



**PUBLICATIONS  
UPDATE**

**Operating System/3 (OS/3)**

**System Activity Monitor (SAM)**

**User Guide/Programmer  
Reference**

**UP-9983-A**

**This Library Memo announces the release and availability of Updating Package A to "SPERRY® Operating System/3 (OS/3) System Activity Monitor (SAM) User Guide/Programmer Reference", UP-9983.**

The system activity monitor (SAM) provides a means of evaluating system performance. This manual describes the system activities and events that can be monitored, the statistical information collected, and the types of reports that can be produced from those statistics. Included are sample monitoring sessions and examples of how to use each statistic to analyze and remedy system performance problems.

This update documents for release 9.0 changes in or clarification of co-channeling and:

- I/O class readings
- DSK class items

Copies of Updating Package A are now available from requisitioning. Either the updating package only or the complete manual with the updating package may be requisitioned by your local Sperry representative. To receive only the updating package, order UP-9983-A. To receive the complete manual, order UP-9983.

**LIBRARY MEMO ONLY**

Mailing Lists  
BZ, CZ, and MZ

**LIBRARY MEMO AND ATTACHMENTS**

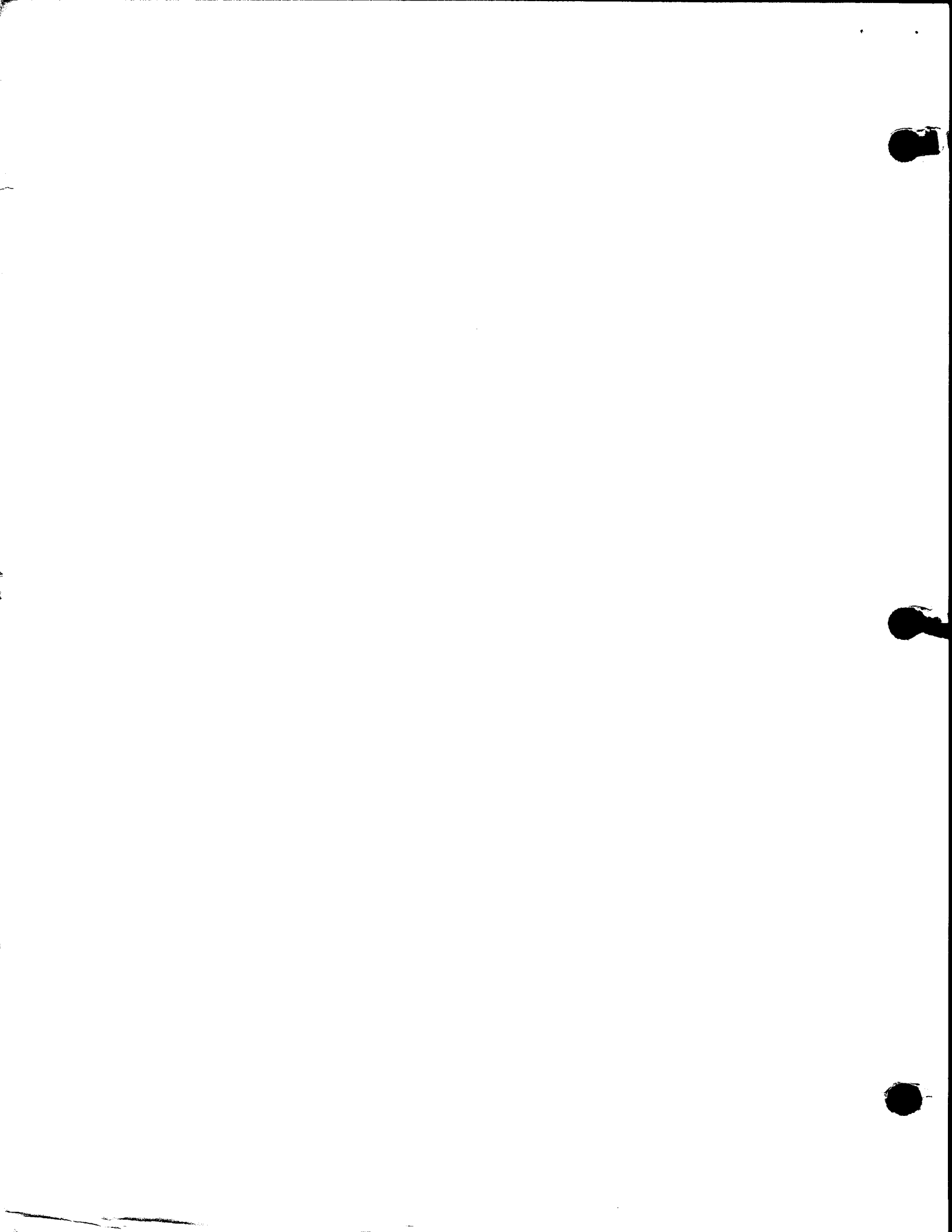
Mailing Lists B00, B01, 28U, and 29U  
(Package A to UP-9983,  
7 pages plus Memo)

**THIS SHEET IS**

Library Memo for  
UP-9983-A

RELEASE DATE:

January, 1985



**UNISYS**

**OS/3**

**System Activity  
Monitor**

**Programming  
Guide**

Relative to Release  
Level 9.0

Priced Item

August 1987

Printed in U S America  
UP-9983



**UNISYS**

**OS/3**

**System Activity  
Monitor**

**Programming  
Guide**

Copyright© 1987 Unisys Corporation  
All Rights Reserved  
Unisys is a trademark of Unisys Corporation.  
Previous Title: System Activity Monitor User Guide/  
Programmer Reference

Relative to Release  
Level 9.0

August 1987

Priced Item

Printed in U S America  
UP-9983

NO WARRANTIES OF ANY NATURE ARE EXTENDED BY THE DOCUMENT. Any product and related material disclosed herein are only furnished pursuant and subject to the terms and conditions of a duly executed Program Product License or Agreement to purchase or lease equipment. The only warranties made by Unisys, if any, with respect to the products described in this document are set forth in such License or Agreement. Unisys cannot accept any financial or other responsibility that may be the result of your use of the information in this document or software material, including direct, indirect, special or consequential damages.

You should be very careful to ensure that the use of this information and/or software material complies with the laws, rules, and regulations of the jurisdictions with respect to which it is used.

The information contained herein is subject to change without notice. Revisions may be issued to advise of such changes and/or additions.

FASTRAND, ✦SPERRY, SPERRY✦UNIVAC, SPERRY, SPERRY UNIVAC, UNISCOPE, UNISERVO, UNIS, UNIVAC, and ✦ are registered trademarks of Unisys Corporation. ESCORT, PAGEWRITER, PIXIE, PC/IT, PC/HT, PC/microIT, SPERRYLINK, and USERNET are additional trademarks of Unisys Corporation. MAPPER is a registered trademark and service mark of Unisys Corporation. CUSTOMCARE is a service mark of Unisys Corporation.

**PAGE STATUS SUMMARY**

**ISSUE: Update B – UP-9983**  
**RELEASE LEVEL: 9.0 Forward**

Part/Section	Page Number	Update Level
Cover		B
Title Page/Disclaimer		B*
PSS	1	B
Preface	1, 2	Orig.
Contents	1 thru 4	Orig.
<b>PART 1</b>	Title Page	Orig.
1	1 thru 5	Orig.
2	1 thru 19	Orig.
<b>PART 2</b>	Title Page	Orig.
3	1 thru 25	Orig.
<b>PART 3</b>	Title Page	Orig.
4	1 thru 5 6, 7 8 thru 14	Orig. A Orig.
<b>PART 4</b>	Title Page	Orig.
Appendix A	1, 2	Orig.
User Comment Form		

Part/Section	Page Number	Update Level

Part/Section	Page Number	Update Level

\*New page

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





## Preface

This manual is intended to guide the system administrator and installation manager in the use of the SPERRY Operating System/3 (OS/3) System Activity Monitor. It describes the procedures for using the system activity monitor and the report producing program SAMRPT. Also, it discusses the type of data measurements monitored and provides a logical guide that will aid you in improving your system's performance.

Before reading this manual, you should be familiar with your system so you can understand the possible benefits of the system activity monitor. You can obtain more information about OS/3 in the following list of manuals (current versions).

- System Description, UP-8800
- System Concepts and Facilities, UP-8870
- System Installation User Guide/Programmer Reference, UP-8839
- System Hardware and Software Summary, UP-8868
- System Hardware and Software Summary, UP-8203
- Supervisor Concepts and Facilities, UP-8831
- Integrated Communications Access Method Concepts and Facilities, UP-8194

The system activity monitor will become even more useful as you gain knowledge of your system's performance and as you gain more experience interpreting the results the system activity monitor gives you. You are responsible for evaluating and interpreting the system activity monitor reports. Sperry assumes no responsibility for the interpretation of the results. You must assess any benefits you may gain from using the system activity monitor yourself.

The guidelines set forth in this manual are a general presentation of the system activity monitor; some of the examples given may not apply in every case. The manual cites many factors that can affect system performance; you must apply your individual results to your system to determine the system's performance problems and possible solutions. Also, try not to draw conclusions from only one run of the system activity monitor, as its data may not represent your system's total work load.

The information contained in this manual is presented in the following parts, each containing one or more sections.

■ **PART 1. THE SYSTEM ACTIVITY MONITOR**

Part 1 describes what the system activity monitor is, what it does, and how to use it. This part introduces the activities and events that can be monitored, the statistics generated, and the ways in which they can be monitored. It also describes the control commands and parameters used to gather these statistics.

■ **PART 2. GENERATING A REPORT OF MONITORED STATISTICS**

Part 2 describes the function and use of the report producing program SAMRPT. Included is a discussion of the types of reports that can be produced, a description of the parameters used to generate them, and examples of how to execute SAMRPT.

■ **PART 3. ANALYZING SYSTEM PERFORMANCE**

Part 3 discusses how the monitored statistics can be used to analyze and improve system performance. It also describes the advantages of the two monitoring modes (class and trace) and their possible applications. Included are examples of how you could use each statistic to analyze a problem and determine its remedy.

■ **PART 4. APPENDIXES**

Appendix A lists the I/O command code subsets most commonly used for devices supported by OS/3, specifically for System 80, Models 3 through 6.

# Contents

## PAGE STATUS SUMMARY

## PREFACE

## CONTENTS

### PART 1. THE SYSTEM ACTIVITY MONITOR

#### 1. INTRODUCTION

1.1.	WHAT IS THE SYSTEM ACTIVITY MONITOR?	1-1
1.2.	SYSTEM ACTIVITY MONITOR COMPONENTS	1-2
1.3.	WHAT CAN YOU MONITOR?	1-2
1.3.1.	Class Mode Monitoring	1-2
1.3.2.	Trace Event Mode Monitoring	1-4
1.3.3.	FIPLAN Mode	1-5
1.4.	GENERATING THE SYSTEM ACTIVITY MONITOR IN YOUR SYSTEM	1-5

#### 2. CONTROLLING THE SYSTEM ACTIVITY MONITOR

2.1.	COMMAND AND PARAMETER SPECIFICATIONS	2-1
2.2.	LOADING AND ACTIVATING THE SYSTEM ACTIVITY MONITOR	2-2
2.3.	DEFINING THE MONITORING CRITERIA DURING THE INITIAL LOAD AND ACTIVATION OF THE MONITOR	2-4
2.3.1.	O Parameter – Directing the Monitored Output	2-5
2.3.1.1.	Creating Output File Space for the System Activity Monitor	2-7
2.3.2.	C Parameter – Selecting Class Monitoring	2-8
2.3.3.	T Parameter – Selecting Trace Event Monitoring	2-10
2.3.4.	F Command – Selecting FIPLAN Mode	2-10
2.3.5.	I Parameter – Defining Class Monitoring Intervals	2-11

2.4.	STOPPING AND STARTING MONITOR ACTIVITY AT ANY TIME DURING THE SESSION	2-12
2.5.	CHANGING THE CRITERIA AND OVERRIDING THE DEFAULT FUNCTIONS DURING THE MONITORING SESSION	2-13
2.6.	DISPLAYING THE CURRENT STATUS OF THE MONITOR	2-14
2.7.	CONTROLLING THE DISPLAY OF CLASS OUTPUT DATA	2-15
2.8.	TERMINATING YOUR MONITORING SESSION	2-16
2.9.	SAMPLE MONITORING SESSIONS	2-17

## PART 2. GENERATING A REPORT OF MONITORED STATISTICS

### 3. SYSTEM ACTIVITY MONITOR REPORT (SAMRPT)

3.1.	WHAT SAMRPT PRODUCES	3-1
3.1.1.	Final Tabular Report	3-2
3.1.2.	Intermediate Tabular Report	3-4
3.1.3.	Histogram Report	3-6
3.1.4.	Subfile Directory Listing	3-10
3.2.	EXECUTING SAMRPT	3-10
3.2.1.	Executing SAMRPT via a Canned Job Stream	3-11
3.2.2.	Executing SAMRPT via Your Own Job Stream	3-12
3.3.	SAMRPT PARAMETERS	3-14
3.3.1.	Specifying the Time Range within a Subfile to Be Processed (TME)	3-15
3.3.2.	Specifying Monitored Classes for Reports (RPT)	3-17
3.3.3.	Producing Histogram Reports (HIS)	3-18
3.3.4.	Specifying a Particular Time Interval (IVL)	3-19
3.3.5.	Specifying Particular Devices (DVC)	3-20
3.3.6.	Report Qualification by Job (JOB)	3-21
3.3.7.	Listing of Job/DVC Information (LST)	3-22
3.4.	SAMPLE SAMRPT JOB STREAM	3-24

