36 | 20031543085 | PETE HORN | ** | ** | 8000FCOF | 0:03:28 | 0:00:00 |
| 45 | 1035203065 | JOHN GRAY | ** | ** | 8000FCOF | 0:00:00 | 0:00:00 |
| 53 | 96036243087 | LANCE PATTI | ** | ** | 8008FCOF | 0:03:10 | 0:00:00 |
| 57 | 18031203089 | SUE SELBY | ** | ** | 8000FCOF | 0:00:00 | 0:00:00 |
| 58 | 1035203065 | BILL DEIBLE | ** | ** | 8000FCOF | 0:56:13 | 0:00:00 |

CHAPTER 4 OS/32 ACCOUNTING FACILITY

4.1 INTRODUCTION

The OS/32 Accounting Facility is a software feature that is optionally specified at OS/32 system generation (sysgen). This facility provides the user with management information for computer center operation. Information is collected on the basis of account numbers and user identifiers (userid), and is reported as a formatted summary on user selected account numbers and time periods. The collected information includes such items as central processing unit (CPU) time, size of a task's segments, elapsed time, roll time, wait time, input/output (I/O) counts to I/O devices, and system space usage. Because data collection is performed by the operating system and the multi-terminal monitor (MTM), an operating system and MTM sysgen are required and should be consistent for both.

The accounting facility is composed of two parts:

- Data collection
- Reporting utility

Data collection is performed by the operating system and MTM through counting, collecting, and logging routines, which together collect and store the accounting information in a user specified disk file called the accounting transaction file (ATF).

The reporting utility is a utility program that selects and transfers accounting information to a printer, disk file or tape file. Through operator specified commands, accounting information can be used to produce cost distribution, billing, control, and information reports. See the OS/32 Systems Support Utilities Reference Manual.

4.2 SYSTEM REQUIREMENTS

The accounting facility requires these system components:

- Perkin-Elmer 32-bit processor
- 256kb memory

- operating system R05 or higher
- accounting support at OS/32 sysgen
- accounting support at MTM sysgen
- disk device
- print device
- temporary files

4.2.1 OS/32 System Generation (Sysgen)

The accounting facility is included in the operating system by specifying the sysgen ACCOUNTING and IOCLASS configuration input commands. See the OS/32 System Generation (SYSGEN/32) Reference Manual.

4.2.2 Multi-Terminal Monitor (MTM) System Generation (Sysgen)

The accounting facility is included in MTM through the parameter specified for the OS/32 accounting facility option (SGN.ACT) at MTM sysgen as described in Chapter 2.

4.2.3 Accounting Transaction File (ATF)

The ATF contains records for all activities performed by each account. The information processed by the reporting utility is provided by the ATF. The ATF is created by MTM through the ATF= parameter of the MTM START command. See Chapter 5. The ATF can also be changed by MTM through a parameter in the .MTM operator command. When the command .MTM ATF fd is entered, the current ATF is closed and a new ATF is opened. This allows data collection to continue on the new ATF while the reporting utility processes the accounting data in the old ATF. If a new ATF is not opened, an assignment error is displayed and data can no longer be collected.

4.3 LINK OPTION

Accounting support can be enabled or disabled for tasks through an optional parameter in the LINK OPTION command. See the OS/32 Link Reference Manual.

4.4 STARTING DATA COLLECTION

After OS and MTM sysgens with accounting support have been performed, MTM is started as described in Sections 5.2 and 5.3. Data collection is started by specifying the ATF= parameter of MTM START command.

4.5 DATA COLLECTION

The OS/32 accounting facility collects data for each account.

All collected data is stored in the appropriate ATF records on the basis of a job or task. Job related data is collected on the total number of tasks submitted as one job. Task related data is collected on a single task.

Job and task related data are collected and stored in specific categories which are explained in the following sections.

See Appendix D for ATF record formats.

4.5.1 Accounting Date and Time Formats

Accounting dates are specified and displayed by MTM in these formats:

- mm/dd/yy is the month, day, and year displayed on the terminal user's console when the terminal user signs off the system.
- mon/dd/yy is the month (3 characters), day, and year specified in the reporting utility SELECT, OPTION, and REPORT commands. See the OS/32 Systems Support Utilities Reference Manual.
- dd/mon/yy is the day, month (at least 3 alphabetic characters), and year specified in the reporting utility SELECT, OPTION, and REPORT commands. See the OS/32 Systems Support Utilities Reference Manual.

Accounting times are specified and displayed by MTM in these formats:

- hh:mm:ss is the hours, minutes, and seconds of elapsed time and signoff time (TIME OFF).
- hh:mm:ss.ms is the hours, minutes, seconds, and milliseconds of user CPU time, wait time, roll time, OS CPU time, maximum CPU time, and maximum signon time.

- utime is the hours, minutes, seconds, and milliseconds of user CPU time.
- ostime is the hours, minutes, seconds, and milliseconds of OS CPU time.
- maxtime is the hours, minutes, seconds, and milliseconds of maximum CPU time.
- signontime is the hours, minutes, and seconds of additional signon time.

4.5.2 Job Related Accounting Data

The accounting data collected for each job processed is:

- elapsed time,
- signon and signoff date and time,
- job number,
- date and time a job is submitted,
- total number of tasks run,
- total number of command substitution system (CSS) calls executed,
- total number of commands entered,
- total number of DISPLAY FILES commands entered,
- user identifier (userid),
- renamed and deleted files, and
- account number.

All job related accounting data is described in the following sections.

4.5.2.1 Elapsed Time

Elapsed time is the amount of time that passes between signon and signoff for a particular user. Elapsed time is displayed as a message on the terminal user's console or batch log device each time the terminal user signs off:

ELAPSED TIME = hh:mm:ss

4.5.2.2 Signon and Signoff Date and Time

The signon and signoff dates and times indicate the dates and times that the terminal user signs on and signs off the system. These dates and times are collected and stored as hexadecimal numbers in the ATF signon and signoff account records. Signon and signoff times are collected and stored in the ATF signon and signoff account records as the current time to the nearest minute.

The signoff date and time are displayed as a message on the terminal user's console or batch log device each time the terminal user signs off.

TIME OFF = mm/dd/yy hh:mm:ss

4.5.2.3 Job Number

The job number is a hexadecimal number assigned to a job consisting of one or more tasks. This number is collected and stored in the ATF signon, signoff, and task completion or data overflow account records.

4.5.2.4 Date and Time a Job is Submitted

The date and time a job is submitted to the batch queue is collected and stored as a hexadecimal number in the ATF signon account record.

4.5.2.5 Total Number of Tasks Executed

The total number of tasks executed as part of a batch job or terminal session is collected and stored as a hexadecimal number in the ATF signoff account record.

4.5.2.6 Total Number of Command Substitution System (CSS) Calls Executed

The total number of CSS calls executed by a batch job or terminal user is collected and stored as a hexadecimal number in the ATF signoff account record.

4.5.2.7 Total Number of Commands Entered

The total number of commands entered in the system by a batch job or during a terminal session is collected and stored as a hexadecimal number in the ATF signoff account record.

4.5.2.8 Total Number of DISPLAY FILES Commands Entered

The total number of DISPLAY FILES commands entered in the system by a batch job or during a terminal session is collected and stored as a hexadecimal number in the ATF signoff account record.

4.5.2.9 User Identifier (Userid)

The userid with which the terminal user signs on the system is collected and stored in the ATF signon, signoff, and task completion or data overflow account records.

4.5.2.10 Renamed and Deleted Files

Data pertaining to files that are renamed or deleted is collected and stored in the ATF renaming or deletion account records. This data consists of the:

- date and time the file was created,
- current file descriptor (fd),
- number of sectors in the file,
- date and time the file was renamed,
- new fd, and
- new account number to which the file was assigned.

4.5.2.11 Account Number

The account number with which the terminal user signs on the system is collected and stored in all the ATF account records.

4.5.3 Task Related Accounting Data

The accounting data collected for each task processed is:

- CPU time,
- roll time,
- wait time,
- date and time a task started and terminated,
- userid,
- memory used,

- count of I/O transfers to I/O devices,
- roll count,
- fd from which a task was loaded,
- user supplied information,
- account number,
- job number, and
- end of task code.

All task related accounting data is described in the following sections.

4.5.3.1 Central Processing Unit (CPU) Time

CPU time is collected and stored in the ATF task completion or data overflow account records as the number of 100 ms increments. All CPU time is collected and stored in three categories:

- user CPU time,
- operating system CPU time, and
- roll CPU time.

User CPU time is the amount of CPU time spent executing user code and nonreentrant system code. Operating system CPU time is the amount of CPU time spent executing reentrant system code on behalf of the user, and roll CPU time is the amount of CPU time spent rolling a task into memory. User and operating system CPU times are displayed as a message on the terminal user's console or batch log device. The following message is displayed when a terminal user's task terminates:

```
taskid-END OF TASK CODE=xxxx CPUTIME=utime/ostime
```

The following message is displayed when data overflow occurs:

```
taskid-ACCOUNTING DATA RESET CPUTIME=utime/ostime
```

The following message is displayed when the terminal user signs off:

```
CPUTIME=utime/ostime
```

The user and operating system CPU times can also be displayed by the DISPLAY ACCOUNTING command. See the OS/32 Multi-Terminal Monitor (MTM) Reference Manual.