---

**PART 3. ANALYZING SYSTEM PERFORMANCE****4. USING THE MEASUREMENT STATISTICS**

<b>4.1.</b>	<b>GENERAL</b>	<b>4-1</b>
4.1.1.	Collecting the Data	4-1
<b>4.2.</b>	<b>CLASS ITEM MEASUREMENTS</b>	<b>4-2</b>
4.2.1.	<b>CPU Class Items</b>	<b>4-2</b>
4.2.1.1.	IDLE	4-3
4.2.1.2.	WAIT	4-3
4.2.1.3.	TCPU	4-3
4.2.1.4.	CPIO	4-3
4.2.1.5.	KEYS (KEYn, KEmm)	4-4
4.2.1.6.	SPVR	4-5
4.2.1.7.	Interrupt Rates	4-5
4.2.2.	<b>I/O Class Readings</b>	<b>4-6</b>
4.2.2.1.	RATE	4-6
4.2.2.2.	QUED	4-6
4.2.2.3.	ERRC	4-6
4.2.3.	<b>DSK Class Items</b>	<b>4-7</b>
4.2.3.1.	DISK	4-7
4.2.3.2.	SEEK	4-7
4.2.3.3.	CYLS	4-8
4.2.3.4.	Using DSK Class with DSK Trace	4-8
4.2.4.	<b>MEM Class Items</b>	<b>4-8</b>
4.2.4.1.	NOAJ	4-8
4.2.4.2.	NOJS	4-8
4.2.4.3.	TRRC	4-9
4.2.4.4.	TRRQ	4-9
4.2.4.5.	TRRL	4-9
4.2.4.6.	TRUC	4-9
4.2.4.7.	TRRU	4-9
4.2.4.8.	MEMS	4-10
4.2.4.9.	HITR	4-10
4.2.4.10.	LGOF	4-10
4.2.4.11.	LGON	4-10
4.2.5.	<b>SVC Class Items</b>	<b>4-11</b>
4.2.5.1.	SVCR	4-11
4.2.5.2.	SCVP	4-11
4.2.6.	<b>COM Class Items</b>	<b>4-11</b>
4.2.6.1.	CIMR	4-12
4.2.6.2.	COMR	4-12
4.2.6.3.	CSEN	4-12
4.2.6.4.	CERR	4-12
4.2.6.5.	CNOT	4-12
4.2.6.6.	CINT	4-13
4.2.6.7.	CPOL	4-13
4.2.6.8.	CBYT	4-13
<b>4.3.</b>	<b>TRACE EVENT MEASUREMENTS</b>	<b>4-13</b>
4.3.1.	I/O Event Trace (T=I/O)	4-14
4.3.2.	Disk Event Trace (T=DSK)	4-14

---

**PART 4. APPENDIXES****A. I/O COMMAND CODES SUBSET (MODELS 3 THROUGH 6)****USER COMMENT SHEET****FIGURES**

1-1.	Overview of System Activity Monitor	1-1
2-1.	System Activity Monitor Output Display	2-3
3-1.	Final Tabular Report for Class Data	3-3
3-2.	I/O Trace Event Report	3-5
3-3.	Histogram Report for Class Data	3-7
3-4.	Disk Trace Event Report	3-9
3-5.	Subfile Directory Listing	3-10
4-1.	Sample Daily Production Work Loads	4-4

**TABLES**

1-1.	Monitoring Classes	1-3
2-1.	Commands to Control and Use the System Activity Monitor	2-1
2-2.	Parameters to Control and Use the System Activity Monitor	2-2

**PART 1. THE SYSTEM ACTIVITY MONITOR**

THE UNIVERSITY OF CHICAGO  
DIVISION OF THE PHYSICAL SCIENCES  
DEPARTMENT OF CHEMISTRY  
5708 SOUTH WOODLAND AVENUE  
CHICAGO, ILLINOIS 60637



# 1. Introduction

## 1.1. WHAT IS THE SYSTEM ACTIVITY MONITOR?

The system activity monitor (SAM) is a system symbiont that you run to monitor and record your system's activity. It is intended to be used by the system administrator and installation manager to aid in the detection of production bottlenecks, to optimize production job mixes, and to identify system variables that can be changed to influence system performance.

After analyzing and evaluating the report produced by the system activity monitor, you can apply this information (in some cases) through certain system parameters to achieve system optimization.

Figure 1-1 shows some performance criteria that can be monitored with the system activity monitor.

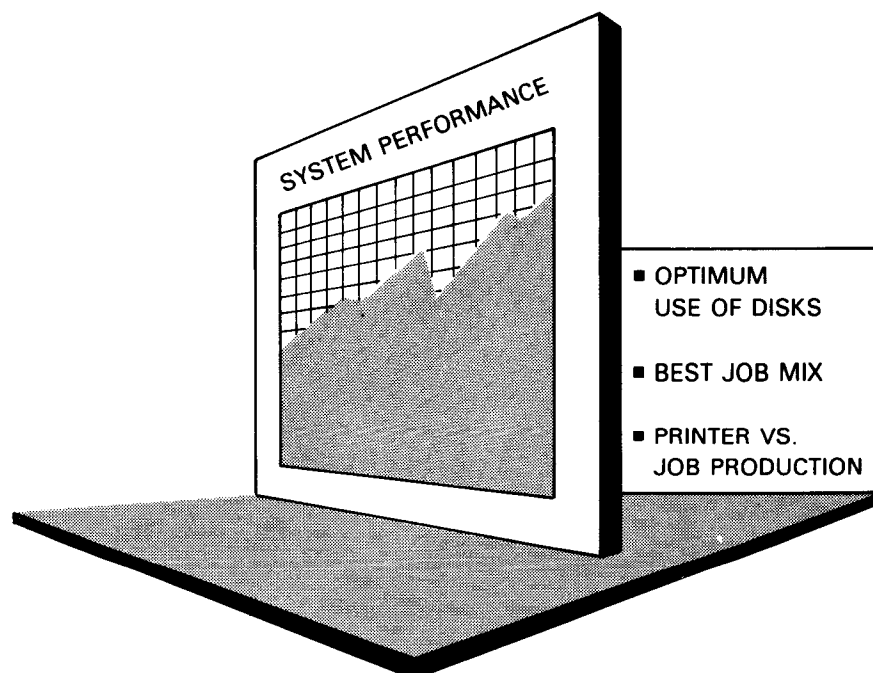


Figure 1-1. Overview of System Activity Monitor

## 1.2. SYSTEM ACTIVITY MONITOR COMPONENTS

The system activity monitor consists of two components: a data collection symbiont and a stand-alone report program called SAMRPT.

The data collection symbiont is dynamically linked to the operating system. You control its activity by keying in various control parameters from the system console. The data collected is accumulated and output to a disk file, where a number of logical subfiles can be created, each containing data from a different monitoring session. The accumulated data can also be presented in an abbreviated form as a real-time display on the system console. After the data is collected, you terminate the data collection symbiont and the monitor is cleared and ready for the next session.

The report producing program, SAMRPT, is a stand-alone program that prints a summary of the data collected by the data collection symbiont. You can create your own job stream to execute SAMRPT or run a canned job control stream by keying in RV SAMRPT on the system console. By creating your own job stream, you can tailor your reports, selecting data on the basis of subfiles, time of day, and measurement classes. You can direct SAMRPT to produce tabular statistical reports for entire classes, and time histograms for specific class items. The canned job control stream lets you produce summary final reports for all monitored classes, and subfile directory listings.

## 1.3. WHAT CAN YOU MONITOR?

The system activity monitor functions in two distinct modes: class monitoring and trace event monitoring. In either mode, the monitor continuously collects data about the activity of your OS/3 system. In addition, the system activity monitor has a special mode to provide data for FIPLAN, the file placement analyzer.

### 1.3.1. Class Mode Monitoring

In class mode, the system activity monitor continuously accumulates real time system profile data. This data is written to a disk file and, if you desire, displayed in abbreviated form on the system console. In either case, the data is written and displayed at fixed time intervals that you have selected. Table 1-1 lists the six classes that you can select for monitoring and their associated measurement items.

Table 1-1. Monitoring Classes (Part 1 of 2)

	Item	Purpose
CPU	IDLE WAIT TCPU CPIO SPVR KEYn KEmm KEYS SVCR I/OR TMER OTHR INTR	Measures CPU activity Indicates percent of time system was inactive or waiting for communication devices Indicates percent of time CPU was inactive while at least one peripheral device was busy Indicates percent of time that the supervisor or some task was actively processing Indicates percent of time CPU was active while at least one peripheral device was busy Indicates percent of time supervisor was doing interrupt processing Indicates percent of time a CPU key was active (n=1-digit hexadecimal key number) Indicates percent of time a CPU key was active (mm=2-digit decimal key number) Indicates total percent of time for all keys being active Indicates number of times per second that supervisor call interrupts occurred Indicates number of times per second that I/O interrupts occurred Indicates number of times per second that timer interrupts occurred Indicates number of times per second that all other interrupts occurred Indicates total interrupt rate
I/O	RATE QUED ERRC	Measures activity of I/O devices. All items are reported by the device ID (DID) with a maximum of 116 devices and by channel number. References by channel number are displayed on system console. Indicates number of times per second I/O requests were issued per channel or device Indicates percent of start I/O commands delayed because of an I/O channel or device being busy Indicates number of times per second error recovery was required
DSK	DISK SEEK CYLS	Measures activity on disk devices. All items are reported by DID (maximum of 48 disks). Indicates number of times per second I/O orders were issued per channel or device Indicates percent of I/O orders issued for separate seek operations Indicates average number of cylinders moved for each separate seek operation
MEM	LGOF LGON NOAJ NOJS TRRC TRRQ TRRL TRUC TRRU HITR  MEMS, { BUFP SHCD SYMB JOBS FREE MAXF CBUF }	Measures activity of system main storage Indicates number of workstation logoffs per interval Indicates number of workstation logons per interval Indicates average number of jobs active per interval Indicates number of jobs started per interval Indicates number of times per second transients were requested Indicates percent of transient calls queued due to lack of available transient areas Indicates percent of transient calls that required loading Indicates percent of transient calls that requested reusable transients Indicates percent of transients that were actually reused Indicates percent of disk cache "hits" for the specified monitoring interval  Indicates number of bytes of main storage allocated to buffer pools, shared code, symbionts, jobs, free (available) main storage space, maximum contiguous free main storage area, and cache buffer size. The default selection is FREE.
SVC	SVCR SCVP	Measures occurrences of individual types of supervisor calls Indicates number of supervisor calls by type Indicates number of EXCP type supervisor calls issued to virtual devices for a spool file

Table 1-1. Monitoring Classes (Part 2 of 2)

Class	Item	Purpose
COM	CIMR	Measures activity of system communications. All items can be referenced by port number. Indicates by line the number of times per second messages were received by ICAM
	COMR	Indicates by line the number of times per second messages were transmitted by ICAM
	CSEN	Indicates number of sense commands issued by ICAM requesting further status information regarding interrupts marked as bad status completion
	CERR	Indicates number of error conditions (time-outs, negative acknowledgments (DLE-NAK) or reply request (DLE-ENQ))
	CNOT	Indicates number of no-traffic responses received
	CINT	Indicates number of interrupts serviced other than those defined above
	CPOL	Indicates rate (per second) of polling interrupts
CBYT	Indicates number of bytes transmitted (per second) over the communications line.	

## NOTE:

Items under the I/O class are referenced by channel number and device ID (DID). Statistics by channel are shown by default. When reporting statistics for I/O and DSK class activity on systems having co-channels, the device ID (DID) will always begin with the primary channel number. In addition, I/O and disk devices that you want SAM to monitor under I/O and DSK classes must have a PUB address of 116 or less. Otherwise, they cannot be monitored. We recommend that all real devices you intend to monitor be given a PUB address of 116 or less during I/OGEN phase of system generation.

### 1.3.2. Trace Event Mode Monitoring

Trace event mode gives you an in-depth look at particular classes. Trace mode is limited to monitoring only one type of event per session. It differs from class mode's method of gathering time and count statistics and recording them at periodic intervals. However, it collects a more detailed level of information about that event. The data records generated from the collected information are buffered and written to a disk file for offline reduction. This data can then be correlated to particular jobs or categories of system tasks for statistical analysis. The following events can be selected for trace event monitoring:

- I/O

Traces all I/O requests by device with command code distribution, byte transfer counts, and requesting job information for each request; and

- DSK

Traces all disk requests by device with requesting cylinder address, providing a file reference pattern by device.

### 1.3.3. FIPLAN Mode

In FIPLAN mode, SAM collects data to be used as input to FIPLAN (file placement analyzer). FIPLAN is a performance tool that can analyze this data and determine a file allocation scheme that optimizes your system's performance if you reallocate your files accordingly.

Note that FIPLAN data files are acceptable input to the standard DSK trace reporting of SAMRPT.

For more information about FIPLAN, see the file placement analyzer user guide/programmer reference, UP-9731 (current version).

## 1.4. GENERATING THE SYSTEM ACTIVITY MONITOR IN YOUR SYSTEM

Before using the system activity monitor, you must establish certain links to your operating system and to your ICAM network if you plan to use the monitor in a communications environment. You establish these links via specific keyword parameters defined during supervisor generation (SUPGEN) and ICAM generation (COMMCT).

The supervisor generation keyword parameter, SAM=YES, provides links for the system activity monitor to the OS/3 system.

To establish links for the system activity monitor to your communications environment, specify the CCA macro keyword parameter, FEATURES=(MONITOR), when generating your ICAM network. This specification links the necessary system activity monitor module to the ICAM network being generated, enabling the monitor to gather ICAM related statistics.

For more information on generating the system activity monitor, see the following manuals:

- System installation user guide/programmer reference, UP-8839 (current version)
- ICAM network definition and operations user guide, UP-8947 (current version)



## 2. Controlling the System Activity Monitor

### 2.1. COMMAND AND PARAMETER SPECIFICATIONS

Control and use of the system activity monitor is directed through a series of commands and parameter specifications you key in from the system console. The control commands let you load the monitor, initiate the monitoring activity, and halt and restart the monitoring activity. The parameter specifications let you select the monitoring mode and define the criteria under which monitoring is performed. The commands and parameter specifications used with the system activity monitor are provided in Tables 2-1 and 2-2, respectively. The formats and methods of using these commands and parameter specifications are included as part of the functional descriptions provided in this section.

*Table 2-1. Commands to Control and Use the System Activity Monitor*

Command	Function
SAM	Loads the basic version of the system activity monitor
SAX	Loads a larger or extended version of the system activity monitor without any class exclusion restrictions
GO	Initiates or reactivates system activity monitoring
STOP	Stops system activity monitoring so you can change the monitoring criteria or override the default functions established during monitoring initialization
STA	Displays the current operating status (monitoring criteria) of the system activity monitor
D	Controls the display of monitored data on the system console screen. (The display can be turned on and off and allowed to roll off the screen.)
EOJ	Ends the monitoring session, stops monitoring, closes the logical and physical file, and terminates the system activity monitor

Table 2-2. Parameters to Control and Use the System Activity Monitor

Parameter	Function
O	Specifies disk output file to which the monitored data is written. Also specifies whether an abbreviated form of the output is to be displayed at the system console. This parameter is optional for class mode monitoring and required for trace event mode monitoring
C	Activates class mode for monitoring and also specifies the classes to be monitored. This is the default monitoring mode of the system activity monitor
T	Activates trace event mode for monitoring and specifies the event to be monitored
F	Activates a special mode whereby SAM generates data to be analyzed by FIPLAN for a file allocation scheme to optimize your system's performance.
I	Specifies the intervals at which monitored data is recorded and displayed. This command can be used only with class (C) monitoring. If used with trace event (T) monitoring, it is ignored.

## 2.2. LOADING AND ACTIVATING THE SYSTEM ACTIVITY MONITOR

Loading and activating the system activity monitor can be as easy as keying in only two commands from the system console: a load command and a GO command. The load command brings the system activity monitor into main storage and the GO command activates it. You have a choice of two commands for loading the monitor: SAM and SAX.