4.5.3.2 Roll Time

Roll time is the time a task spends waiting to execute because it was rolled out of memory. All roll time is collected and stored in the ATF task completion or data overflow account records as the number of 100 ms increments. Roll time can be displayed by the DISPLAY ACCOUNTING command. See the OS/32 Multi-Terminal Monitor (MTM) Reference Manual.

4.5.3.3 Wait Time

Wait time is the time a task spends waiting to execute because of a request that task issued for itself (e.g., I/O request). Wait time excludes all PAUSE requests and the time a task spends suspended by another task or by the terminal user. All wait time is stored in the ATF task completion or data overflow account records as the number of 100 ms increments. Wait time can be displayed by the DISPLAY ACCOUNTING command. See the OS/32 Multi-Terminal Monitor (MTM) Reference Manual.

4.5.3.4 Date and Time a Task Started and Terminated

The date and time a task started and terminated is collected and stored in the ATF task completion or data overflow account records as the number of 100 ms increments.

4.5.3.5 User Identifier (Userid)

The userid with which the terminal user signs on the system is collected and stored in the ATF signon, signoff, and task completion or data overflow account records.

4.5.3.6 Memory Used by a Task

The total amount of memory used by a task during execution is collected and stored in the ATF task completion or data overflow account records as the number of 256-byte segments. Memory used is collected and stored in three categories:

- impure segment space,
- shared segment space, and
- system space.

Impure segment space is the maximum amount of impure memory used by a task during execution. Shared segment space is the maximum amount of shared memory (i.e., pure, TCOM, RTL) used by a task during execution, and system space is the maximum amount of system space used by a task during execution.

4.5.3.7 Count of Input/Output (I/O) Transfers to I/O Devices

All I/O transfers (read, write, test and set) made to an I/O device assigned to a particular device class are counted, collected, and stored as a hexadecimal number in the ATF task completion or data overflow account records. These numbers are stored in the appropriate class category in the ATF. Each I/O class supported by the accounting facility occupies eight bytes in each ATF account record. The number of bytes transferred every time an I/O transfer occurs is counted and stored as a hexadecimal number in the ATF. The count of I/O transfers can be displayed by the DISPLAY ACCOUNTING command. I/O transfer limits are established at MTM SYSGEN time by the IOTAB or UIOLIM macros (Chapter 2 of this manual) and can be overridden by the terminal user for batch jobs through the SIGNON command. See the OS/32 Multi-Terminal Monitor (MTM) Reference Manual.

4.5.3.8 Roll Count

Roll count is the total number of times the impure segment of a task is rolled into memory. The roll count is collected and stored as a hexadecimal number in the ATF task completion or data overflow account records. The roll count can be displayed by the DISPLAY ACCOUNTING command. See the OS/32 Multi-Terminal Monitor (MTM) Reference Manual.

4.5.3.9 File Descriptor (fd) from which a Task was Loaded

Every time a task is loaded, the fd from which it was loaded is stored in the ATF task completion or data overflow account records. This data aids the user in determining which system tasks are used more often, and whether they should be made resident or nonresident.

4.5.3.10 User Supplied Information

User supplied information is eight bytes of data supplied by the terminal user through the SETACCT macro. This information is stored in the ATF task completion or data overflow account records. See the OS/32 System Macro Library Reference Manual.

4.5.3.11 Account Number

The account number with which the terminal user signs onto the system is collected and stored in all ATF account records.

4.5.3.12 Job Number

The job number to which a particular task is assigned is collected and stored in the ATF signon, signoff, and task completion or data overflow account records.

4.5.3.13 End of Task Code

The end of task code with which each task terminates is collected and stored in the ATF task completion or data overflow account records. The code is displayed on the terminal user's console or batch log device each time a user's task terminates:

```
taskid-END OF TASK CODE=xxxx  CPUTIME=utime/ostime
```

If a task exceeds the CPU time limit, the end of task code 1200 is displayed when the task terminates.

If a task exceeds the I/O transfer limit, the end of task code 1210 is displayed when the task terminates.

4.6 TERMINAL USER ACCESS TO ACCOUNTING INFORMATION

A terminal user can access accounting information:

- at task termination,
- when a task control block (TCB) counter overflow occurs,
- at signoff time,
- through the DISPLAY ACCOUNTING command (see the OS/32 Multi-Terminal Monitor (MTM) Reference Manual), or
- through the fetch accounting information (FETACCT) macro (see the OS/32 Multi-Terminal Monitor (MTM) Reference Manual).

4.6.1 Task Termination

When a terminal user's task terminates execution, accounting information is displayed on the terminal user's console or batch log device in this message:

```
taskid-END OF TASK CODE=xxxx CPUTIME=utime/ostime
```

4.6.2 Task Control Block (TCB) Counter Overflow

When the counters for data collection in the TCB overflow, this accounting information is displayed on the terminal user's console or batch log device in this message:

```
taskid-ACCOUNTING DATA RESET CPUTIME=utime/ostime
```

4.6.3 Signoff Time

When a terminal user signs off the system, accounting information is displayed to the terminal user's console or batch log device in these messages:

```
ELAPSED TIME=hh:mm:ss          CPUTIME=utime/ostime  
SIGNON LEFT=hh:mm:ss          CPU LEFT=hh:mm:ss  
TIME OFF=mm/dd/yy  hh:mm:ss
```

If there are no sign on or CPU time limits set for the account, SIGNON LEFT= and CPU LEFT= are not displayed at sign off.

CHAPTER 5 MULTI-TERMINAL MONITOR (MTM) OPERATING PROCEDURES

5.1 INTRODUCTION

Before loading and starting MTM, an authorized user file (AUF) must already exist. See Chapter 3. The commands used to load, establish, start, and control MTM are presented in this chapter.

5.2 LOADING AND ESTABLISHING THE MULTI-TERMINAL MONITOR (MTM)

The following two commands load and establish MTM as the current task:

Formats:

```
LOAD .MTM,fd
```

```
TASK .MTM
```

Parameter:

fd is the file descriptor of the device from which MTM is loaded.

Functional Details:

The priority of MTM is established when MTM is linked. To ensure that MTM is not at the same level as the real-time system, MTM should be set at a priority lower than the lowest priority of real-time tasks in the operating system. If MTM is set at a lower priority, it executes as a background system. Since MTM is almost totally input/output (I/O) bound, it does not interfere with other tasks. Tasks executing in an MTM environment will run at a maximum priority of one less than the priority of MTM.

| START |

5.3 START COMMAND

The START command starts MTM. Parameters for the START command may be specified in any order.

Format:

```
START ,AUF=fd [,ATF=fd] [,TERMINAL=(fd1 [,fd2,...,fdn])]
      [,BATCH=(n,fd)] [,PAGE=fd]
```

Parameters:

AUF= fd is the file descriptor of the authorized user file (AUF). This file must reside on account 255. The authorized user utility must be used to create the AUF prior to running MTM. This parameter is required.

ATF= fd is the file descriptor of the accounting transaction file (ATF). The ATF accumulates accounting data in systems with accounting support. If the indicated file exists, records are appended. If it does not exist, MTM automatically allocates it. This parameter must be specified if the operating system is generated with accounting support. A user who does not wish to retain accounting information may specify ATF=NULL.

TERMINAL= fd is the file descriptor of the terminal devices that MTM is to monitor. Up to 64 terminal fds can be specified. If the command line entered exceeds one line, specify additional terminal devices with the .MTM ADD command.

BATCH= n is a decimal number from 1 through 64 indicating the number of concurrent batch streams that MTM can start.

fd is the file descriptor of the batch queue. The batch queue is an indexed file containing fds of the batch streams to be processed. If the fd specified does not exist, MTM allocates the file.

PAGE= fd is the file descriptor of the page file used by local and global variables. If this parameter is omitted, MTM automatically allocates a file named PAGE on the system volume, account 255.

Functional Details:

The number of concurrent batch streams allowed, plus the number of terminals to be monitored, cannot exceed the maximum number of users allowed (NUSERS) at MTM system generation (sysgen).

MTM can optionally load and start an assistant task (which runs under the task name .MTMASST). An assistant task executes a supervisor call 7 (SVC7) on behalf of MTM. This task can improve MTM response time because MTM does not have to wait while an SVC7 is in progress.

When started, MTM tries to load and start the assistant task. MTM assumes that the assistant task image is on the file MTMASST.TSK on the system volume. If this file does not exist, or if the load fails, MTM is not affected. However, MTM issues its own SVC7 calls and response time is slower.

Because the operating system cancels all subtasks when the monitor task goes to end of task, .MTMASST is also cancelled when MTM goes to end of task. If an attempt is made to restart MTM, MTM goes to end of task 254.

When MTM is started from a CSS file, and an .MTM command immediately follows the START command, an SVC6-ERR TYPE=QUE POS=.MTM error may occur. This error results if MTM task has not completed initialization when the .MTM command is entered. To avoid this problem, put time delay commands such as \$WAIT 2 between the START and .MTM commands.

Example:

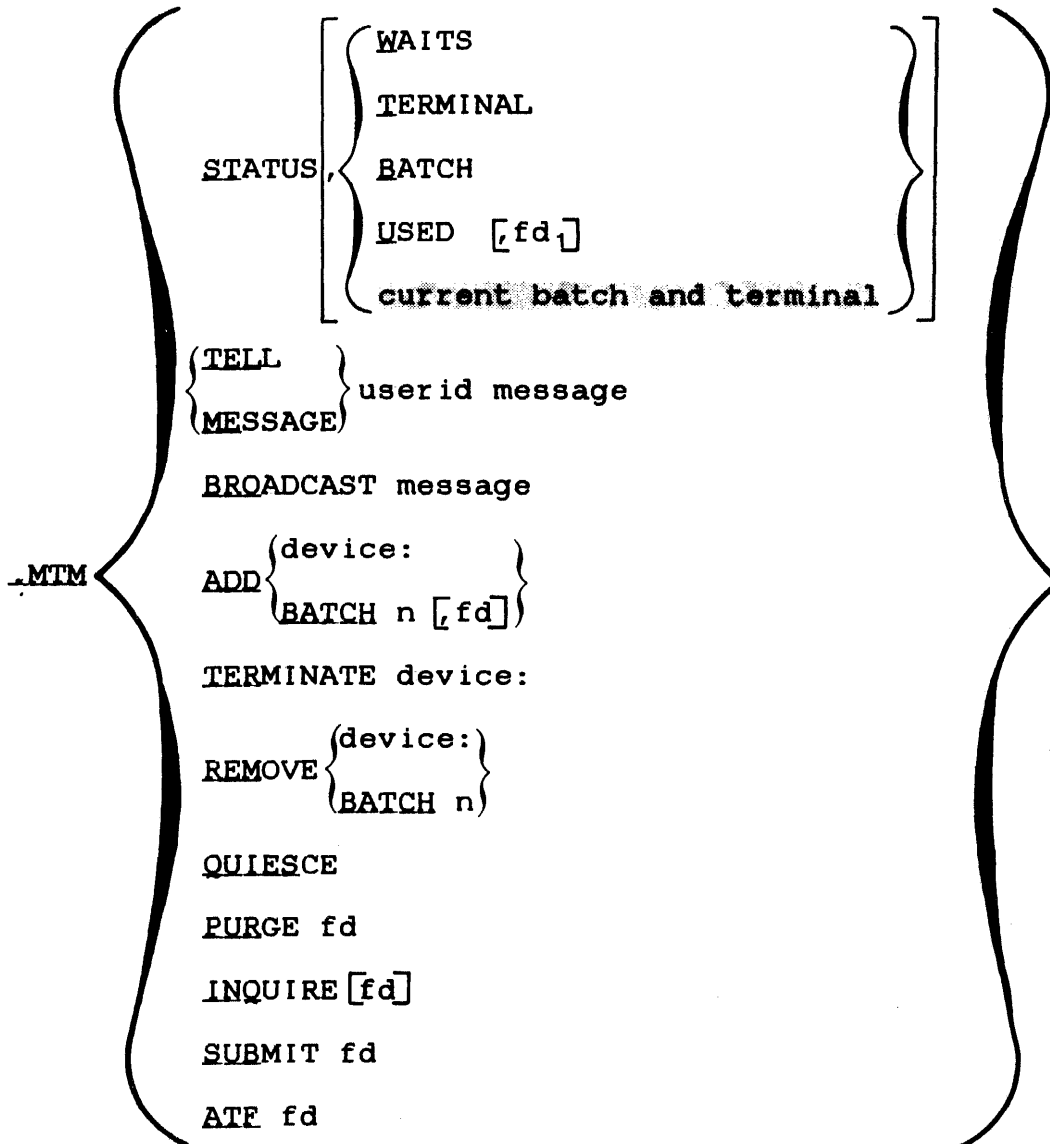
This command starts MTM using an AUF named ACCTS.32/255 and an ATF named ATF./255. Three terminals are to be monitored. Four batch jobs may run concurrently, and the file BATCH.Q is assigned as the batch queue.

```
ST,AUF=ACCTS.32,ATF=ATF,T=(CR10:,CR12:,CR14:),B=(4,BATCH.Q)
```

5.4 .MTM COMMAND

The .MTM command enables the system operator to control MTM, its terminals, and users from the system console.

Format:



Parameters:

STATUS displays current information. If no further parameter is specified, information on terminal users and batch jobs is displayed.

WAITS displays to the system console system resources available to MTM, the peak usage of system resources, and the number of times users had to wait for resources to become available.

TERMINAL indicates that terminal information is to be displayed.

BATCH indicates that batch information is to be displayed. The number of batch streams that can run concurrently is also displayed.

USED displays in histogram form the number of times a command buffer was used during MTM execution. See Figure 5-1 for an example of a listing output by the USED parameter.

fd₁ specifies the file or device to which the output from the USED parameter is sent. If fd is omitted, the system console is the display device.

TELL displays a message on a specified user terminal.

MESSAGE displays a message on a specified user terminal.

userid identifies the terminal user.

BROADCAST displays a message on all terminals whether or not they are in use.

message is a 1- to 62-character alphanumeric string indicating the message to be broadcast.

ADD adds an additional terminal or concurrent batch to the MTM system.

device is the new terminal to be added. The total number of terminals supported by MTM must not exceed (NUSERS - number of batch streams).

n is the number of additional concurrent batch streams to be allowed. The total number of concurrent batch streams supported by MTM must not exceed (NUSERS - number of terminals). This parameter increases the number of allowable concurrent batch streams supported by MTM but does not add batch streams to the batch queue. To add batch streams to the batch queue, use the SUBMIT command.

fd specifies the file descriptor of the batch queue file. If a batch queue currently exists, this parameter is ignored.

TERMINATE immediately cancels a terminal user's task and signs off a terminal user. This parameter does not affect batch jobs.

device is the terminal to which the user to be terminated is signed on.

REMOVE stops MTM from accepting commands at a specific terminal or reduces the number of allowable concurrent batch jobs.

device is the terminal to be removed. If the terminal is currently in use, the terminal user is allowed to continue until a SIGNOFF command is entered. When the terminal is removed, MTM closes the lu assignment for the terminal.

n is the number of concurrent batch streams to be removed. Currently active jobs are allowed to complete execution. If n is greater than the number of current batch streams, the number of concurrent batch streams supported by MTM is reduced to zero. If the new number of batch streams is less than the number of currently active batch jobs, the extra batch jobs are allowed to go to completion.

QUIESCE terminates MTM; new signons are prevented and batch job initiation is stopped. When all terminal users sign off the system and all currently active batch jobs terminate, MTM goes to end of task. Before MTM goes to end of task, information on the availability and usage of, and the number of times users had to wait for, system resources is output. If MTM discovers that command buffers are missing (the buffer chain has been broken) it pauses to allow user action (such as a panic dump). When the CONTINUE command is entered MTM goes to end of task code 210.

PURGE removes a batch job from the system and from the batch queue. If the batch job is currently active, it is cancelled.

fd is the file descriptor of the file containing the batch job.

INQUIRE displays the status of a batch job.

fd is the file descriptor of the file containing the batch job. If fd is omitted, the status of all submitted jobs is displayed.

SUBMIT adds a job to the batch queue.

fd is the file descriptor of the file containing the batch job to be added to the batch queue.

ATF changes the accounting transaction file. The current ATF is closed and a new ATF is allocated (if necessary) and assigned.

fd is the file descriptor of the new ATF.

Functional Details:

.MTM STATUS, WAITS, .MTM QUIESCE, and .MTM STATUS, USED can be used to determine if the numbers specified at MTM sysgen for command buffers (NCSBFS), console buffers (NCONBFS), sector buffers (NSECTS), and logical units (LNKOPT) are optimal. Increasing the available resources at MTM sysgen can help to eliminate excessive wait conditions. Decreasing resources can eliminate unnecessary memory overhead.