The SAM command loads the basic version of the system activity monitor. It is the smaller of the two versions available. It lets you perform both class and trace event monitoring. However, it does have certain exclusions for class monitoring that are described in 2.3.3. It also provides smaller data buffers for trace event monitoring.

The SAX command loads an extended or larger version of the monitor. This version performs all the functions of the basic version without any of the class exclusion restrictions. It also provides larger data buffers for trace event monitoring.

The following keyins initially load and activate the system activity monitor:

```
SAM GO
or
SAX GO
```

As the load (SAM or SAX) and GO commands are entered, the following response messages appear on the screen of the system console, indicating the status of the monitor:

```
SAM00 LOADED
SAM01 ACTIVE (n)
```

where:

n

Is the number of the subfile currently opened for recording monitored output.



If you incorrectly key in a command, one of two messages appears at the system console:

```
SAM07 KEYIN ERR:n
      or
SAM07 KEYIN ERR:SNTAX
```

where:

n

Identifies the incorrect entry and SNTAX indicates a syntactical error in your command. To correct the error, simply reenter the corrected command.

**NOTE:**

*On a multiple command entry, all valid commands up to the error are processed.*

Loading and activating the monitor in this manner is a very basic approach to system activity monitoring. It does not let you define the criteria for monitoring but conducts the monitoring activity according to a set of preselected defaults. In this case, the monitor defaults to class mode of operation. The default class selection is ALL, but only the CPU and I/O classes are displayed at the system console. Figure 2-1 shows an example of the output displayed at the console when the monitor is operating under the default criteria.

```
+++ SAM +++ INTERVAL= 00:02:00/ ENDING AT 11:24:32
CPU= 84.1% KEYS= 39.7% WAIT= 15.9% IDLE= 0.0%
CH0 R= 46.6 Q= 14.8% E= 0.0 CH1 R= 4.2 Q= 0.0% E= 0.0
```

Figure 2-1. System Activity Monitor Output Display

LEGEND:

CPU	Percent of time CPU is busy
KEYS	Percent of time CPU is busy by keys
WAIT	Percent of time active TCBs are waiting
IDLE	Percent of idle time (no active TCBs or I/O requests)

NOTE:

The I/O statistics are:

CHn	Channel numbers for the statistics measured
R	The I/O request rate by channel
Q	The percent of requests queued by channel
E	The error recovery count by channel

If you want to exercise more control over the monitoring function, you can define the criteria for your session during the initial loading and activating of the monitor (see 2.3). Of course you also have the option of overriding the defaults or changing the criteria any time during the session (see 2.5).

### 2.3. DEFINING THE MONITORING CRITERIA DURING THE INITIAL LOAD AND ACTIVATION OF THE MONITOR

When initially loading and activating the system activity monitor, you have the option of defining the criteria for your monitoring session rather than accepting the default criteria provided by the monitor (see 2.2).

If you elect to define the criteria for your monitoring session, you key in parameter specifications from the system console. These parameter specifications (listed in Table 2-1) can be entered as part of the load/activate command sequence:

```
load command [parameters,] GO
```

Or, you can enter them as a series of separate commands:

```
load command
00 SA [parameters]
.
.
.
00 SA [parameters]
00 SA GO
```

Note that the load command can be entered as SA, SAM, or SAX and that the prefix 00 SA must be used when entering parameter specifications and commands separately. Otherwise, they will not be accepted as valid entries.

Both entry forms accomplish the same results, but one gives you more control over the actual start of monitoring activity. When you use the first form (load command [parameters,] GO), the monitor is loaded, but actual monitoring is delayed until all the parameter and default values are set. This could create a problem when you want to activate the monitor to coincide with the start of a particular event. The delay could hinder proper synchronization of the two events.

The second form of entry lets you load the monitor, set the parameter and default values, and, at the desired time, instantly activate the monitor by entering the GO command. Minimal delay occurs and synchronization of events is easily accomplished.

The parameter specifications are not positional and therefore do not have to be specified in the exact sequence described in this section.

### 2.3.1. O Parameter – Directing the Monitored Output

The O parameter directs the monitor to write its output to a disk file that you have previously allocated. It also specifies whether the output is to be displayed at the system console. The O parameter is required for trace event monitoring and optional for class monitoring. You may specify the O parameter **only once** during a monitoring session (from the initial load to termination, i.e., SA ... to 00 SA EOJ).

If you omit the O parameter when conducting class mode monitoring, the output will not be written to disk but will be displayed on the system console in the abbreviated form shown in Figure 2-1. If you omit this parameter when conducting trace event monitoring, you will receive an error message on the system console and monitoring will not start. Simply enter the O parameter and proceed.

The subparameters for this parameter specification are not positional. Therefore, commas need not be inserted in place of any parameters that you omit. The format of the O parameter is:

$$O=[D,] \left[ \begin{array}{l} \{XT\} \\ \{IN\} \end{array} \right] did[,filename]$$

where:

D

Indicates that the output is to be displayed at the system console in addition to being written to your specified disk file. Note that when conducting a class monitoring session in which CPU and I/O are among the classes monitored, the displayed output is abbreviated as shown in Figure 2-1. The data written to the disk file, however, contains all the classes specified for the monitoring session.

If you omit the D subparameter, the monitored output is not displayed, but is written to your disk file.

XT

Indicates that the output from this monitoring session is to be written to an existing file created for some previous monitoring session. The data previously recorded in the file is saved, the file is extended for the current session, and the new data is written as an added subfile. No more than eight subfiles may be created for any one file. The monitor uses this subparameter as the default if you do not specify how output is to be handled (by omitting both the XT and IN subparameters).

IN

Indicates that the output from this monitoring session is to be written as the first subfile of an existing file. All the subfiles previously recorded in the file are lost. If you omit this subparameter and do not specify the XT subparameter, the monitor uses XT as the default specification.

did

Specifies the device address of the disk on which your output file resides. If you specify an incorrect address, the response message

```
SAM07 KEYIN ERR:did
```

appears at the system console. To eliminate this error, enter a corrected O parameter with the correct device address.

filename

Specifies the name of a previously created output disk file to which the monitored output data for this session is written. If you omit the file name, the specified device is searched for a default file named SAMFILE in which to write the output. However, you must have previously allocated a SAMFILE to take advantage of this feature. If you specify an incorrect file name or, in the case of the default situation SAMFILE cannot be found, the response message

```
SAM03 OPEN ERR:03
```

is displayed at the system console, indicating that the output file cannot be opened. To correct the error, respecify the O parameter with the correct file name. If the output file does not exist, allocate an output file and reinitiate your monitoring session. If the specified file name is valid, the following response message is displayed at the system console:

```
SAM10 SFL n OPENED
```

where:

n

Is the subfile number.

#### NOTE:

*When you set up your output file or the default SAMFILE, remember that the system activity monitor requires a SAT disk file formatted in 256-byte sectors for reading and writing the output data. The file space should be contiguous. See 2.3.1.1 for calculating needed sectors for class and trace event monitoring.*

Here are three examples of the O parameter:

Example 1:

```
O=D,103
```

The system activity monitor records the output to disk and displays it on the system console as well. Since no file name is specified, the statistics are to be recorded to the default file name, SAMFILE, provided that you have previously

allocated an output file with that file name. Because neither XT nor IN is specified, the system activity monitor defaults to XT, the data subfiles from previous monitoring sessions are saved, and the file is logically extended for the current output (another subfile is opened) by the current monitoring session. The device address for the disk file containing the monitored output in this case is 103 (the hexadecimal device ID).

Example 2:

```
O=XT,103,TESTFILE
```

The output is recorded to disk only and is not displayed on the system console because the D parameter is not specified. Subfiles from previous monitoring sessions are saved and the file is logically extended to create another subfile for the current output as the XT parameter dictates. The monitored statistics are recorded on device 103 in the file TESTFILE.

Example 3:

```
O=IN,103,TESTFILE
```

The output is not displayed because the D parameter is not specified. The statistics are recorded on disk device 103 in the file named TESTFILE, which is initialized.

### 2.3.1.1. Creating Output File Space for the System Activity Monitor

The system activity monitor requires a disk file formatted to 256-byte sectors for reading and writing output data. When creating this file, you can allocate and format the proper file space by using the following extent statement in a job control stream:

```
//EXT ST,CF,0,BLK,(256,n)
```

In this statement, the ST indicates that the system access technique is to be used, and the CF specifies contiguous formatted disk space. The zero specifies that no additional space is needed. Since space must be allocated in 256-byte blocks, you must specify BLK and 256 in the extent statement. The n in this statement is the number of 256-byte blocks.

Interactively, you can allocate file space through the use of this command:

```
AL ST,filename,vsn,SIZE=m
```

The AL in this command allocates space, the ST indicates that the system access technique is to be used, and the file name is the name of your output file. The vsn is the volume serial number of the disk on which your output will reside.

For class monitoring, you can calculate the number of 256-byte blocks needed by using this formula:

```
number of blocks=(e/i * b) + 10%
```

where:

- e  
Is the expected measurement duration in minutes.
- i  
Is the measurement in intervals.
- b  
Is either 12 blocks per interval (using c=ALL) or 20 blocks per interval when extended (SAX) version is loaded. Other class selections require less space, which means that a higher limit is provided by this formula for the file size.

For trace event monitoring, the formula for estimating the system activity monitor output file size (number of 256-byte blocks) in SAM or SAX version is:

$$\text{number of blocks} = (r * e \div 16) + 10\%$$

where:

- r  
Is the estimated rate of event.
- e  
Is the estimated time of monitor session in seconds.

**NOTE:**

*For efficiency reasons, you should locate the output file on the least active volume mounted.*

### **2.3.2. C Parameter – Selecting Class Monitoring**

The C parameter places the system activity monitor into the class monitoring mode of operation. It also lists the classes to be monitored and specifies whether these classes are to be included with, deleted from, or used to replace those classes previously defined for monitoring.

Class monitoring is completely distinct from trace event monitoring (see 2.3.2). So, you cannot perform both types of monitoring operations simultaneously. Table 1-1 defines the classes that you can specify for monitoring.

The specific classes that can be monitored at one time depend on which version of the monitor you load for the monitoring session. If you use the SAM command to load the monitor, you are operating under the basic version and therefore cannot monitor the I/O, SVC, and COM classes at the same time. These classes are mutually exclusive under the basic version of the monitor. Attempting to monitor any combination of these three classes at the same time under the basic version of the monitor results in the display of the following mutually exclusive error message at the system console:

SAM 07 KEYIN ERR: #MX#

The mutually exclusive restriction for the I/O, SVC, and COM classes does not apply when monitoring under the extended version of the monitor. To operate under the extended version of the system activity monitor, you must use the SAX command for loading the monitor.

The formats for the C parameter are:

Format 1 (monitor loaded by SAM command):

$$C = \left\{ \begin{array}{l} \text{ALL} \\ \left\{ \begin{array}{l} \text{I} \\ \text{X} \end{array} \right\} \left( \left[ \text{CPU} \right] \left[ \text{,DSK} \right] \left[ \text{,MEM} \right] \left[ \text{,} \left\{ \begin{array}{l} \text{I/O} \\ \text{SVC} \\ \text{COM} \end{array} \right\} \right] \right) \end{array} \right\}$$

Format 2 (monitor loaded by SAX command):

$$C = \left\{ \begin{array}{l} \text{ALL} \\ \left\{ \begin{array}{l} \text{I} \\ \text{X} \end{array} \right\} \left( \left[ \text{CPU} \right] \left[ \text{,DSK} \right] \left[ \text{,MEM} \right] \left[ \text{,I/O} \right] \left[ \text{,SVC} \right] \left[ \text{,COM} \right] \right) \end{array} \right\}$$

where:

ALL

Indicates that only the CPU, I/O, DSK, and MEM classes are monitored when operating under the basic version of the monitor. When operating under the extended version of the monitor, ALL indicates that the CPU, I/O, DSK, MEM, SVC, and COM classes are monitored.

$\left\{ \begin{array}{l} \text{I} \\ \text{X} \end{array} \right\}$

Indicates whether the class subparameters specified are to be included (I) with or excluded (X) from any classes previously indicated for this monitoring session. If you omit this subparameter, the classes specified replace all classes previously indicated for monitoring.

$\left( \left[ \text{CPU} \right] \left[ \text{,DSK} \right] \left[ \text{,MEM} \right] \left[ \text{,} \left\{ \begin{array}{l} \text{I/O} \\ \text{SVC} \\ \text{COM} \end{array} \right\} \right] \right)$  and  $\left( \left[ \text{CPU} \right] \left[ \text{,DSK} \right] \left[ \text{,MEM} \right] \left[ \text{,I/O} \right] \left[ \text{,SVC} \right] \left[ \text{,COM} \right] \right)$

Indicates the specific classes you want monitored. The classes listed replace any classes previously defined for monitoring unless the list is prefixed with the I or X subparameter. When prefixed with the I subparameter, the classes specified are included (added) with the list of classes previously specified for monitoring. When preceded with the X subparameter, the classes specified are excluded (deleted) from the list of classes previously defined for monitoring.

You need not code the parentheses if you are specifying only one class and both the I and X subparameters are omitted.

### 2.3.3. T Parameter – Selecting Trace Event Monitoring

The T parameter places the system activity monitor into the trace event mode of operation and defines the event to be monitored (traced). You may trace only one event per monitoring session. Trace event monitoring is completely distinct from class monitoring; you cannot perform both trace event monitoring and class monitoring simultaneously. If you omit the T parameter, the monitor defaults to class mode of operation. See 2.3.2 for details.

It is suggested that you load the extended version of the system activity monitor (SAX) when operating in trace mode; trace event monitoring is more efficient when larger data buffers are used. Refer to 2.2 for the use of the SAX version of the system activity monitor.

The format for the T parameter is:

$$T = \left\{ \begin{array}{l} I/O \\ DSK \end{array} \right\}$$

where:

T=I/O

Selects I/O events for trace monitoring. This monitoring mode records all occurrences of I/O to devices generated in your supervisor. It traces all I/O requests by device with command code and job information for each request.

T=DSK

Selects disk events for trace monitoring. This monitoring mode records all occurrences of I/O applicable to disk devices. It traces all disk requests by device with requested cylinder addresses, providing a cylinder reference pattern by device.

### 2.3.4. F Command – Selecting FIPLAN Mode

The F command places the system activity monitor in a special data collection mode. FIPLAN (the file plan analyzer) uses the data collected to determine a file allocation scheme that optimizes your system's performance.

The F command format is given here for illustration only. The actual application of this command is explained in the file placement analyzer user guide/programmer reference, UP-9731 (current version). If you intend to use the F command, it is strongly recommended that you consult that manual first.

The F command format is:

$$F = \left\{ \begin{array}{l} IN, \\ \end{array} \right\} did \left\{ \begin{array}{l} , filename \\ \end{array} \right\}$$



where:

IN

Initiates the output SAMFILE to the first subfile.

XT

Extends the output SAMFILE to the next subfile of the SAM output file.

did

Is the address of the disk on which your file resides.

filename

Is the user-defined output name of the previously allocated disk file.

SAMFILE

Is the previously allocated default output disk file, used when you do not specify a file name.

### 2.3.5. I Parameter – Defining Class Monitoring Intervals

The I parameter provides two monitoring interval specifications for class mode monitoring only. The first (interval) specifies the rate at which monitored data is written to disk and displayed at the system console. The second (number of intervals) limits the monitoring session period to a specific number of occurrences at the time interval specified. The format of the I parameter is:

$$I = \left\{ \begin{array}{l} \text{mmm} \\ \text{Sxx} \end{array} \right\} [ , \text{nnn} ]$$

where:

mmm

Is a decimal number of up to three digits representing the time of the monitoring interval in minutes.

Sxx

Is a 2-digit decimal number representing the time of the monitoring interval in seconds. The smallest interval is 10 seconds and the largest is 59 seconds.

nnn

Is a decimal number of up to three digits representing the number of occurrences at the specified time interval. If omitted, monitoring continues indefinitely or until the monitoring session is ended with a control command (see 2.1).

If you omit the I parameter, the time interval defaults to 2 minutes if you are displaying the output only at the system console, or 5 minutes if you are writing the output to disk (and optionally displaying it at the console). Monitoring continues indefinitely.

**NOTE:**

*When the specified occurrence limit is reached during the monitoring session, monitoring stops and the response message*

**SAM04 INACTIVE** (subfile number)

*appears at the console. This message indicates that monitoring has stopped and the subfile, if recording to disk, is closed. You can either reactivate monitoring by keying in 00 SA GO or terminate the session by keying in 00 SA EOJ.*

**2.4. STOPPING AND STARTING MONITOR ACTIVITY AT ANY TIME DURING THE SESSION**

You can stop and start the system activity monitor any time through the STOP and GO commands. To stop the monitoring activity, key in:

**00 SA STOP**

The following response message is displayed on the system console:

**SAM04 INACTIVE** (n)

This indicates that the monitoring activity has been halted.

To restart the activity, key in the GO command as follows:

**00 SA GO**

After you enter the GO command, this response message appears at the system console:

**SAM01 ACTIVE** (subfile number)

This indicates that the monitor has been reactivated and a new output subfile has been opened. If you want to redefine any monitoring criteria while monitoring is stopped, you must do so before issuing the GO command (see 2.5).

The STOP command does not end the monitoring session but puts the monitor in an idle state until you wish to restart it through the GO command. Only the activity of the monitor is halted with the STOP command; therefore, you can change parameter specifications and restart the activity if you wish. Keep in mind that the monitor only records information collected for a completed monitoring interval. Therefore, any time you issue a STOP command to interrupt a monitoring interval, the information collected during the interrupted interval is lost (not written to your output file). Using the STOP command closes the current output subfile and the GO command opens a new subfile. The load command is not used when reactivating the system activity monitor through the GO command because the monitor is already loaded; it is just in an idle state.

## 2.5. CHANGING THE CRITERIA AND OVERRIDING DEFAULT FUNCTIONS DURING THE MONITORING SESSION

After the system activity monitor is loaded and activated, you can change the monitoring criteria and override the default functions or change a previous selection. To do this, you must first halt the monitoring activity by keying in the following command:

```
00 SA STOP
```

The response message

```
SAM04 INACTIVE
```

appears on the system console, indicating that monitoring has been halted. Once monitoring is halted, you can change the monitoring criteria and reactivate the monitor by specifying the necessary parameters and entering the GO command via either of the following forms:

```
00 SA [parameters,] GO
```

or

```
00 SA [parameters]
```

```
.
```

```
.
```

```
.
```

```
00 SA GO
```

The parameter specifications used to change the monitoring criteria are the same in format and function as those described in 2.3.

Whether you key in these parameters and GO as a single entry or as a series of separate commands, monitoring activity begins again, with the new criteria in effect when the GO command is entered. The response message

```
SAM01 ACTIVE (subfile number)
```

appears at the system console, indicating that the monitoring activity is reactivated and the output subfile identified has been opened. The subfile number is 0 if your output is not being written to disk.

You can continue to change the monitoring criteria as often as you like. However, each time you must first enter the 00 SA STOP command to halt the monitor. If you try to key in parameters or activate the monitor without first stopping the monitor, the response message

```
SAM07 KEYIN ERR: #ON#
```

appears on the system console, indicating that the monitor is currently active and must be stopped before any new parameters are entered. To correct this error, simply key in the 00 SA STOP command and reenter the required parameters.

If you decide not to change the criteria after stopping the monitor, simply reactivate the monitor by use of a single 00 SA GO command, and the most recently specified parameters and defaults will be used as the monitoring criteria.

## 2.6. DISPLAYING THE CURRENT STATUS OF THE MONITOR

You can display the system activity monitor's current operating status at any time while it is loaded by keying in the status command (STA) from the system console. The information displayed at the console relates to whether the monitor is active or inactive, lists the current criteria for monitoring, and identifies the output device and file used to record the monitored data if applicable.

To use the status display function, key in the command

```
00 SA STA
```

The following status information appears on the screen of the console in response to the command:

```
SAM08 ST={ INACTIVE  
          { HH:MM:SS, } IT={ mmm } [ ,xxx ] [ ,C ] [ ,T ] [ ,did,n,filename ]  
                          { Sxx }
```

where:

ST=INACTIVE

Indicates that the monitor is currently inactive (stopped).

ST=HH:MM:SS

Indicates the clock time in hours, minutes, and seconds (HH:MM:SS) the monitor was started by using the GO function.

IT= { mmm }  
      { Sxx }

Indicates the monitoring time interval in minutes (mmm) or seconds (Sxx) that the monitored output data is being recorded and/or displayed. This field is applicable only if the monitor is operating in class mode (C parameter specified).

xxx

Indicates the number of intervals (occurrences of the specified time interval) to be recorded to the current output subfile. In fact, this field represents the number of data buffers written to the output file and is displayed for either class or trace mode. When the monitor is in class mode (C parameter specified), it also corresponds to the number of interval expirations, which is its primary use.

C

Indicates that the monitor is operating in class mode (classes being monitored). This field is displayed only when the monitor is operating in class mode (C parameter specified).

T

Indicates that the monitor is operating in trace mode (events being traced). This field is displayed only if the monitor is operating in trace mode (T parameter specified).

did,n,filename

Indicates the device address (did), the last opened subfile (n), and the name of the file being used to record monitored data. This field is displayed only when monitored data is being recorded to disk (O parameter is specified).

If you want a listing of the active classes being monitored, key in the command

```
00 SA STA,C
```

When this form of the status command is entered, the response message

```
SAM08 [CPU][I/O][DSK][MEM][SVC][COM]
```

appears at the system console, indicating those classes currently being monitored.

It is not necessary to stop the monitor before entering the status command. However, if the monitor is busy generating a data display output when the status command is entered, the message

```
SAM06 BUSY REENTER
```

appears at the console. Simply reenter the command to obtain the status.

## 2.7. CONTROLLING THE DISPLAY OF CLASS OUTPUT DATA

The display of SAM output data on the screen of the system console is usually controlled by using the display option in the output parameter (O=D) that you specified when establishing the criteria for your monitoring session. If, for some reason, you did not include the display option when setting up your output criteria, or if you want to override the display option during the session, you can do so by using the display (D) command. This command allows you, through a console keyin, to turn the display on and off and to roll the data off the screen before the established time interval is reached. The D command can be keyed in any time that the monitor is active; you need not stop the monitor before using this command. To issue the D command, key in:

```
00 SA D={ ON
          OFF
          ROLL }
```

where:

**D=ON**

Turns on the display of monitored class data at the system console. This keyin, if entered while the monitor is in trace (T) mode, results in the display of the SAM07 keyin error message.

**D=OFF**

Turns off the display of monitored class data. When this command is entered, the current monitored display is allowed to roll off the console screen and any new monitored data is prevented from being displayed until the ON command is entered. Do not use the OFF parameter unless data is also being recorded to disk (O=D specified). If you do, the monitor ignores the command and displays the SAM07 keyin error message.

**D=ROLL**

Allows the SAM data to roll off the screen before the time interval is reached. Normally, as monitored data appears on the system console, it rolls upward as each new message or command is displayed. Once it is at the top of the screen, the SAM data remains, and all other information rolls off the screen beneath the display. When the specified time interval is reached, the SAM display rolls off the screen as each new line of data appears on the system console. By entering the ROLL command, you allow the SAM data to roll off the screen normally before the next time interval is reached. Although the current data is allowed to roll off the screen when this command is entered, new monitored data continues to be displayed at the specified monitoring interval without being rolled off.

## 2.8. TERMINATING YOUR MONITORING SESSION

You terminate the system activity monitor, ending your monitoring session, by keying in the command:

```
00 SA EOJ
```

When this command is entered, the response messages

```
SAM04 INACTIVE(n)
SAM05 RELEASED
```

appear on the system console, indicating that monitoring has stopped, the output file (n) is closed (if recording), and all system resources are released.

Once the EOJ command is entered, unsolicited 00 SA commands cannot be entered. To perform monitoring after the EOJ command is entered, you must first reload and activate the system activity monitor.

## 2.9. SAMPLE MONITORING SESSIONS

The following sample monitoring sessions illustrate how you control monitoring via specific commands from the system console. Included in the sample sessions are examples of what happens if you enter a command incorrectly, and the response message indicating the error that appears on the system console screen. The message indicates either the incorrect entry or an error code.

### NOTE:

*The entry of 00 SA is mandatory.*

### Class Mode Monitoring

Command	Response	Explanation
SAM I=10, GO	SAM00 LOADED SAM01 ACTIVE	Monitors and displays default classes indefinitely at 10-minute intervals
00 SA STA	SAM08 ST=13:51:59, IT=10,,C	Displays the operating status of the monitor as of the latest entry
00 SA STOP	SAM04 INACTIVE	Halts the monitoring process, allowing criteria to be changed
00 SA L=D, 103, I=15, 10, GO	SAM07 KEYIN ERR:L=	Command is invalid because an illegal parameter (L) was specified.
00 SA O=D, 103, I=15, 10, GO	SAM10 SFL# 1 OPENED SAM01 ACTIVE (1)	Monitors and displays default classes and records default statistics to SAMFILE on device 103 for ten 15-minute intervals. The response message indicates that subfile #1 has been opened.
00 SA STOP	SAM04 INACTIVE (1)	Halts the monitoring process, allowing criteria to be changed
00 SA C=(COM,SVC), GO	SAM07 KEYIN ERR: #MX#	Response message indicates that the command is invalid since COM and SVC are mutually exclusive.

(continued)

**Class Mode Monitoring (cont)**

Command	Response	Explanation
00 SA C=X(MEM), GO	SAM01 ACTIVE	Monitors and displays default data and records all previously monitored classes except the MEM class. Default classes were established by the fifth command.
00 SA STA	SAM06 BUSY, REENTER	Monitor was busy generating display data to system console. Command should be reentered.
00 SA STA	SAM08 ST=14:16:38, IT=15,C,103,2, SAMFILE	Displays the operating status of the monitor as of the latest entry
00 SA D=OFF	No response	Turns off console display of monitored data
00 SA EOJ	SAM04 INACTIVE (2) SAM05 RELEASED	Terminates the monitoring session, closes output file (2), and releases system resources

**Trace Mode Monitoring**

Command	Response	Explanation
SAX	SAM00 LOADED	Larger version of SAM is loaded.
00 SA 0=103,IN, TRACEFILE	SAM10 SFL#1 OPENED	Monitor records statistics to TRACEFILE on device 103. The response message indicates that subfile #1 has been opened.
00 SA T=DSK,GO	SAM01 ACTIVE (1)	Monitor is active, tracing disk events.
00 SA STA	SAM08 ST=11:21:52, I=____,____,__,T,103, 1, TRACEFILE	Displays the operating status of the monitor as of the latest entry
00 SA STOP	SAM04 INACTIVE (1)	Halts the monitoring process, allowing criteria to be changed

(continued)



**Trace Mode Monitoring (cont)**

Command	Response	Explanation
00 SA R=I/O	SAM07 KEYIN ERR:R=	Command is invalid because an illegal parameter (R) is specified.
00 SA T=I/O,GO	SAM01 ACTIVE (2)	Monitor selects I/O events for trace event monitoring. GO activates monitoring.
00 SA EOJ	SAM04 INACTIVE (2) SAM05 RELEASED	Terminates the monitoring session, closes the output file (2), and releases system resources.