Examples:

.MTM STATUS Command:

```
.MTM STA
.MTM: TERMINAL USERS -
.MTM: OS832-BATCH>MTM-MODE OS3220-BATCH>MTM-MODE LEE-CT42:>MTM-MODE
.MTM: ROD-CT4C:>MTM-MODE RICH-CT7A:@MTM-MODE LAURIE-CT7C:@MTM-MODE
.MTM: JOB M300:OS832.JOB/96 EXECUTING USER ID=OS832
.MTM: JOB M300:OS3220.JOB/96 EXECUTING USER ID=OS3220
.MTM: WAITING JOBS = 1
.MTM: 2 BATCH STREAMS
```

| .MTM STATUS, WAITS Command:

| .MTM STA, W

| .MTM: | TOTAL | PEAK | CURRENT | WAITS | |
|-------|-------|------|---------|-------|-------------|
| .MTM: | 42 | 32 | 29 | 0 | CMD BUFFERS |
| .MTM: | 0 | 0 | 0 | 0 | CON BUFFERS |
| .MTM: | 7 | 4 | 0 | 0 | SEC BUFFERS |
| .MTM: | 50 | 10 | 6 | 0 | LU |

| TOTAL indicates the maximum number available.

| PEAK indicates the maximum number concurrently used.

| CURRENT indicates the number in use at the time the command was executed.

| WAITS indicates the number of times users were placed in a wait state.

| .MTM STATUS, TERMINAL Command:

| .MTM STA, T

| | | | |
|----------------------------|-----------------------|-----------------------|--|
| .MTM: TERMINAL USERS - | | | |
| .MTM: OS832-BATCH>MTM-MODE | OS3220-BATCH>MTM-MODE | LEE-CT42:>MTM-MODE | |
| .MTM: ROD-CT4C:>MTM-MODE | RICH-CT7A:@MTM-MODE | LAURIE-CT7C:@MTM-MODE | |

| .MTM STATUS, BATCH Command:

| .MTM STA, B

| | | |
|------------------------------|-----------|----------------|
| .MTM: JOB M300:OS832.JOB/96 | EXECUTING | USER ID=OS832 |
| .MTM: JOB M300:OS3220.JOB/96 | EXECUTING | USER ID=OS3220 |
| .MTM: WAITING JOBS = 1 | | |
| .MTM: 2 BATCH STREAMS | | |

.MTM STATUS,USED Command:

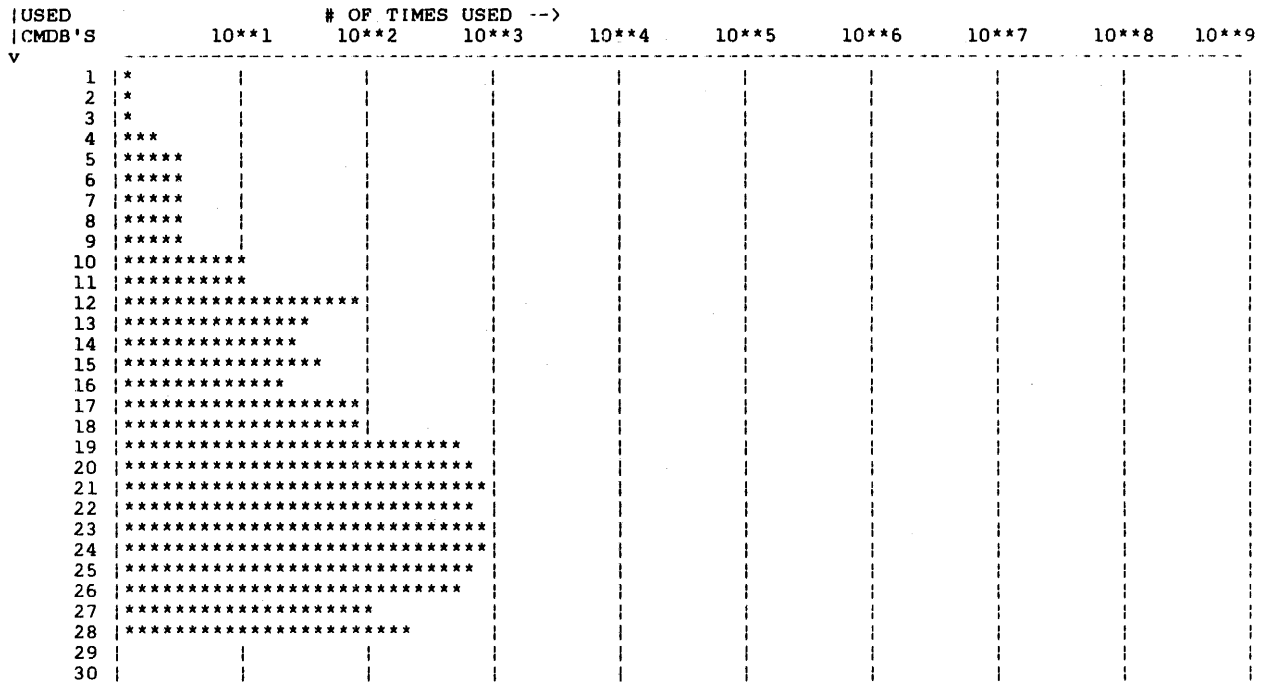


Figure 5-1 Sample Output from the .MTM STATUS,USED Command

This command displays, in histogram format, the number of times the indicated number of command buffers were used concurrently. Figure 5-1 indicates, for example, that four command buffers were concurrently used three times, 12 were concurrently used 90 times, and 24 were concurrently used 900 times.

| .MTM TELL and .MTM MESSAGE Commands:

| .MTM TEL SAM, YOUR MAG TAPE IS MOUNTED.

| .MTM ME SAM, YOUR MAG TAPE IS MOUNTED.

| Sends the indicated message to the terminal signed on with a
| user-id of SAM.

| .MTM BROADCAST Command:

| .MTM BRO MTM IS GOING DOWN IN 15 MINUTES.

| Sends the indicated message to all terminals currently
| assigned to MTM.

| .MTM ADD Command:

| .MTM ADD CRT2:

| Adds the device CRT:2 to MTM.

| .MTM ADD BATCH 2

| Increases the number of batch streams allowed by 2.

| .MTM TERMINATE Command:

| .MTM TER CRT8:

| Terminates the user signed on to CRT8:.

| .MTM REMOVE Command:

| .MTM REM CR10:

| Removes CRT10: from MTM upon the next sign off command.

.MTM REM BATCH 3

Decreases the number of batch streams allowed by 3.

.MTM QUIESCE Command:

.MTM QUIES

Terminates MTM after all users have signed off and all batch jobs have terminated. The following table is output prior to MTM termination.

.MTM QUIESCE

| .MTM: | TOTAL | PEAK | CURRENT | WAITS | |
|-------------------------|-------|------|---------|-------|-------------|
| .MTM: | 42 | 32 | 29 | 0 | CMD BUFFERS |
| .MTM: | 0 | 0 | 0 | 0 | CON BUFFERS |
| .MTM: | 7 | 4 | 0 | 0 | SEC BUFFERS |
| .MTM: | 50 | 10 | 6 | 0 | LU |
| .MTM END OF TASK CODE 0 | | | | | |

.MTM PURGE Command:

.MTM PUR MT30:TEST.JOB/8

Cancels the indicated batch job if active, or removes it from the batch queue if not active.

.MTM INQUIRE Command:

.MTM INQ

| | | |
|----------|--------------------|-----------|
| .MTM:JOB | M300:OS832.JOB/96 | EXECUTING |
| .MTM:JOB | M300:OS3220.JOB/96 | EXECUTING |
| .MTM:JOB | M300:OS3230.JOB/96 | WAITING |

Displays the status of all submitted jobs.

.MTM INQ FRED:TASK.JOB/18
JOB FRED:TASK.JOB/18 NOT FOUND

Displays the status of the specified job.

| .MTM SUBMIT Command:

| .MTM SUB MT30:TEST.JOB/8

| Adds the specified job to the batch queue.

5.5 MULTI-TERMINAL MONITOR (MTM) END OF TASK CODES

With the exception of end of task codes 0 and 255, MTM goes to end of task due to a situation that it cannot resolve. The format of the end of task message on the system console is:

.MTM END OF TASK CODE nnn

The MTM end of task codes and meanings are listed in Table 5-1.

TABLE 5-1 MTM END OF TASK CODES

| CODE | MEANING |
|------|---|
| 0 | Normal end of task |
| 1 | No message buffers |
| 2 | Entry is removed from the task queue. |
| 3 | Task has just been loaded but its task control block cannot be located. |
| 4 | An SVC 6 error has occurred but the error is not listed in the error table. |
| 6 | .MTM START parameter(s) is wrong. |
| 7 | A command buffer length of less than 72, or NUSERS less than 1, was specified. |
| 8 | Address for the DMT in the VMT is 0 for a direct access volume. |
| 9 | A read or write request was made to record 0 in the batch queue. This request is illegal. |
| 11 | Number of I/O classes in the operating system is larger than those in the MTM sysgen. |

TABLE 5-1 MTM END OF TASK CODES (Continued)

| CODE | MEANING |
|------|--|
| 12 | MTM could not assign the system console. |
| 13 | MTM could not assign the accounting transaction file. |
| 200 | PSM was specified without accounting support. |
| 210 | Command buffers missing upon .MTM QUIESCE (Buffer chain broken). |
| 254 | User attempted to restart MTM. MTM is not a restartable task. |
| 255 | MTM has been cancelled. |

5.6 MULTI-TERMINAL MONITOR (MTM) ERROR MESSAGES

BATCH ASGN-ERR TYPE=ASGN JOB=fd/act

indicates the lu was not assigned when trying to assign a file at the start of a batch job.