**Class to Trace Mode Monitoring**

Command	Response	Explanation
SAM I=10, GO	SAM00 LOADED SAM01 ACTIVE	Monitors and displays default classes indefinitely at 10-minute intervals
00 SA STA	SAM08 ST=13:51:59, IT=10,,C	Displays the operating status of the monitor as of the latest entry
00 SA STOP	SAM04 INACTIVE	Halts the monitoring process, allowing criteria to be changed
00 SA O=103, T=DSK, GO	SAM10 SFL#n OPENED SAM02 ACTIVE (n)	Monitoring begins in trace event mode. Subfile n is opened, and all previous class parameters are ignored once the system activity monitor is in trace event mode.
00 SA EOJ	SAM04 INACTIVE (n) SAM05 RELEASED	Terminates the monitoring session, closes the output file (n), and releases system resources



**PART 2. GENERATING A REPORT OF  
MONITORED STATISTICS**



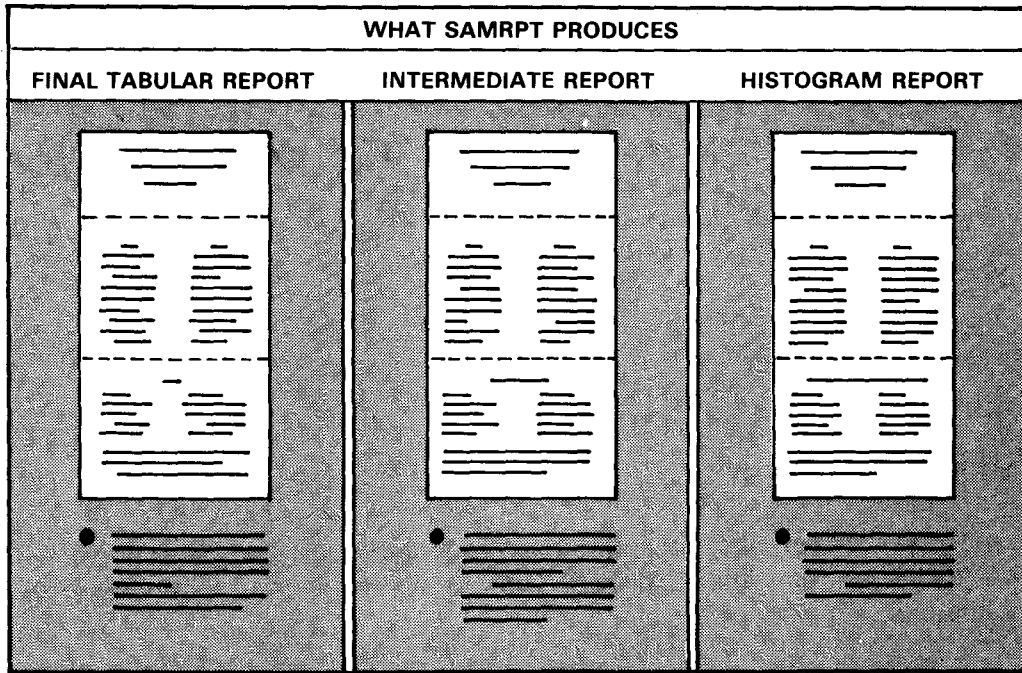
### 3. System Activity Monitor Report (SAMRPT)

#### 3.1. WHAT SAMRPT PRODUCES

Once the system activity monitor has collected the requested data and recorded it to your disk file, you can then use the SAMRPT routine to produce different reports presenting these statistics:

Report Type	Class Mode	I/O Trace	DSK Trace
Final tabular report	X	X	
Intermediate tabular report	X		
Histogram	X		X

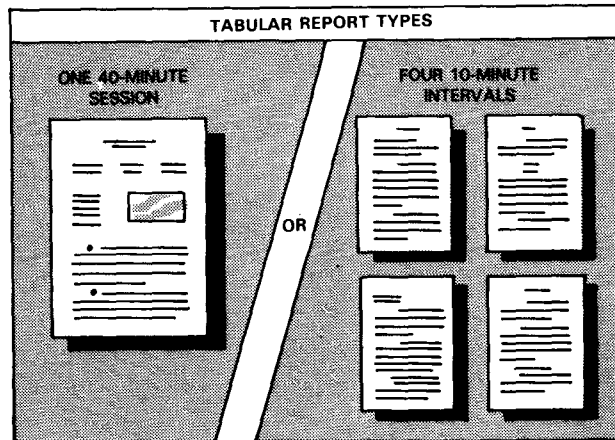
Final tabular reports show the collected statistics of a subfile monitored by the system activity monitor for all or specific classes within a subfile in class mode or only for I/O in trace mode. Intermediate tabular reports display the collected statistics at specific time intervals. Histograms show the system's activity as measured by either a disk in trace mode or a specific class item over a specified time interval within a subfile.



You can produce any one or a combination of these reports for a specific subfile (data from a specific monitoring session) or for all the subfiles on a particular system activity monitor file.

### 3.1.1. Final Tabular Report

You can use the SAMRPT routine to generate a final tabular report showing measurement statistics for all or specific classes monitored and recorded to a particular subfile. Normally, the statistics in the final tabular report are measurements of your system's activity over the elapsed time of the monitoring session (time length of the subfile). However, you can also produce a final tabular report for a specified time range within a subfile.



For example, if data was recorded to subfile 1 for five 1-minute intervals, a final tabular report showing your system's activity for the entire 5-minute monitoring session would be produced for subfile 1. If, however, you specified a particular time range within subfile 1, a final tabular report for that time range (multiples of 1 minute) would be produced for subfile 1.

Figure 3-1 shows an example of a final tabular report for class data.

SPERRY OS/3 SYSTEM ACTIVITY MONITOR						
DATA OBTAINED FROM S80 OS/3 - CECG ON 03/06/71 (SFL# 1)						
++ FINAL ++ REPORT PERIOD = 17.00 MIN. - FROM 15:33:15 TO 15:50:15						
++ RECORDED AT 1.0 MIN. INTERVALS ++						
----- CPU -----						
		PERCENT			INTERRUPT/SEC	
	REPORT PERIOD	MIN	MAX	REPORT PERIOD	MIN	MAX
WAIT	61.18	20.60	88.06	I/OE	16.61	3.53 23.51
IDLE	2.87	0.0	48.71	SVCR	74.77	27.53 151.69
SPVR	17.62	5.85	31.18	TMR	0.10	0.09 0.11
CPIO	30.26	11.63	48.50	CTHR	0.0	0.0 0.0
TCPU	35.95	11.94	60.74	TOTL	91.49	
KEY 0	13.20	6.06	29.57			
KEY 1	4.91	0.0	16.48			
KEY 2	0.20	0.0	7.33			
KEY 3	0.01	0.0	0.15			
KEY 4	0.01	0.0	0.16			
----- CHN -----						
	I/O'S PER SECOND			%QUEUED	ERROR COUNT	
CHANNEL	REPORT PERIOD	MIN	MAX			
1	12.38	2.53	19.93	47.73	1	
3	4.04	0.11	13.24	0.0	1	
----- I/O -----						
	I/O'S PER SECOND			%QUEUED	ERROR COUNT	
DEVICE	REPORT PERIOD	MIN	MAX			
310	0.33	0.0	0.78	0.0	0	
330	3.21	0.0	12.33	0.0	1	
103	12.04	2.03	19.86	49.02	1	
104	0.02	0.0	0.14	0.0	0	
105	0.31	0.0	0.88	0.64	0	
340	0.29	0.0	2.46	0.0	0	
311	0.08	0.0	0.41	0.0	0	
312	0.13	0.0	0.46	0.0	0	

Figure 3-1. Final Tabular Report for Class Data

The first four lines comprise a header that appears on all reports produced by the SAMRPT routine. This header shows such information as the report date, the subfile that was processed, the report type, the time period from which data was reported, and the recording interval length.

The report in Figure 3-1 shows measurement statistics for the CPU and I/O class. Notice that the I/O class readings are divided into two sections: I/Os by channel and I/Os by device.

Each statistic shown in the final tabular report for subfile 1 represents an average value for the entire 17-minute report period. Since readings were recorded to subfile 1 for every 1-minute interval and the report period is 17 minutes, each final value on the report is an average of the five 1-minute readings. Also listed on the report are minimum and maximum interval values for various class items.

**NOTES:**

1. *For final tabular reports that automatically follow intermediate reports, asterisks are printed in place of minimum and maximum values.*
2. *For classes where data is reported by ID or type, measurement values are suppressed if no activity was observed. If a value of zero is reported, activity was observed but was less than 0.005.*

In trace event mode, a tabular report is produced for I/O trace only. This report contains the standard SAMRPT heading information followed by a list of the I/O requests issued; the I/O requests are listed by command code and device identity. If the report is for requests qualified for a specific job, a statement with the specified job name will appear following the heading. If the report is tailored to a specific device type, only requests to devices of that type will be reported. Figure 3-2 shows an example of this type of tabular report.

**NOTE:**

*Any device id=000 with a total count equal to zero should be ignored. This is a result of the standard report format and the columns are therefore redundant.*

The parameters used to request a final tabular report and specify classes and time ranges are discussed in 3.3.

### **3.1.2. Intermediate Tabular Report**

You can use the SAMRPT routine to generate intermediate tabular reports for class data showing measurement statistics for all or specific classes monitored and recorded to a particular subfile. Intermediate tabular reports are very similar to final tabular reports (see Figure 3-1). However, intermediate tabular reports are printed for each specific time interval, rather than the entire time length of the subfile.



SPERRY OS/3 SYSTEM ACTIVITY MONITOR  
 DATA FROM S80 OS/3 - 0800 ON 830114 (SFL # 1)  
 TRACE DATA REPORT FOR 5.56 MIN., STARTING AT 13:15: 3  
 REPORT OF I/O COMMAND CODES

DEVICE =	310		331		104		105		340	
CMD CODE	#REQ./AV.LTH	#REQ./AV.LTH	#REQ./AV.LTH	#REQ./AV.LTH	#REQ./AV.LTH	#REQ./AV.LTH	#REQ./AV.LTH	#REQ./AV.LTH	#REQ./AV.LTH	#REQ./AV.LTH
01	100	77	0	0	0	0	0	0	0	0
02	22	22	0	0	1914	873	41	1173	1599	253
04	0	0	0	0	0	0	0	0	6	6
05	0	0	0	0	2321	322	8	256	0	0
07	0	0	0	0	0	0	0	0	5	0
09	0	0	2207	117	0	0	0	0	0	0
0F	0	0	32	0	0	0	0	0	0	0
11	0	0	1655	120	0	0	0	0	0	0
17	0	0	20	0	0	0	0	0	0	0
19	0	0	14	120	0	0	0	0	0	0
1F	0	0	6	0	0	0	0	0	0	0
22	0	0	0	0	183	49	55	42	0	0
27	0	0	0	0	0	0	0	0	0	0
2F	0	0	7	0	0	0	0	0	0	0
32	44	2	0	0	20	245	0	0	0	0
0F	0	0	15	0	0	0	0	0	0	0
DF	0	0	114	0	0	0	0	0	0	0
TOTAL	166	50	4070	113	4438	548	104	504	1612	151

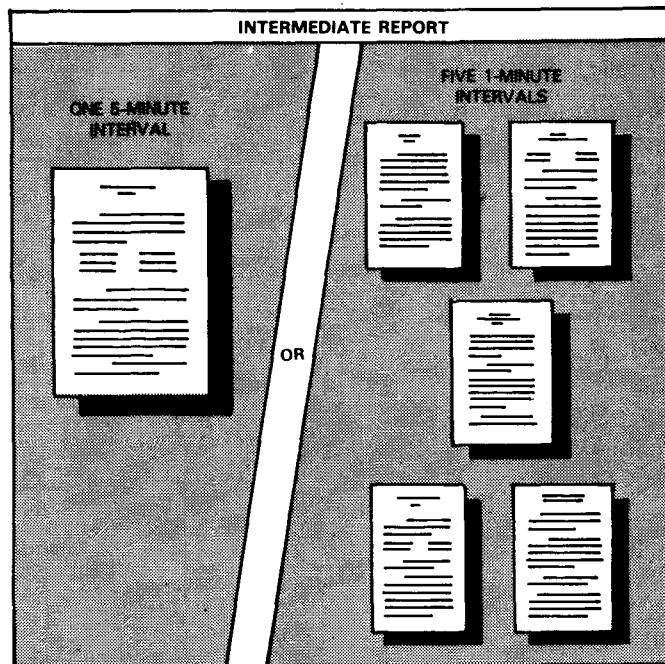
DEVICE =	314		322		323		100		101	
CMD CODE	#REQ./AV.LTH	#REQ./AV.LTH	#REQ./AV.LTH	#REQ./AV.LTH	#REQ./AV.LTH	#REQ./AV.LTH	#REQ./AV.LTH	#REQ./AV.LTH	#REQ./AV.LTH	#REQ./AV.LTH
01	2	80	0	0	0	0	0	0	0	0
02	2	14	0	0	0	0	0	0	0	0
07	2	0	0	0	0	0	0	0	0	0
09	39	163	0	0	0	0	0	0	0	0
0A	35	18	0	0	0	0	0	0	0	0
23	2	0	0	0	0	0	0	0	0	0
32	43	2	0	0	0	0	0	0	0	0
43	3	0	0	0	0	0	0	0	0	0
TOTAL	128	56	0	0	0	0	0	0	0	0

SRP04 === SAMRPT COMPLETED

NOTE:

The first 48 command codes are individually represented. If there are more than 48 command codes, they will be included in OTHERS.

Figure 3-2. I/O Trace Event Report



For example, suppose the duration of the subfile is 5 minutes and you specified a time interval of 1 minute when requesting intermediate reports. The SAMRPT routine will generate five intermediate tabular reports, one for each interval.

After all the intermediate reports for a particular subfile are produced, a final tabular report for that subfile is also generated. However, asterisks are printed on this report instead of minimum and maximum values.

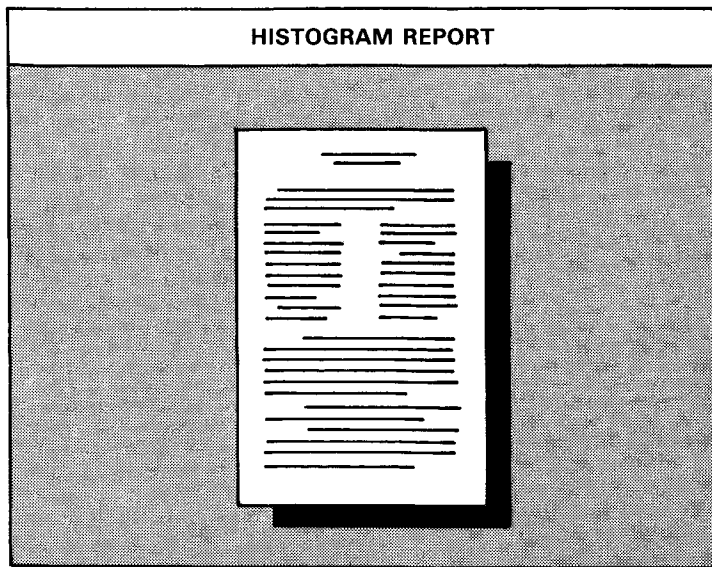
You can produce intermediate tabular reports for each recording interval within the subfile, or you can specify larger time intervals.

For example, suppose data was recorded to subfile 2 at four 10-minute intervals. If you specified a time interval of 10 minutes, intermediate reports for each of the four 10-minute intervals would be produced. If however, you specify a time interval of 20 minutes when executing SAMRPT, an intermediate report for each of the two 20-minute intervals is produced.

The parameters used to request intermediate tabular reports and specify classes and time intervals are discussed in 3.3.

### 3.1.3. Histogram Report

You can use the SAMRPT routine to produce histogram reports that graphically show the system's activity as measured by a specific monitored class item or trace event.



Histogram reports show measurements for each recording time interval within a particular subfile unless a specific time interval is requested. Figure 3-3 is an example of a histogram report for class data showing the number of times per second that supervisor call interrupts occurred.

```

      SPERRY + OS/3 SYSTEM ACTIVITY MONITOR
DATA OBTAINED FROM S80 OS/3 - 0800 S1 ON 82/03/01 (SFL# 3)
REPORT PERIOD = 2.84 MIN. - FROM 10:52:53 TO 10:55:43
++ RECORDED AT =.2 MIN. INTERVALS ++
      SUBFILE 3 CLASS REPORT
ACTIVITY FOR CLASS = CPU ITEM = SVCR
PLOT VALUE * = RATE OF 2.0/SEC TIME INTERVAL = 0.2 MIN

```

END TIME	RATE	1	2	3	4	5	6	7	8	9	0	CUM. AVG.
10:53: 3	30.30	.....										30.30
10:53:13	64.50	.....										47.40
10:53:23	79.50	.....										58.10
10:53:33	58.10	.....										58.10
10:53:43	70.00	.....										60.48
10:53:53	76.30	.....										63.12
10:54: 3	40.00	.....										59.81
10:54:13	69.70	.....										61.05
10:54:23	74.10	.....										62.50
10:54:33	82.10	.....										64.46
10:54:43	80.30	.....										65.90
10:54:53	27.30	.....										62.68
10:55: 3	59.60	.....										62.45
10:55:13	79.50	.....										63.66
10:55:23	86.20	.....										65.17
10:55:33	70.40	.....										65.49
10:55:43	41.80	.....										64.10

Figure 3-3. Histogram Report for Class Data

In Figure 3-3, the first five lines constitute a header that includes the date and time that the data for the histogram report was recorded. The following lines indicate the name of the item being reported, in this case, the supervisor call interrupts (SVCR) from the CPU class, the plot value of the asterisk (\*), and the time interval at which data was generated.

The x coordinate of the histogram shows the number of occurrences per second (RATE). In this example, the x coordinate measures the SVC rate. The y coordinate (TIME) of the histogram shows the time at which data is generated. In this example, since the time interval is 5 minutes, each time line shows monitored data for a particular 5-minute interval. The data under the RATE column lists the actual rate for a particular time interval, and the data under the CUM AVG. column shows the cumulative average rate from one time line to the next.

**NOTES:**

- 1. Histogram time lines are calculated by adding the histogram time line interval to the end time of the previous interval. If, however, monitoring was delayed (the SRP13 message will indicate a delay), the time lines shown in the histogram will not reflect this delay. Also, if the last monitored interval is less than the time line interval, it will not be printed.*
- 2. When plotting the average number of cylinders moved, values for count and cumulative average may not precisely match report values due to slight variations in calculation techniques.*

In trace event mode, a histogram is produced for DSK trace event only. This report contains the standard SAMRPT heading information followed by a graphic plot of accesses or references to each cylinder or the particular disk device being reported. Figure 3-4 is an example of a histogram report for trace event data.

SPERRY OS/3 SYSTEM ACTIVITY MONITOR  
 DATA FROM S80 OS/3 - 0800 S1 ON 820305 (SFL#4)  
 TRACE DATA REPORT FOR 6.88 MIN., STARTING AT 10:43:23  
 SUBFILE 4 DISK TRACE REPORT

REPORT OF ACCESSES TO DEVICE 103      PLOT VALUE \* = 1 PERCENT

CYLADR	PCT.											FREQ.	
		1	5	0	5	0	5	0	5	0	5		
72	0.10	.	-										8
104	0.95	.	*										74
106	0.21	.	-										16
107	2.02	.	**										157
108	0.75	.	*										58
114	6.83	.	*****										531
115	2.61	.	***										203
116	7.89	.	*****										613
118	4.94	.	*****										384
119	3.49	.	***										271
120	5.28	.	*****										410
121	4.85	.	*****										377
122	5.23	.	*****										406
123	4.57	.	*****										355
124	1.56	.	**										121
125	0.62	.	*										48
127	0.19	.	-										15
128	1.75	.	**										136
129	0.60	.	*										47
132	0.21	.	-										16
133	15.48	.	*****										1203
140	11.49	.	*****										893
141	1.16	.	*										90
143	0.41	.	-										32
169	9.59	.	*****										745
170	1.65	.	**										128
207	0.03	.	-										2
304	1.08	.	*										84
402	0.01	.	-										1
403	0.26	.	-										20
404	0.59	.	*										46
405	0.05	.	-										4
485	0.82	.	*										64
569	2.63	.	***										204
609	0.08	.	-										6
633	0.01	.	-										1
654	0.01	.	-										1

.....

TOTAL ACCESSES FOR DEVICE 103 = 7770  
 EST. NO. SEEKS = 5347 FOR 68.82% OF ACCESSES  
 AVG. NO. CYLINDERS MOVED = 63.4, MIN. = 1 MAX = 521

Figure 3-4. Disk Trace Event Report

### 3.1.4. Subfile Directory Listing

You can obtain a subfile directory listing for a specific file that shows the subfile number, creation date, classes monitored, and starting and elapsed monitoring times for each subfile. You can use this listing as a guide when selecting specific classes, time ranges, and time intervals for subfile processing. Figure 3-5 shows a typical subfile directory listing.

```

SRP00   === OS/3 SAMRPT R2.0 S2  STARTED
SFL=LST

                SPERRY - OS/3 - SYSTEM ACTIVITY MONITOR
                <===== SUB-FILE DIRECTORY LISTING =====>
SFL   CREATION   START      ELAPSED      RECORD      NO.      CLASSES:
NO.   DATE       TIME        TIME         INTRVL      BLOCKS   CIDMSC
  1    820301    10:46:40    0: 2:50     0           184      T
  2    820301    10:49:45    0: 2:51     0           156      T
  3    820301    10:52:53    0: 2:50     10          104      XXXX
  4    820305    10:43:23    0: 6:52     0           429      T
END OF SFL=LST - 874 TOTAL BLOCKS USED

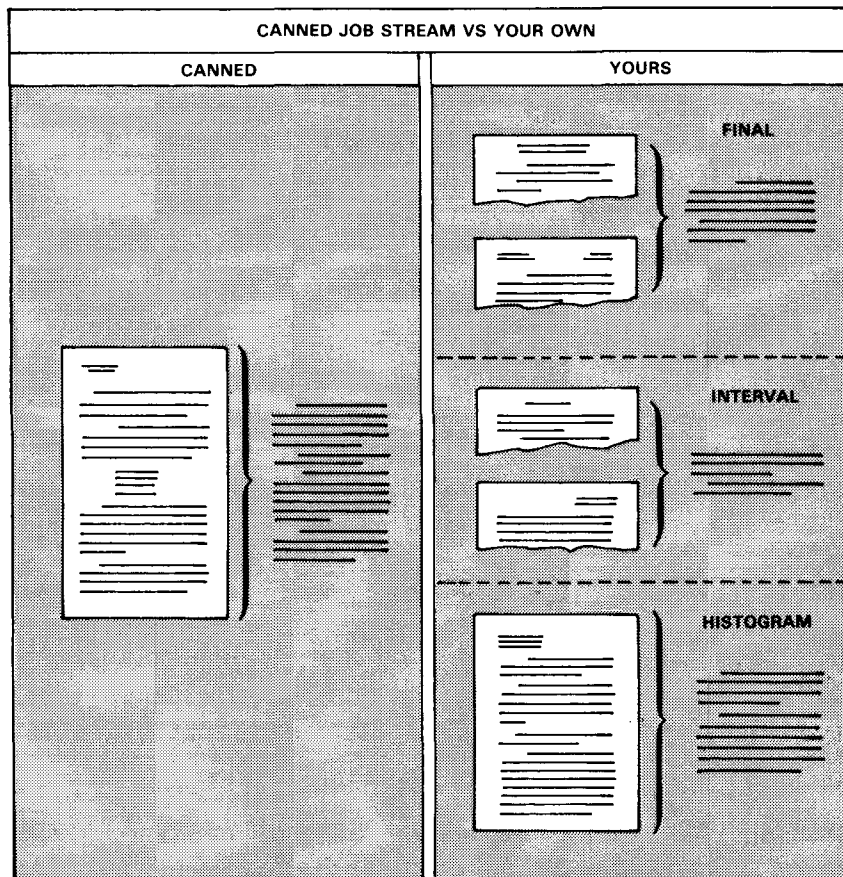
```

Figure 3-5. Subfile Directory Listing

In Figure 3-5, CIDMSC represents the monitored classes: C is CPU, I is I/O, D is DSK, M is MEM, S is SVC, and C is COM. The Xs under CIDMSC indicate which classes were monitored. In SFL NO. 3, for example, the Xs show that CPU, I/O, DSK, and MEM classes were monitored for that subfile. If Ts appear in the classes column, trace events were monitored, e.g., SFL NO. 1. (The Ts would appear under the I for I/O or D for DSK columns only because these are the only trace events that are possible.)

## 3.2. EXECUTING SAMRPT

You can execute SAMRPT from the system console by using a canned job stream, or you can create your own job streams. SAMRPT runs as a batch job using either method.



If you use the canned job control stream, you are limited to producing final tabular reports for all monitored classes or subfile directory listings showing the subfile number, creation date, classes monitored, and start and elapsed monitoring times. By creating your own job control stream, you can produce final tabular reports, intermediate tabular reports, and histogram reports; produce subfile directory listings; and specify particular classes, time ranges, and time intervals.

### 3.2.1. Executing SAMRPT via a Canned Job Stream

You can run SAMRPT via a canned job stream to produce the system activity monitor report. However, you are limited to producing trace event reports (tabular report for I/O or histogram for DSK) or final tabular reports for all monitored classes or subfile directory listings. Any final tabular report produced for a particular subfile by using the canned job stream shows all class statistics measured over the elapsed time of the subfile. To produce final tabular reports for specific classes and time ranges, you must create your own job stream by using specific parameters. The format of the canned job stream is:

```
RV SAMRPT,,V=vol-ser-no. [ ,F={filename} ] [ ,SFL={ALL} ]
                        [ {SAMFILE} ] [ {LST} ]
```

where:

V=vol-ser-no.

Specifies the volume serial number (one to six characters) of the disk pack containing the system activity monitor file.

F=filename

Specifies the name of the file containing the monitored data.

F=SAMFILE

Specifies that the file SAMFILE contains the monitored data. If you omit the file name, the SAMRPT routine searches the specified volume for a file named SAMFILE, and, if found, automatically processes this file. If SAMFILE or the specified file name is not found, an error message (SRP01) is generated.

SFL=ALL

Specifies that a final tabular class report will be produced for monitored classes or a trace event report will be produced. The type of data in the subfile determines which report is produced. If the subfile contains class data, a tabular class report results; if the subfile contains trace data, a trace event tabular report for I/O trace or histogram for DSK trace results.

SFL=LST

Specifies that a subfile directory listing will be produced showing the subfile number, creation date, classes monitored, and start and elapsed monitoring times for each subfile. When LST is specified, no other reports are produced.

**NOTE:**

*If a system failure or abnormal termination occurs during a monitoring session, the subfile being produced will not be closed. By executing SAMRPT through the system console, the subfile is automatically closed to recover any data in that subfile. This is also applicable when SAMRPT is run batch or as a background job.*

If you omit the SFL parameter or specify SFL= without specifying any subparameters, either a final tabular class report or trace event report is produced, depending on the type of data in the subfile from the last subfile created.

### 3.2.2. Executing SAMRPT via Your Own Job Stream

You can create your own job stream on cards, using the EXEC SAMRPT statement to produce the system activity monitor report. When creating this job stream, you can select which subfiles you want processed, which monitored classes or trace events you want reported, the type or types of reports you want produced, and specific time ranges and intervals within a subfile. You can also request a subfile directory listing. When executing SAMRPT through your own job stream, specify the job size on your JOB card as noted in the SRD.



The format of this job stream is:

```

1. // JOB SAMRPT,,xxxxx (where xxxxx is job size as defined in SRD)
2. // DVC 20 // LFD PRNTR
3. // DVC 50 // VOL D00028 // LBL SAMTEST // LFD SAMIN
4. // EXEC SAMRPT
5. /&

```

The first line in this job control stream is your job statement, which specifies the name of your job. Line 2 is the device assignment set for the printer. You must always specify the printer as your output medium when running SAMRPT. The device assignment set for the disk you are using for input is shown in line 3. In this example, the volume serial number is D00028, and the file label is SAMTEST. The device number for disk is DVC 50. The LFD name SAMIN must always be specified in the device assignment set for disk. Line 4 is the EXEC SAMRPT statement used to execute the SAMRPT routine. The /& in line 5 indicates the end of the job control stream. Running this job stream will cause a final tabular report or histogram to be produced. A class report will result if the last subfile created on the specified file contains class data, and a trace event report will be produced if the last subfile contains trace event data.

You can produce specific reports for particular subfiles by inserting various SAMRPT parameters in this job stream. The format is:

```

1. // JOB SAMRPT,,xxxxx (where xxxxx is job size as defined in SRD)
2. // DVC 20 // LFD PRNTR
3. // DVC 50 // VOL D00028 // LBL SAMTEST // LFD SAMIN
4. // EXEC SAMRPT
5. /$
   .
   .
   .
   [one or more parameter sets]
   .
   .
   .
6. /*
7. /&

```

Lines 1 through 4 are the same as the job control stream without parameters. Line 5 signifies the start of the embedded parameters, and line 6 indicates the end. The /& in line 7 signifies the end of the job control stream.

You can indicate the subfile or subfiles you want processed and, if desired, specify additional parameters to produce specific reports for these subfiles. The subfile specification parameter and any parameters used to control the processing of the specified subfile are called a parameter set. Each parameter set must begin with a subfile specification parameter (SFL) indicating the subfile or subfiles being processed. The subfile specification parameter acts as a delimiter for each parameter set and initiates the production of reports.

The following shows how you would list parameter sets in your job stream:

```
parameter set { /$
               { SFL
               { parameters
parameter set { SFL
               { parameters
               .
               .
               .
parameter set { SFL
               { parameters
               /*
               /*
               /*
```

Here, if you omit the first subfile specification (SFL), but specify parameters, the last subfile created on the specified file will be processed according to the specified parameters. If you include the first subfile specification and omit the parameters, a final report is produced of the specified subfile. SAMRPT will determine if class or trace data is in the subfile and then produce the appropriate report.

You can indicate as many parameter sets in the job stream as you need. However, once a set of parameters is specified for a subfile specification, they remain in effect for all subsequent subfile specifications until they are changed by new parameters. The SFL parameter used in this job stream has the same format as the one used in the canned job stream. However, if you are specifying a specific subfile for processing (SFL=n), you can also include an optional heading (up to 60 characters) that will be printed on any report produced for that subfile. This heading will print on trace event reports as well as on monitored class reports. When a heading is specified, the format of the SFL parameter is:

```
SFL=n [,heading]
```

You can use the SFL=LST parameter to produce a subfile directory listing as shown in 3.1.4.

### 3.3. SAMRPT PARAMETERS

In addition to the SFL parameter, you can use the following parameters to control the output of reports:

- TME

Specifies the time range within a subfile to be processed.

For monitor class reports:

- RPT  
Specifies monitored classes to be processed.
- HIS  
Indicates histograms are to be processed.
- IVL  
Specifies time interval for intermediate reports and histograms.

For trace event reports:

- DVC  
Specifies report for specific device or device types.
- JOB  
Specifies report for a specific user job.
- LST  
Provides for special listing options.

If you are executing SAMRPT and no parameters are specified, defaulted reports will be provided. These are:

- monitor classes (tabular report with RPT=ALL); and
- trace events (tabular report for I/O trace or histogram for disk trace).