BATCH ASGN-ERR TYPE=BUFF JOB=fd/act

indicates insufficient system space to obtain file control block (FCB) when trying to assign a file at the start of a batch job.

BATCH ASGN-ERR TYPE=FUNC JOB=fd/act

indicates invalid function detected for specified device/file when trying to assign it at the start of a batch job.

BATCH ASGN-ERR TYPE=I/O JOB=fd/act

indicates I/O error detected on specified file when trying to assign file at the start of a batch job.

BATCH ASGN-ERR TYPE=lu JOB=fd/act

indicates illegal or unassigned lu detected when trying to assign a file when starting a batch job.

BATCH ASGN-ERR TYPE=NAME JOB=fd/act

indicates mismatched filename.ext detected when trying to assign a file at the start of a batch job.

BATCH ASGN-ERR TYPE=PRIV JOB=fd/act

indicates mismatched access privilege detected when trying to assign a file at the start of a batch job.

BATCH ASGN-ERR TYPE=SIZE JOB=fd/act

indicates erroneous record length/size detected when trying to assign a file at the start of a batch job.

BATCH ASGN-ERR TYPE=VOL JOB=fd/act

indicates no such volume/device detected when trying to assign a file at the start of a batch job.

BATCH NOT ACTIVE

indicates MTM was started without a batch queue, and a batch request had been made.

BATCH QUEUE ASGN-ERR

indicates the batch queue specified in the START command could not be assigned.

BATCH QUEUE I/O-ERR TYPE=DU- - -ALL BATCH REMOVED- - -

indicates device unavailable detected on batch queue.

BATCH QUEUE I/O-ERR TYPE=EOF- - -ALL BATCH REMOVED- - -

indicates end of file error detected on batch queue.

BATCH QUEUE I/O-ERR TYPE=EOM- - -ALL BATCH REMOVED- - -

indicates end of medium error detected on batch queue.

BATCH QUEUE I/O-ERR TYPE=PRTY- - -ALL BATCH REMOVED- - -

indicates parity or recoverable error detected on batch queue.

BATCH QUEUE I/O-ERR TYPE=UNRV- - -ALL BATCH REMOVED- - -

indicates unrecoverable error detected on batch queue.

BATCH QUEUE LRECL-ERR

indicates batch queue given in the START command has an invalid record length of less than 40 bytes.

CANNOT ASSIGN AUF FILE

indicates that MTM cannot assign an AUF file.

xxxx CMD BUFFERS MISSING

indicates that the number of command buffers specified by xxxx have been discovered to be missing upon .MTM QUIESCE.

COMMAND BUFFER LENGTH LESS THAN 72

indicates that a command buffer length of 72, or NUSERS less than 1 was specified.

dev-I/O ERROR status USER TERMINATED

indicates unrecoverable error has occurred on a device; error status returned indicated as status. MTM automatically signs off a user signed on to the device.

EXTRANEIOUS OPERANDS

indicates an MTM command has too many operands.

FILE DESCRIPTOR ERROR

indicates the fd specified in the command is omitted, or its format is incorrect.

INVALID COMMAND

indicates the entered MTM command is unrecognized.

INVALID PARAMETER

indicates a command has been entered with an invalid parameter, bad syntax, or the device cannot be found.

NO BUFFER AVAIL FOR CMD

all the command and message buffers are full.

START OPTIONS FORM-ERR

indicates invalid format in the START command.

START OPTIONS PARM-ERR

indicates invalid START parameter.

TERMINAL ASSIGN ERROR

indicates adding a new terminal to the system has failed. The device is not recognized or not available.

TERMINALS OR BATCH EXCEED MAXIMUM

indicates number of concurrent batch streams allowed exceeds 64; number of terminals exceeds 64; number of batch streams and terminals exceed the NUSERS established at MTM sysgen time.

USERID NOT FOUND

indicates a current user has not been found with this identification.

APPENDIX A
MULTI-TERMINAL MONITOR (MTM) SYSTEM GENERATION (SYSGEN) OPTIONS

CDBUFSIZ EQU n (size of sector buffers)

CONBFSZ EQU n (size of console buffers)

CPULIMIT EQU n (CPU time limit)

CSBFSZ EQU n (size of command buffers)

IOTAB macro: (I/O class number, identification, limit)

[label] IOTAB (classno,classid [,iocount])₀
 '.....'
 (classno,classid [,iocount])_n

NCONBFS EQU n (number of console buffers)

NCSBFS EQU n (number of extra command buffers)

NCSLVS EQU n (number of CSS levels)

NMSGs EQU n (number of message buffers)

NSECTS EQU n (number of sector buffers)

NUSERS EQU n (number of concurrent users)

PRMTDEV macro: (device to which prompt sequence applies)

[label] PRMTDEV (pname₁, dev₁ [, ..., dev_n])₁
[, ..., (pname_n, dev₁ [, ..., dev_n])_n]

[label] PRMTDEV (SIGMA, dev₁ [, ..., dev_n])

PRMTSEQ macro: (prompt sequence; command, task, build)

[label] PRMTSEQ (SIGMA)

[label] PRMTSEQ (pname₁, (com-req, x), (task-exec, x),
(task-read, x), (build-read, x))₁
, ...,
(pname_n, (com-req, x), (task-exec, x),
(task-read, x), (build-read, x))_n

[label] PRMTSEQ (STANDARD, (com-req, x), (task-exec, x),
(task-read, x), (build-read, x))₁

QUESIZE EQU n (size of task queue)

SGN.ACT EQU n (accounting support/number of classes)

SGN.BTCP EQU $\left. \begin{matrix} 0 \\ 1 \end{matrix} \right\}$ (batch job pause)

SGN.MVOL EQU C'name' (default user volume)

SGN.PRIO EQU n (batch priority)

SGN.SCHR EQU n (subtask scheduling)

SGN.ROLL EQU $\left. \begin{matrix} 0 \\ 1 \end{matrix} \right\}$ (tasks rollable)

SGN.SPL EQU n (SPL/32 support/number of buffers)

SGN.VAR EQU n (numbers of variables)

SGN.@GLO EQU n (number of new global variables)

SGN.@INT EQU n (number of internal variables)

SGN.\$HSP EQU $\left. \begin{matrix} 0 \\ 1 \end{matrix} \right\}$ (HASP interface)

UCPULIM macro: (CPU time limits for I/O classes)

[label] UCPULIM limit_{0t}, limit_{0b}, ..., limit_{nt}, limit_{nb}

UIOLIM macro: (I/O limits, all I/O classes per user class)

[label] UIOLIM (u_{0io0t}, ..., u_{0iont}), (u_{0io0b}, ..., u_{0ionb})
,
,
(u_{nio0t}, ..., u_{niont}), (u_{nio0b}, ..., u_{nionb})

VERSION macro: (MTM parameter file identification)

[label] VERSION 'id'

APPENDIX B
 AUTHORIZED USER UTILITY COMMANDS

ACCOUNTS n

ADD actno,groupno,password, [userid] [{signontime}] [{CPUtime}]
 [,privileges]

CHANGE actno, [groupno] , [password] , [userid] [{signontime}] [{CPUtime}]
 [,privileges]

DELETE actno

END

LIST [actno₁ [,actno₂]]

MODE [{CREATE}]
 [{UPDATE}] =fd₃
 [{REPORT}]

PAUSE

PRIVILEGE [actno₁ [,actno₂] [{privileges}]]

START, {COMMAND} =fd₁ , LIST=fd₂ , {CREATE} =fd₃ [,ACCOUNTS=n]
 {CMD} {UPDATE} {REPORT}

TOTAL [C]

APPENDIX C
AUTHORIZED USER UTILITY MESSAGE SUMMARY

taskid > for CON:
or
---> under MTM

is the prompt, generated for an interactive device,
indicating an input request.

ACTUTILITY rr-uu

identifies the account utility. The revision level is rr;
the update level is uu. This message is output only at the
initial start of the utility.

ACCOUNT UTILITY RUN xx/xx/xx yy:yy:yy

appears on the system console when the utility is started.
It informs the operator that the account utility has been
run.

***SET DATE & TIME

is the date request that pauses the utility so the current
date and time can be set for the system. The system CONTINUE
command transfers control back to the utility. This message
is output when the date and time are not set.

***ACCOUNT NUMBER xxx WAS INACTIVE

indicates the requested command could not be performed
because the indicated account record was inactive. The
indicated account number is xxx.

***ACCOUNT NUMBER xxx WAS ACTIVE

indicates the requested command could not be performed
because the indicated account record was already active. The
indicated account number is xxx.

ACCOUNTS=n

indicates the maximum account number on the current AUF.

***xxxxxx IS THE CURRENT MODE

indicates the output from the MODE command. The name of the current mode (CREATE, UPDATE, or REPORT) is xxxxxx.

?

is the question mark, along with some of the following messages, used to denote which parameter in the command caused the error. The command and a pointer (?) to the location of the error is displayed to the command input device.

***ILLEGAL INPUT COMMAND

indicates the command word or abbreviation could not be found in the table of legal commands.