**NOTE:**

*If you specify class parameters for a trace mode subfile or trace parameters for a class mode subfile, you will not receive an error message and the parameters will be ignored.*

### **3.3.1. Specifying the Time Range within a Subfile to Be Processed (TME)**

You can use the TME parameter to process a specific portion of either a class or trace subfile. The TME parameter specifies a particular time range for processing within the elapsed time range of the subfile. The format is:

TME=[start-time][,end-time]

where:

**start-time**

Specifies the start time in hours, minutes and, if wanted, seconds (hhmm[ss]) for the subfile being processed. If the requested start time is omitted or less than the original start time of the specified subfile, the original start time of the subfile is assumed.

**end-time**

Specifies the end time in hours, minutes and, if wanted, seconds (hhmm[ss]) for the subfile being processed. If the requested end time is omitted or greater than the original end time of the specified subfile, the original end time of the subfile is assumed.

**NOTES:**

1. *You can obtain a subfile directory listing indicating the start and elapsed times of all the subfiles in a particular file by specifying SFL=LST in your job stream. You can use these times as a guide when choosing a specific time range.*
2. *The TME parameter also acts as a delimiter for a parameter set. Therefore, consecutive TME parameters produce consecutive reports for the respective time range specified, all for the same subfile as long as there is no intervening SFL parameter.*

*Be aware that the TME parameter is not carried over to any subsequent parameter sets; if the same time frame is subsequently desired, the TME parameter must be explicitly redefined.*

3. *When time selection is specified, the actual report start time will be a time equal to or greater than the requested start time. The report end time will be equal to or less than the requested end time. In no case will a report span a time outside the requested time range.*
4. *The SAM collection symbiont always maintains increasing clock values for time stamping intervals, even when crossing midnight. For example, if a monitor session is started at 11:00 p.m. (23:00:00 hours) and ends at 1:00 a.m., the end time of the session will be 25:00:00 hours, regardless of the SUPGEN DAYCHANGE parameter selection. To process this subfile, the TME = parameter would be TME = 2300,2500.*

### 3.3.2. Specifying Monitored Classes for Reports (RPT)

You can use the RPT parameter to specify which monitored classes you want listed in tabular report form. When you are processing more than one subfile in your job stream, you must be sure to specify the classes you want reported for each subfile. If you specify particular classes for a subfile and these classes were not monitored and recorded to the subfile, they will not appear in the requested report for that subfile.

The RPT parameter also gives you the option of using an adjusted base time when reporting system activity. The adjusted base time represents the elapsed time of the monitoring session minus the time your system was idle during that session. Thus, it enables you to produce reports that show system activity measured over the precise time your work load was actually executing. The format is:

$$\text{RPT} = \left. \begin{array}{l} \text{ADJ} \\ \text{ALL[,ADJ]} \\ \text{class name[,...class names][,ADJ]} \\ \text{NO} \end{array} \right\}$$

where:

**ADJ**

Specifies that class statistics are to be reported by using the adjusted base time. If you specify this parameter for the first subfile, all monitored classes are reported by using the adjusted base time. If, however, you specify this parameter for any subsequent subfiles, the classes specified in the previous RPT parameter are reported by using the adjusted base time.

**ALL[,ADJ]**

Specifies that all monitored classes will be reported for the subfile being processed. If ADJ is specified, the classes will be reported by using the adjusted base time.

**class name[,...class names][,ADJ]**

Specifies which monitored classes will be reported for the subfile being processed. You should select only those classes actually monitored and recorded to this subfile. If ADJ is specified, each class is reported by using the adjusted base time. The first column of Table 1-1 lists the class names.

**NO**

Specifies that no tabular reports are to be produced for the subfile being processed.

**NOTE:**

*In most cases, if an error is detected after the RPT parameter is specified, RPT=NO is set.*

If you omit the RPT parameter for the first subfile of monitored class data, the default of RPT=ALL will be used. The RPT parameter specified for the first subfile or its default remains in effect for all subsequent subfile specifications until a new RPT parameter is specified. Therefore, if you omit an RPT parameter for any subsequent subfiles, it will not default. Instead, the classes established by the previous RPT parameter remain in effect. RPT parameter specifications or defaults are not affected by intervening trace event subfiles and remain in effect for all class data subfiles until explicitly reset.

### 3.3.3. Producing Histogram Reports (HIS)

You can use the HIS parameter to produce histogram reports for a specific class item. However, only four histogram reports per subfile specification can be produced. The format is:

$$\text{HIS} = \left\{ \begin{array}{l} \text{item} \left\{ \begin{array}{l} , \text{dvc-addr.} \\ , \text{CHn} \\ , \text{svc-code} [, \text{HI}] \\ , \text{line-no.} [, \text{channel no.}] \\ , \text{mem-type} \\ , \text{HI} \end{array} \right\} \\ \text{mem-type} \\ \text{NO} \end{array} \right\}$$

where:

#### item

Specifies the item name (see item column in Table 1-1) for the histogram being produced. The item name must precede any channel, number, or code specification.

The CPU class item OTHR is not to be used when producing histogram reports and, therefore, should not be specified with the HIS parameter.

#### dvc-addr.

Specifies the 3-digit hexadecimal device address of the device being monitored. You use this subparameter only when specifying an item that measures specific device activity.

#### CHn

Specifies the 1-digit hexadecimal channel number of the channel being monitored. You can use this subparameter only when specifying an item that measures channel activity (for example, RATE).

#### svc-code

Specifies, by number (up to three decimal digits), the type of supervisor call being monitored. You can use this subparameter only when specifying the item SVCR.

**HI**

Specifies a 500-per-second histogram plot scale for INTR and SVCR items only. This is useful when these items have a high frequency, because it changes the histogram plot scale from the standard 100 per second. You can use this parameter only when specifying the items INTR or SVCR.

**line-no.**

Specifies the number of the line being monitored. You can use this subparameter only when specifying the items in the COM class.

**channel no.**

Specifies the number of the channel for the particular line. The channel number default is 2 for Models 3 through 6. It is D (hexadecimal 13) for Model 8.

**mem-type**

Specifies the main storage type being monitored. Note that this subparameter may be used with or without the item MEMS.

**NO**

Specifies that any histograms specified in a previous parameter set are to be canceled. It actually turns off all histogram reporting.

Once you specify a histogram parameter for a subfile, that parameter remains in effect until HIS=NO is specified. If you want to change histogram parameters for a particular subfile, you must first specify HIS=NO and then specify the desired histograms. The duration of each histogram time line is equal to the recorded time interval unless a specific time interval is specified via the IVL parameter (see 3.3.4). A maximum of 500 lines will be generated for each histogram produced, except for item CYLS, which generates a maximum of 250 lines.

**NOTE:**

*HIS parameter specifications or defaults are not affected by intervening trace event subfiles and remain in effect for all class data subfiles until explicitly reset.*

**3.3.4. Specifying a Particular Time Interval (IVL)**

You can use the IVL parameter to specify a particular time interval for subfile reports. This parameter defines the duration of the histogram time lines and/or the interval for intermediate tabular reports. The format is:

$$IVL = \left\{ \begin{array}{l} [mmm] [, RPT] \\ NO \end{array} \right\}$$

where:

**mmm**

Specifies the time interval in minutes. The time interval must be equal to or a multiple of the recording interval, or it is rounded up to the nearest multiple. If omitted for the first subfile specification, the recording time interval is used.

**mmm,RPT**

Specifies that an intermediate tabular report is to be produced for each class specified in the RPT parameter. These reports are produced at the interval (mmm) specified in the IVL parameter.

**NO**

Cancels previously specified time intervals for histogram time lines and intermediate tabular reports. Once you specify the IVL parameter, it remains in effect for all subsequent subfiles until IVL=NO is specified. If you want to change the IVL parameter, you must first specify IVL=NO and then specify the desired time interval.

**NOTE:**

*IVL parameter specifications or defaults are not affected by intervening trace event subfiles and remain in effect for all class data subfiles until explicitly reset.*

**3.3.5. Specifying Particular Devices (DVC)**

You can use the DVC parameter to specify trace event reports for particular devices or device types. The DVC parameter takes two forms.

1. The following parameter can be used to specify that I/O trace event reports are to be produced for only the particular device type specified. Normally, an I/O trace event report will include all device types. Once a specific device type has been requested, all subsequent reports for I/O trace events will be for the particular device type previously requested. DVC=ALL will reset this specification back to its default of reporting all devices.

$$DVC = \left( \begin{array}{l} \text{DISK} \\ \text{DSKT} \\ \text{TAPE} \\ \text{UNIT} \\ \text{WSTA} \\ \text{ALL} \end{array} \right)$$

where:

**DVC=DISK**

Specifies the device type is a disk.

**DVC=DSKT**

Specifies the device type is a diskette.

**DVC=TAPE**

Specifies the device type is a tape.

**DVC=UNIT**

Specifies the device type is a unit record such as a printer.



DVC=WSTA

Specifies the device type is a workstation.

DVC=ALL

Specifies the devices are disk, diskette, tape, unit record, and workstation.

- The following parameter is used to specify designated disks by device ID for which cylinder reference patterns are to be produced. This parameter is used with disk trace event data. Up to 12 devices can be specified per parameter set on one or more DVC parameters. Normally, a disk trace event report will be produced for accesses to the first 12 disks observed. To report on any other than the first 12 disks, the specific identity of the remaining disks must be given. Also, the normal disk trace event report provides the access pattern by cylinder for each disk. By using the SEEK option, the report will only present those accesses that required seeking or arm positioning.

DVC=[SEEK,]did1[,...did12]

### 3.3.6. Report Qualification by Job (JOB)

You can use the JOB parameter to specify that a trace event report be produced for the duration of a particular job. Also, for traced I/O reports, only those I/O requests issued by the particular job will be reported as compared to DSK, which does all DSK accesses during the job's duration. The format is:

$$\text{JOB} = \left( \begin{array}{l} \text{name[, \#]} \\ \text{, \#} \\ \text{SYMB} \\ \text{SYST} \\ \text{TRAN} \end{array} \right)$$

where:

JOB=name[, \#]

Specifies the particular job by name for which a report is to be produced. You can optionally specify a job number (#) to distinguish between two jobs with the same name that were run during a monitoring session.

JOB=, \#

Specifies the particular job, by job number, for which a report is to be produced. Job numbers can be found in job accounting information or by using the LST parameter (see 3.3.7). To determine the job number, see the note at the end of 3.3.6.

JOB=SYMB

Specifies that I/O event reports are to be produced only for I/O requests issued by symbionts.

**JOB=SYST**

Specifies that I/O event reports are to be produced only for I/O requests issued by system tasks.

**JOB=TRAN**

Specifies that I/O event reports are to be produced only for I/O requests issued by transient routines.

**NOTE:**

*The JOB parameter default is no job qualification. No carryover from one parameter set to the next occurs. If you would like subsequent reports with the same job qualification, you must respecify the JOB parameter in each desired parameter set.*

*In order to use job name qualification, the job must have started after the SAM monitor was activated. If the job was started prior to SAM monitor activation, the job number can be used for qualification.*

*In order to use job number qualification, you must know the binary job number. To determine the binary job number, use the JOB NUMB value on the spool header (or page separator) of the job, or consult the console log for information given by the MI DA command, which lists the job name and number. If the job started or ended during the SAM monitor session, run SAMRPT with the LST=JOB parameter. This will produce a binary job number for each job started or ended during the SAM monitor session.*

**3.3.7. Listing of Job/DVC Information (LST)**

You can obtain a listing of job names, job numbers, and job start/end times from trace event data subfiles by using the LST parameter. This parameter will also provide you with a listing of configured devices.

The format is:

$$\text{LST} = \left\{ \begin{array}{l} \text{JOB} \\ \text{DVC} \end{array} \right\}$$

where:

**LST=JOB**

Produces a listing of jobs started/ended for the previously opened subfile (SFL=). The start/end times listed are the data block start and end times and are presented for time selecting into the data file. An example of this type of listing is shown here:



### 3.4. SAMPLE SAMRPT JOB STREAM

You can produce histograms and intermediate reports as well as final reports or subfile directory listings through a tailored job control stream. The following is a typical SAMRPT job control stream:

```

Section 1 { // JOB SAMRPT
           // DVC 20 // LFD PRNTR
           // DVC 50 // VOL D01234 // LBL MONFLE // LFD SAMIN
           // EXEC SAMRPT
           /$
           SFL=LST
Section 2 { SFL=1, PRODUCTION LOAD - 103=SYSRES
           RPT=I/O, DSK
           HIS=RATE, 103
           HIS=SEEK, 103
Section 3 { SFL=3, PRODUCTION LOAD - 102=DATABASE
           HIS=NO
           HIS=RATE, 102
           HIS=SEEK, 102
           IVL=10
Section 4 { SFL=4, TEST LOAD WITH - 101=DATABASE
           HIS=NO
           HIS=RATE, 101
           HIS=SEEK, 101
           IVL=10, RPT
           TME=1615
Section 5 { SFL=5, XYZ BENCHMARK
           LST=JOB
           DVC=100, 101
           /*
           /&

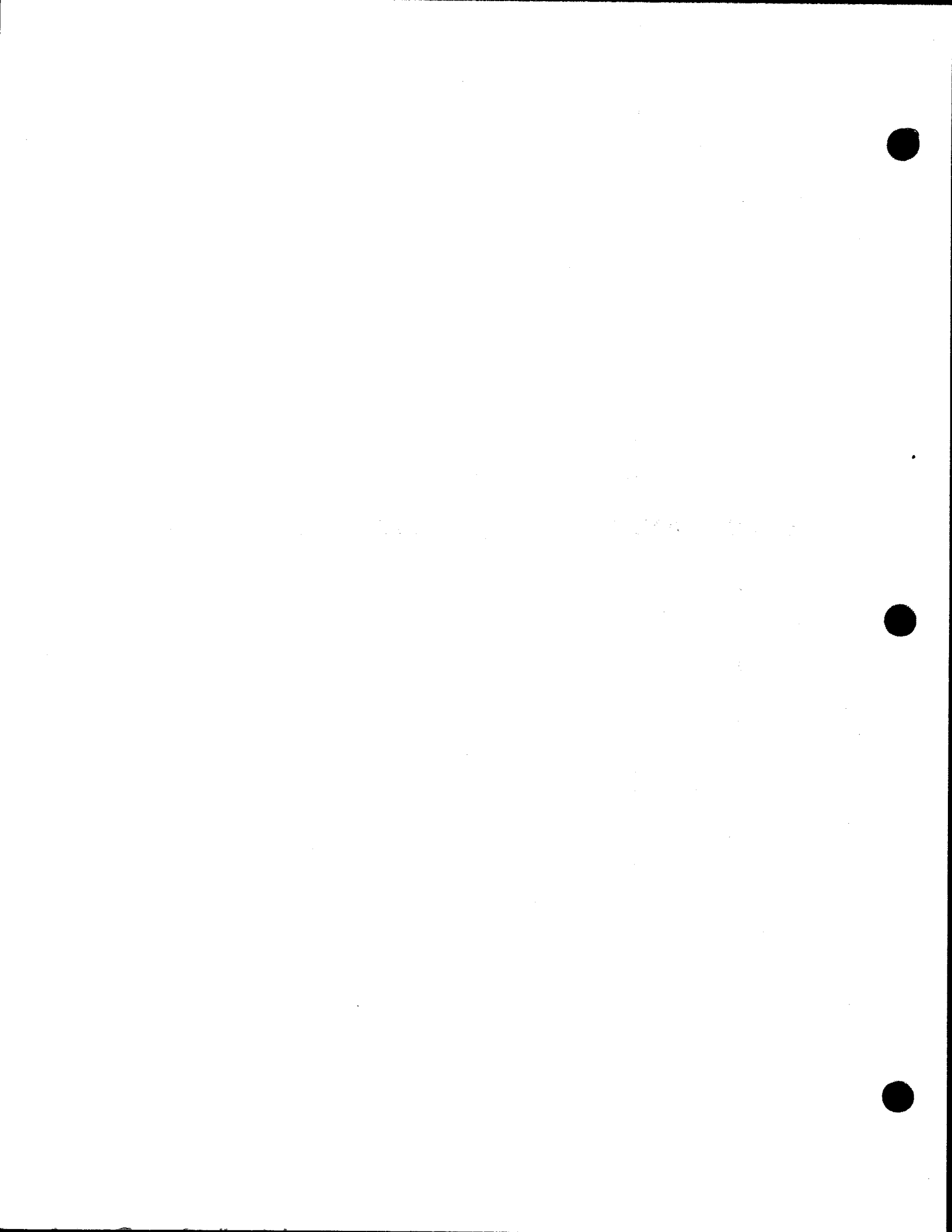
```