***WRONG MODE FOR INPUT COMMAND

indicates the current mode does not allow the input command.

***ILLEGAL COMMAND FORM

indicates the input command uses too many separators or a separator other than a comma or equal character.

***ILLEGAL PARAMETER

indicates the input command is missing a required parameter or has too many parameters.

***ILLEGAL START PARAMETER

indicates a START command word or abbreviation was not found in the START command table.

***DUPLICATE START PARAMETER

indicates the START command contains a duplicate command word or more than one mode request.

***ILLEGAL FILE DESCRIPTOR

indicates the file descriptor used is syntactically incorrect.

***MISSING START PARAMETER

indicates at least one START parameter is missing from the specified mode.

***ALLOCATION/ASSIGNMENT ERROR

indicates a file cannot be allocated because of a filename conflict or the filename specified for the assignment could not be found on the specified disk.

***lu xx NOT ASSIGNED

indicates a requested function could not be performed because the logical unit xx was not assigned. The utility is paused and control transfers to the system. The system CONTINUE command continues the utility after the assignment.

***INPUT DEVICE ERROR xx on LOGICAL UNIT yy

indicates an input request has an I/O error on device xx assigned to logical unit yy. The authorized user utility pauses. The system CONTINUE command transfers control back to the utility.

***OUTPUT DEVICE ERROR xx ON LOGICAL UNIT yy

indicates an output request has the I/O error on device xx assigned to logical unit yy. The authorized user utility pauses. The system CONTINUE command transfers control back to the utility.

***MISSING PARAMETER

indicates the input command is missing a required parameter. On input errors, where the form of the command or its characters are incorrect, the command and a pointer (?) to the location of the error is displayed to the user console.

***MODE NOT ALLOWED

indicates an attempt to perform an UPDATE or a REPORT was made from an account other than 255.

***WARNING ACCOUNT 255 DOES NOT EXIST

indicates an attempt to end utility when account 255 does not exist.

APPENDIX D
ACCOUNTING TRANSACTION FILE (ATF) RECORD FORMATS

The data structure of the ATF is defined by the \$ATF data structure macro contained in SYSSTRUC.MLB. The fields and functional values in the ATF account records have standard symbolic names. These names may be accessed by coding and expanding the \$ATF data structure macro. See the Common Assembly Language Macro/32 Processor (CAL Macro/32) and Macro Library Utility Reference Manual. Each of the six ATF account record types is associated with a particular type of account activity. The record length of an ATF record varies with the number of input/output (I/O) classes supported by the system. The record formats shown in this appendix are 128 bytes and accept entries for the four default I/O classes. The record types are listed in Table D-1.

TABLE D-1 ATF RECORD TYPES

| TYPE | RECORD ACTIVITY |
|------|--------------------|
| 1 | Signon |
| 2 | Signoff |
| 3 | Task completion |
| 4 | Task data overflow |
| 5 | File deletion |
| 6 | File renaming |

These six record formats are illustrated in the following sections.

ATF FORMAT FOR SIGNON ACCOUNT RECORD TYPE

| | | | |
|---------|--|------|------------------------------|
| 0(0) | RECORD TYPE (ATF.TYPE) | 2(2) | ACCOUNT NUMBER (ATF.ACCT) |
| 4(4) | CURRENT DATE AND TIME (ATF.CRTM) | | |
| 8(8) | USER IDENTIFICATION | | |
| 12(C) | (ATF.UID) | | |
| 16(10) | JOB SERIAL NUMBER (ATF.JOB) | | |
| 20(14) | | | |
| 24(18) | FILE DESCRIPTOR OF | | |
| 28(1C) | COMMAND INPUT FILE | | |
| 32(20) | (ATF.UFD) | | |
| 36(24) | | | |
| 40(28) | DATE AND TIME JOB WAS SUBMITTED TO BATCH QUEUE (ATF.SUBT) | | |
| 44(2C) | FILLER | | |
| 48(30) | | | |
| | | . | |
| | | . | |
| | | . | |
| 116(74) | | | |
| 120(78) | FILLER | | |
| 124(7C) | | | |

ATF FORMAT FOR SIGNOFF ACCOUNT RECORD TYPE

| | | | | |
|---------|---|--|--------|---------------------------------------|
| 0(0) | RECORD TYPE (ATF.TYPE) | | 2(2) | ACCOUNT NUMBER (ATF.ACCT) |
| 4(4) | CURRENT DATE AND TIME (ATF.CRTM) | | | |
| 8(8) | USER IDENTIFICATION | | | |
| 12(C) | (ATF.UID) | | | |
| 16(10) | JOB NUMBER (ATF.JOB) | | | |
| 20(14) | NUMBER OF TASKS RUN (ATF.TASK) | | 22(16) | NUMBER OF CSS CALLS (ATF.CSS) |
| 24(18) | NUMBER OF COMMANDS ENTERED (ATF.CMDS) | | 26(1A) | NUMBER OF DISPLAY FILES (ATF.DISF) |
| 28(1C) | ELAPSED TIME (ATF.ELPS) | | | |
| 32(20) | FILLER | | | |
| 36(24) | | | | |
| | | | | |
| | | | | |
| 116(74) | | | | |
| 120(78) | FILLER | | | |
| 124(7C) | | | | |

ATF FORMAT FOR TASK COMPLETION (ATYP.TSK) AND TASK DATA
OVERFLOW ACCOUNT RECORD TYPES

| | | | | |
|--------|---|--|-----------|--------------------------------|
| 0(0) | RECORD TYPE (ATF.TYPE) | | 2(2) | ACCOUNT NUMBER (ATF.ACCT) |
| 4(4) | CURRENT DATE AND TIME(ATF.JOB) | | | |
| 8(8) | USER IDENTIFICATION | | | |
| 12(C) | (ATF.UID) | | | |
| 16(10) | JOB SERIAL NUMBER(ATF.JOB) | | | |
| 20(14) | TIME TASK WAS STARTED(ATF.STAR) | | | |
| 24(18) | | | | |
| 28(1C) | FD OF TASK | | | |
| 32(20) | (ATF.TFD) | | | |
| 36(24) | | | | |
| 40(28) | ACCOUNT NUMBER (ATF.ACT) | | 42(2A) | END OF TASK CODE (ATF.RC) |
| 44(2C) | USER-SUPPLIED INFORMATION | | | |
| 48(30) | (ATF.USER) | | | |
| 52(34) | IMPURE SEGMENT SIZE (ATF.IMP) | | 54(36) | PURE SEGMENT SIZE (ATF.PUR) |
| 56(38) | SYSTEM SPACE (ATF.SYS) | | 58(3A) | ROLL COUNT (ATF.ROLL) |
| 60(3C) | USER CPU TIME | | (ATF.CPU) | |
| 64(40) | OS CPU TIME | | | |
| 68(44) | RESERVED | | | |
| 72(48) | | | | |
| 76(4C) | WAIT TIME(ATF.WTIM) | | | |
| 80(50) | ROLL QUEUE TIME(ATF.RLTM) | | | |
| 84(54) | I/O COUNT FOR CLASS 0(ATF.IOCT) | | | |
| 88(58) | NUMBER OF BYTES TRANSFERRED FOR CLASS 0 (ATF.IOXF) | | | |

ATF FORMAT FOR TASK COMPLETION (ATYP.TSK) AND TASK DATA
 OVERFLOW ACCOUNT RECORD TYPES (Continued)

| 0(0) | RECORD TYPE (ATF.TYPE) | 2(2) | ACCOUNT NUMBER (ATF.ACCT) |
|---------|---|------|------------------------------|
| 92(5C) | I/O COUNT FOR CLASS 1 | | |
| 96(60) | NUMBER OF BYTES TRANSFERRED FOR CLASS 1 | | |
| 100(64) | I/O COUNT FOR CLASS 2 | | |
| 104(68) | NUMBER OF BYTES TRANSFERRED FOR CLASS 2 | | |
| 108(6C) | I/O COUNT FOR CLASS 3 | | |
| 112(70) | NUMBER OF BYTES TRANSFERRED FOR CLASS 3 | | |
| 116(74) | | | |
| 120(70) | FILLER | | |
| 124(7C) | | | |

ATF FORMAT FILE RENAMING ACCOUNT RECORD TYPE

| | | | | |
|---------|---|--|--------|------------------------------|
| 0(0) | RECORD TYPE (ATF.TYPE) | | 2(2) | ACCOUNT NUMBER (ATF.ACCT) |
| 4(4) | CURRENT DATE AND TIME (ATF.CRTM) | | | |
| 8(8) | VOLUME NAME (AFT.VOL) | | | |
| 12(C) | FILENAME (First Word) (ATF.ENAM) | | | |
| 16(10) | MESSAGE POINTER FOR SVC 6 (ATF.MSGP) | | | |
| 20(14) | FILENAME (Second Word) (ATF.ENM2) | | | |
| 24(18) | EXTENSION (ATF.EXT) | | | |
| 28(1C) | NUMBER OF RECORDS (ATF.RECS) | | | |
| 32(20) | NUMBER OF SECTORS (ATE.SECS) | | | |
| 36(24) | FILE CREATION DATE (ATF.CDAT) | | | |
| 40(28) | NEW FILENAME | | | |
| 44(2C) | | | | |
| 48(30) | EXTENSION (ATF.NWEX) | | | |
| 52(34) | ACCOUNT NUMBER (ATF.NWBC) | | 54(36) | FILLER |
| 56(38) | ID OF TASK THAT RENAMED FILE | | | |
| 60(3C) | | | | |
| 64(40) | FILLER | | | |
| 124(7C) | | | | |

ATF FORMAT FOR FILE DELETION RECORD TYPE

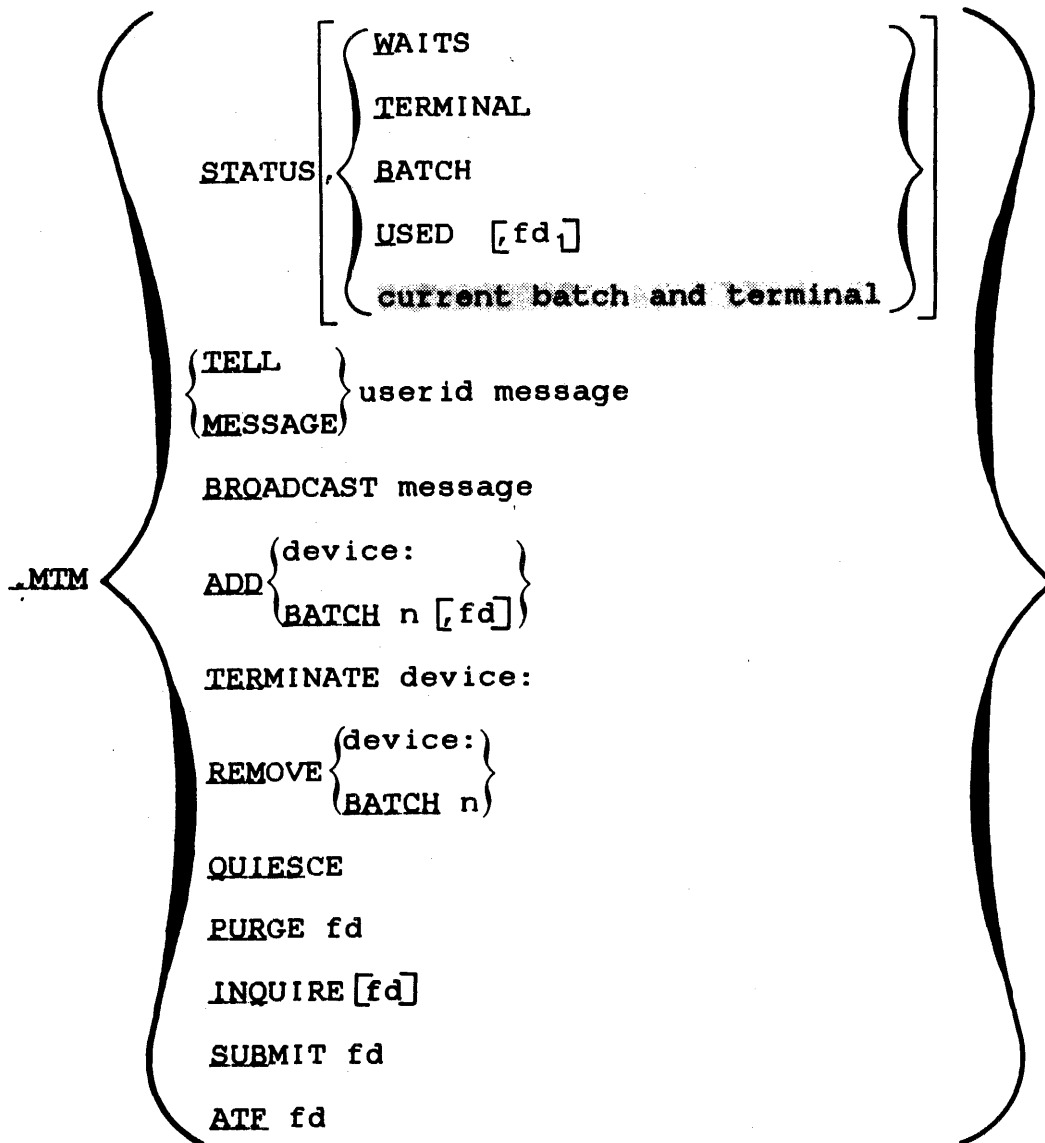
| | | | | |
|---------|---|--|------|------------------------------|
| 0(0) | RECORD TYPE (ATF.TYPE) | | 2(2) | ACCOUNT NUMBER (ATF.ACCT) |
| 4(4) | CURRENT DATE AND TIME (ATF.CRTM) | | | |
| 8(8) | VOLUME NAME (AFT.VOL) | | | |
| 12(C) | FILENAME (First Word) (ATF.FNAM) | | | |
| 16(10) | MESSAGE POINTER FOR SVC 6 (ATF.MSGP) | | | |
| 20(14) | FILENAME (Second Word) (ATF.FNM2) | | | |
| 24(18) | EXTENSION (ATF.EXT) | | | |
| 28(1C) | NUMBER OF RECORDS (ATF.RECS) | | | |
| 32(20) | NUMBER OF SECTORS (ATE.SECS) | | | |
| 36(24) | FILE CREATION DATE (ATF.CDAT) | | | |
| 40(28) | FILLER | | | |
| 44(2C) | | | | |
| 48(30) | | | | |
| 52(34) | | | | |
| 56(38) | ID OF TASK THAT DELETED FILE (ATF.TKID) | | | |
| 60(3C) | | | | |
| 64(40) | FILLER | | | |
| 124(7C) | | | | |

APPENDIX E
MULTI-TERMINAL MONITOR (MTM) OPERATOR COMMAND SUMMARY

LOAD .MTM,fd

TASK .MTM

START ,AUF=fd [ATF=fd] [,TERMINAL=(fd₁ [,fd₂,...,fd₃])]
[BATCH=(n,fd) [,PAGE=fd]



INDEX

| A | |
|-----------------------------|------|
| Account limit | 3-8 |
| Account number | 3-3 |
| | 4-6 |
| Accounting data | |
| job related | 4-4 |
| task related | 4-6 |
| terminal user access to | 4-10 |
| Accounting facility | 2-25 |
| system requirements | 4-1 |
| Accounting transaction file | 4-2 |
| ACCOUNTS command | 3-8 |
| ADD command | 3-10 |
| ADD parameter | 5-5 |
| Assistant task | 5-3 |
| ATF parameter | |
| .MTM command | 5-7 |
| MTM START command | 5-2 |
| ATF. See accounting | |
| transaction file. | |
| AUF parameter | 5-2 |
| AUF. See authorized user | |
| file. | |
| Authorized user file | 3-1 |
| Authorized user utility | 3-1 |
| ACCOUNTS command | 3-8 |
| ADD command | 3-10 |
| building and starting | 3-4 |
| CHANGE command | 3-12 |
| CREATE mode | 3-7 |
| | 3-18 |
| DELETE command | 3-14 |
| END command | 3-15 |
| LIST command | 3-16 |
| MODE command | 3-18 |
| PAUSE command | 3-19 |
| PRIVILEGE command | 3-20 |
| REPORT mode | 3-7 |
| | 3-18 |
| START command | 3-5 |
| TOTAL command | 3-26 |
| UPDATE mode | 3-7 |
| | 3-18 |
| | |
| B | |
| BATCH parameter | |
| .MTM ADD | 5-6 |
| .MTM REMOVE | 5-6 |
| MTM START command | 5-5 |
| .MTM STATUS | 5-2 |
| Batch processing | |
| CPU limit | 2-8 |
| CPU limit per user class | 2-40 |
| I/O limit | 2-42 |
| job pause | 2-26 |
| relative default priority | 2-28 |
| BROADCAST parameter | 5-5 |

| C | |
|------------------------------|------|
| CDBUFSIZ | 2-6 |
| CHANGE command | 3-12 |
| Command buffers | 2-9 |
| | 2-15 |
| CONBFSZ | 2-7 |
| Concurrent users | 2-20 |
| Console buffers | 2-7 |
| | 2-13 |
| CONTINUE command | 3-19 |
| Count of I/O transfers | 4-9 |
| CPU limit | |
| batch jobs | 2-8 |
| per account number | 3-3 |
| per accounting user class | 2-40 |
| CPU time | 4-7 |
| CPULIMIT | 2-8 |
| CREATE mode | 3-5 |
| | 3-7 |
| CSBFSZ | 2-9 |
| CSS levels | 2-17 |
| | |
| D | |
| Data collection | 4-1 |
| | 4-3 |
| Data structures | 2-46 |
| Date and time submitted | 4-5 |
| Date formats | 4-3 |
| Default volume | 2-27 |
| DELETE command | 3-14 |
| Dynamic scheduling | 2-32 |
| | |
| E | |
| Elapsed time | 4-4 |
| END command | 3-15 |
| End of task codes | 4-10 |
| | 5-12 |
| | |
| F | |
| File descriptor, task loaded | |
| from | 4-9 |
| File descriptors | 1-3 |
| Forced roll | 2-31 |
| | |
| G | |
| Global variables | 2-36 |
| new | 2-16 |
| | 2-37 |

| | | | |
|---------------------------|------|------------------------------|------|
| H | | MTM sysgen options | |
| HASP interface | 2-39 | (Continued) | |
| I | | batch job priority | 2-28 |
| I/O class table | 2-10 | command buffer size | 2-9 |
| I/O control block | 2-11 | console buffer size | 2-7 |
| I/O limit | | CPU limit per user class | 2-40 |
| per accounting user class | 2-42 | default user volume | 2-27 |
| per I/O class | 2-10 | forced roll | 2-31 |
| INQUIRE parameter | 5-7 | global and local | |
| Interactive processing | | variables | 2-36 |
| CPU limit | 3-4 | HASP interface | 2-39 |
| | 3-10 | I/O class table | 2-10 |
| | 3-13 | I/O count limit per user | |
| CPU limit per user class | 2-40 | class | 2-42 |
| default priority | 2-28 | internal variables | 2-38 |
| I/O limit | 2-42 | memory requirements | 2-46 |
| signon time limit | 3-4 | MTM link | 2-11 |
| | 3-10 | new global variables | 2-37 |
| | 3-12 | number of command buffers | 2-15 |
| Internal variables | 2-16 | number of concurrent | |
| | 2-38 | users | 2-20 |
| IOTAB macro | 2-10 | number of console buffers | 2-13 |
| J,K | | number of CSS levels | 2-17 |
| Job number | 4-5 | number of message buffers | 2-18 |
| | 4-10 | number of sector length | |
| L | | buffers | 2-19 |
| LINK OPTION command | 2-12 | OS/32 accounting facility | 2-25 |
| | 4-2 | prompt device | 2-21 |
| LINKOPT | 2-11 | prompt sequence | 2-22 |
| LIST command | 3-16 | scheduling | 2-32 |
| Local variables | 2-36 | sector buffer size | 2-6 |
| Logical unit | 2-11 | SPL/32 interface | 2-35 |
| M | | task queue size | 2-24 |
| Memory requirements | 2-46 | version identification | 2-45 |
| Memory used by a task | 4-8 | MTM version identification | 2-45 |
| Message buffers | 2-18 | MTM. See multi-terminal | |
| MESSAGE parameter | 5-5 | monitor. | |
| MODE command | 3-18 | .MTMASST task | 5-3 |
| .MTMASST task | 5-3 | MTMMAIN | 2-1 |
| .MTM command | | MTMPARMS.MAC | 2-2 |
| examples | 5-7 | MTMSGN | 2-1 |
| format | 5-4 | MTMSGN command | 2-4 |
| parameters | 5-5 | MTMSGN.CSS | 2-4 |
| MTM link | 2-11 | Multi-terminal monitor | |
| MTM operating procedures | | data structures | 2-46 |
| loading and establishing | | description | 1-1 |
| MTM | 5-1 | end of task codes | 5-12 |
| MTM START command | 5-2 | error messages | 5-13 |
| MTM START command | 4-2 | establishing accounts | 3-1 |
| | 5-2 | operating procedures | 5-1 |
| MTM sysgen options | 2-4 | system generation | 2-1 |
| batch job CPU limit | 2-8 | N,O | |
| batch job pause | 2-26 | NCONBFS | 2-13 |
| | | NCSBFS | 2-15 |
| | | NCSSLVS | 2-17 |
| | | NMSGs | 2-18 |
| | | NSECTS | 2-19 |
| | | Number of commands entered | 4-5 |
| | | Number of CSS calls executed | 4-5 |
| | | Number of DISPLAY FILE | |
| | | commands | 4-6 |
| | | Number of tasks executed | 4-5 |
| | | NUSERS | 2-20 |

| | |
|---|------|
| P | |
| PAGE parameter | 5-3 |
| PAUSE command | 3-19 |
| Priority scheduling | 2-32 |
| Priority scheduling mechanism | 2-33 |
| PRIVILEGE command | 3-20 |
| Privileges | |
| account | 3-21 |
| account 255 | 3-11 |
| default | 3-11 |
| PRMTDEV macro | 2-21 |
| PRMTSEQ macro | 2-22 |
| Program development feature | 2-16 |
| Prompt device | 2-21 |
| Prompt sequence | 2-22 |
| PSM. See priority scheduling mechanism. | |
| PURGE parameter | 5-7 |

| | |
|-------------------|------|
| Q | |
| QUESIZE | 2-24 |
| QUIESCE parameter | 5-6 |

| | |
|---------------------------|-----|
| R | |
| REMOVE paramater | 5-6 |
| Renamed and deleted files | 4-6 |
| REPORT mode | 3-5 |
| | 3-7 |
| Reporting utility | 4-1 |
| Roll count | 4-9 |
| Roll time | 4-8 |

| | |
|------------------------------|------|
| S | |
| Secondary directory | 2-6 |
| Sector buffer | 2-6 |
| | 2-19 |
| SGN.ACT | 2-25 |
| SGN.BTCP | 2-26 |
| SGN.@GLO | 2-37 |
| SGN.\$HSP | 2-39 |
| SGN.@INT | 2-38 |
| SGN.MVOL | 2-27 |
| SGN.PRIO | 2-28 |
| SGN.ROLL | 2-31 |
| SGN.SCHR | 2-32 |
| SGN.SPL | 2-35 |
| SGN.VAR | 2-36 |
| Signoff time | 4-11 |
| Signon time limit | 3-4 |
| Signon/signoff date and time | 4-5 |
| SPL/32 interface | 2-35 |
| START command | |
| authorized user utility | 3-5 |
| MTM | 5-2 |
| STATUS parameter | 5-5 |
| SUBMIT parameter | 5-7 |

| | |
|----------------------------|------|
| Subtask scheduling | |
| dynamic | 2-32 |
| priority | 2-32 |
| PSM | 2-33 |
| Subtasks of MTM | 1-1 |
| Sysgen options | |
| MTM | 2-4 |
| OS/32 | 1-3 |
| Sysgen parameter file | 2-2 |
| SYSSTRUC.MLB macro library | 2-4 |
| System generation | |
| MTM | 2-1 |
| | 4-2 |
| OS/32 | 1-3 |
| | 4-2 |
| System requirements | 1-2 |
| | 4-1 |

| | |
|----------------------|------|
| T | |
| Task date and time | 4-8 |
| Task queue | 2-24 |
| Task termination | 4-11 |
| TCB counter overflow | 4-11 |
| TELL parameter | 5-5 |
| TERMINAL parameter | 5-2 |
| | 5-5 |
| Terminal prompt | |
| device name | 2-21 |
| sequence | 2-22 |
| TERMINATE parameter | 5-6 |
| Time formats | 4-3 |
| TOTAL command | 3-26 |

| | |
|---------------------------|------|
| U | |
| UCPULIM macro | 2-40 |
| UIOLIM macro | 2-42 |
| UPDATE mode | 3-5 |
| | 3-7 |
| USED parameter | 5-5 |
| User identifier | 4-6 |
| | 4-8 |
| User supplied information | 4-10 |

| | |
|---------------|------|
| V | |
| Variables | |
| global | 2-36 |
| internal | 2-38 |
| local | 2-36 |
| new global | 2-37 |
| VERSION macro | 2-45 |

| | |
|-----------------|-----|
| W-Z | |
| Wait time | 4-8 |
| WAITS parameter | 5-5 |

PUBLICATION COMMENT FORM

We try to make our publications easy to understand and free of errors. Our users are an integral source of information for improving future revisions. Please use this postage paid form to send us comments, corrections, suggestions, ect.

1. Publication number _____

2. Title of publication _____

3. Describe, providing page numbers, any technical errors you found. Attach additional sheet if necessary.

4. Was the publication easy to understand? If not, why?

5. Were illustrations adequate? _____

6. What additions or deletions would you suggest? _____

7. Other comments: _____

From _____ Date _____

Position/Title _____

Company _____

Address _____

STAPLE

STAPLE

FOLD

FOLD



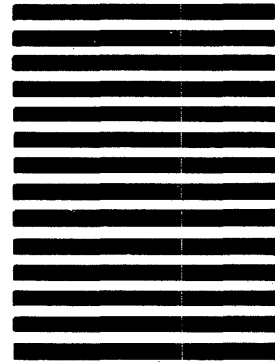
NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 22 OCEANPORT, N.J.

POSTAGE WILL BE PAID BY ADDRESSEE

PERKIN-ELMER

Data Systems Group
106 Apple Street
Tinton Falls, NJ 07724



**ATTN:
TECHNICAL SYSTEMS PUBLICATIONS DEPT.**

FOLD

FOLD

STAPLE

STAPLE