1. This section indicates that a subfile directory listing for the file named MONFLE on disk D01234 is to be generated. Notice that both SAMIN and PRNTR must be specified as the input and output logical file names on the LFD job control statement.
2. Line 1 indicates that subfile 1 is processed and the heading PRODUCTION LOAD - 103=SYSRES is printed on the report. Line 2 specifies that a final tabular report is produced for I/O and DSK classes. Lines 3 and 4 specify that histograms are plotted showing RATE and SEEK activity for device 103 at intervals equal to the default recording interval specified by the I parameter when monitoring.
3. Line 1 indicates that subfile 3 is to be processed and the heading PRODUCTION LOAD - 102=DATABASE is printed on the report. Line 2 indicates that the histograms specified in Section 2 are canceled. The last three lines specify that histograms are plotted showing RATE and SEEK activity for device 102 at 10-minute intervals. Since RPT=I/O, DSK was specified for subfile 1, this parameter remains in effect for subfile 3.

4. Line 1 indicates that subfile 4 is processed and the heading TEST LOAD WITH - 101=DATABASE is printed on the report. Line 2 indicates that the histograms specified in Section 3 are canceled. Lines 3 and 4 specify that histograms are plotted showing RATE and SEEK activity for device 101. Line 5 specifies the time interval for the histogram and also specifies that an intermediate class report is to be produced at 10-minute intervals. Line 6 specifies the starting time of a specific time range within subfile 4 to be processed. Since RPT=I/O, DSK was specified for subfile 2 and wasn't changed in subfile 3, this parameter remains in effect for subfile 4.
  
5. Line 1 indicates that subfile 5 is to be processed and the heading XYZ BENCHMARK is to be printed on the report. Line 2 requests a listing of job start and end events that occurred during the monitored period. Line 3 specifies that reports are to be produced for activity only for devices 100 and 101. This assumes that subfile 5 contains trace data for disk cylinder addresses.



**PART 3. ANALYZING SYSTEM PERFORMANCE**





## 4. Using the Measurement Statistics

### 4.1. GENERAL

To analyze your system's performance, study the measurement summary reports. You may need to investigate specific items in more detail to pinpoint the major cause of any performance problems. This detailed study is aided through the SAM intermediate, histogram, and trace event reports.

Measurement data from the system activity monitor is displayed on the system console or printed. Once this data is generated, you can use the measurement statistics to analyze your system's performance and determine what steps to take to remedy any problems that may be reducing your system's efficiency.

It is not unusual to remedy the most apparent performance problem and yet not obtain the performance improvement anticipated. This usually occurs when the initial problem is masking another performance bottleneck that hinders the anticipated performance improvement. You should examine all possibilities before trying to draw any conclusions.

The discussions of the measurements in the following subsections provide you with a logical guide that will help you optimize your system's performance. It must be noted that the examples in these subsections present hypothetical situations that reflect one possible interpretation of any of the class item or trace event measurements. The examples are provided as guidelines to aid you in analyzing a problem and determining its possible remedy.

#### 4.1.1. Collecting the Data

The monitor's class and time parameter selections are dependent on your particular reason for using the product. If a general profile for a typical day's processing is the objective, then choosing C=ALL and I=15 minutes would be suitable. Collecting data over the entire day would enable you to produce a summary report for the day or intermediate reports with granularity to 15 minutes for detecting peak loads. A SAMRPT final tabular report along with histograms at the 15-minute interval provide a convenient starting point for viewing your system profile.

Conversely, if you are tracking a perceived problem or have narrowed the investigation to a particular area, then choosing the 10-second recording rate for specific classes or trace mode would be appropriate.

Suppose for example that the I/O class report of a final tabular report showed that access to a particular disk started to peak around 3 o'clock on Friday afternoons, and this correlated well with reports of delayed response times from your terminal operators. Further investigation using the DSK class revealed that the seek rate and the average number of cylinders moved for this disk increased significantly. You conclude correctly that the staff's push to finish up by the weekend is putting a load on that particular disk and the additional seek time is causing the degraded response times. The next step to take in using the system activity monitor would be to trace that disk's activity with the T=DSK trace feature. Most commonly, you would see a ping-pong effect between the heaviest used files. The solution would be to reallocate the files experiencing contention. Ideally, these files should be on separate disks or at least allocated back-to-back on the same disk.

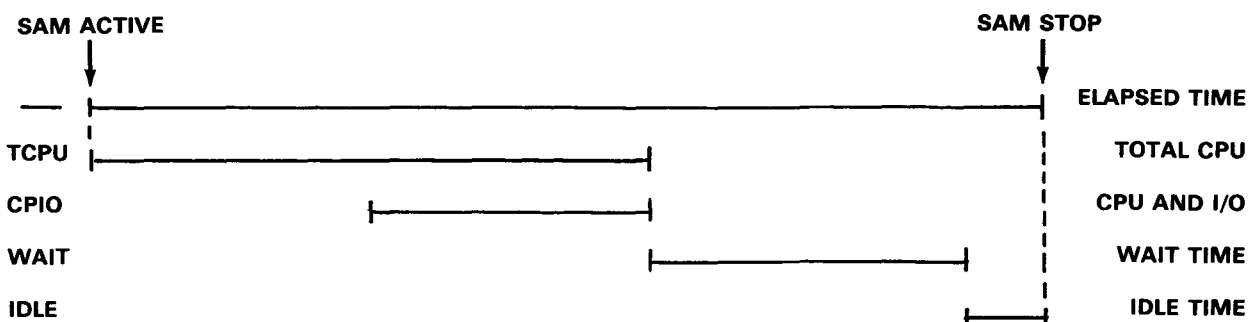
## 4.2. CLASS ITEM MEASUREMENTS

The classes discussed in the following subsections are CPU, I/O, DSK, MEM, SVC, and COM. The item readings for each class are described and examples are given to help you analyze your statistics.

### 4.2.1. CPU Class Items

The item readings under the CPU class measure the activity of your CPU. They are IDLE, WAIT, TCPU, CPIO, KEYS (specified by KEYn and KEmm), SPVR, SVCR, I/OR, TMER, OTHR, and INTR. You can use these measurement values to see whether your system is operating efficiently. If you find from these readings that your production work load is not making efficient use of CPU time, you can use the values generated by SAM to help you determine how to remedy the problem. By monitoring job mixes for a specific interval and taking CPU class readings for that interval, you may determine which jobs are CPU bound versus which jobs are I/O bound. For example, after balancing CPU-bound jobs with I/O-bound jobs, you may find, in some cases, that it is possible to obtain enough additional run time to let you schedule more jobs in a particular work-load interval. A more detailed definition of each CPU class item value is presented in the following subsections.

The CPU class items provide a statistical breakdown of the elapsed times of the monitor session. The following diagram shows the basic relationship of the primary items.



Understanding the relationships of these items and what affects them is the basis of systems analysis. Bear in mind that rarely, if ever, are there absolute numbers or values (for an item) that denote acceptance or failure by themselves. The individual item, the relationship between items, and the total profile must be examined in assessing your system's performance. The meaning of the data generated by the system activity monitor depends on your knowledge of your particular system, the types and number of jobs and tasks running in your system, and the environment in which they are run.

#### 4.2.1.1. IDLE

The IDLE value tells you the percent of time (for the specified interval) that tasks in the system and all peripherals were inactive. You can use the IDLE and WAIT items to gauge the amount of processor time available for additional work. In a communications or interactive environment, the IDLE value could reflect transmission and operator *think* time.

#### 4.2.1.2. WAIT

The WAIT value tells you the percent of time that tasks in the system were inactive but some peripheral device was busy. WAIT time usually occurs when the CPU is inactive because the jobs in the system were waiting for the completion of an I/O operation.

You can use the WAIT value, like the IDLE value, to tell you just how effective your job mix is. For example, a high WAIT value indicates an ineffective use of CPU time. In some cases, mixing your I/O dependent jobs with other jobs that require less I/O can decrease the WAIT time and thus increase throughput. You can examine statistics from the I/O trace reports by job to determine which jobs in the mix require substantial I/O processing.

For other cases, the remedy for excessive WAIT time is to examine device contention, specifically, the disks. The DSK class statistics and cylinder trace feature are useful for optimizing file placement, both per disk and among all disks.

#### 4.2.1.3. TCPU

The TCPU value tells you the percent of time the central processor was busy on behalf of the supervisor, system tasks, or user tasks. In conjunction with the IDLE and WAIT values, this measurement reflects the balance of your environment. A high TCPU value usually indicates a CPU-bound mix and a low TCPU value usually indicates an I/O-bound mix.

#### 4.2.1.4. CPIO

This item is commonly referred to as CPU and I/O overlap. It is a measure of a concurrency between the CPU actively executing instructions and at least one busy peripheral device. An objective for optimum system performance is to have as much overlap of CPU and I/O as possible. As a rule of thumb, the sum of CPIO and WAIT will approximate a measure of I/O activity as a percent of elapsed time.

#### 4.2.1.5. KEYS (KEYn, KEmm)

Specified by KEYn and KEmm (where n represents one hexadecimal digit and mm represent two decimal digits), the KEYS values tell you the percentage of CPU time that tasks were executing. The item KEYS shows the percentage of CPU time for all keys, and its parameters, KEYn and KEmm, show the percentage of CPU time for specific keys. Note that the valid key number range is 0-48 for Model 8 and 0-15 for Models 3 through 6. Key 0 represents symbionts, such as the run processor, output writers, and interactive services; OS/3 system tasks, such as transients and the spoolers; and some data management shared code.

Keys 1 to n represent user jobs. It is possible to correlate the key n statistics to particular jobs by retaining the console log from the monitor session. For System 80, Model 3 through 6, the job slot number is the same as its key number.

For all models, the key value for a job is available also from the MI DA command. It is possible to correlate the key n statistics to particular jobs by retaining the accounting information of those jobs. The 2-digit value after JOB # on the AC01 message is the key (or slot) number for that job.

For example, this measurement can help you tailor your batch work load through the creation of a more efficient job mix. To understand how you can accomplish this, consider the following example.

Suppose that each week on Monday and Tuesday you are running the production work loads shown in Figure 4-1.

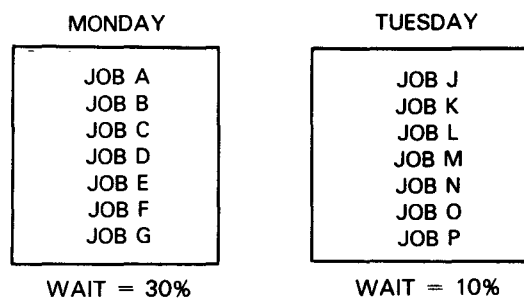


Figure 4-1. Sample Daily Production Work Loads

As discussed earlier, you can eliminate CPU wait time by mixing I/O dependent jobs with other jobs that require less I/O. Suppose, in our example, that the KEY value for JOB M in Tuesday's work load showed a relatively high percentage of CPU time. On the other hand, the KEY value for JOB G in Monday's work load shows a relatively low percentage of CPU time. By switching the CPU dependent JOB M with I/O dependent JOB G, you can enhance Monday's job mix and thus reduce WAIT time. Monday's production work load will then run faster, increasing the IDLE time that could perhaps be used to run more jobs.

#### 4.2.1.6. SPVR

The supervisor state item (SPVR) value tells you the percentage of CPU time that was used for interrupt processing and task dispatching functions. A high value for this item could reflect a significant amount of interrupt processing that may be slowing down job processing. You can use the individual interrupt rate items (4.2.1.7) to determine which specific interrupts are contributing to this value.

#### 4.2.1.7. Interrupt Rates

The interrupt rates tell you the number of times a specific type of interrupt occurred during the specified interval. The interrupt rates are:

- SVCR

Tells you the number of times per second that supervisor call interrupts occurred during the specified interval.

- I/OR

Tells you the number of times per second that I/O interrupts occurred during the specified interval.

- TMER

Tells you the number of times per second that timer interrupts occurred during the specified interval.

- OTHR

Tells you the number of times per second that interrupts other than those specifically identified occurred during the specified interval.

- INTR

Tells you the number of times per second that all interrupts occurred during the specified interval.

You can use all these values to determine what kind of interrupt activity is occurring over the specified monitoring interval. A high I/O interrupt rate (I/OR value), for example, could mean that the blocking factor for records in your job is too small. Since an I/O interrupt occurs each time a block of records is read, increasing the number of records per block can decrease the number of I/O interrupts.

### 4.2.2. I/O Class Readings

The item readings under the I/O class measure the I/O activity in your system. They are RATE, QUED, and ERRC. The items under the I/O class can be referenced by either the device ID (did), with a maximum of 116 devices, or by the channel number (CHn).

Channel statistics provided by the I/O class do not include I/O activity initiated by ICAM. This means that no activity is reported for CH2-SLCA in an environment with ICAM actively handling the physical I/O. (See 4.2.6.)

For system configurations with cochanneling, the I/O class reports accesses via the primary and secondary channel and device ID. Only the I/O class produces statistics for both primary and secondary channel/device addresses.

#### 4.2.2.1. RATE

The RATE value tells you the number of times per second that I/O commands were issued for a specific channel or I/O device during the specified interval. You can use this value to examine I/O processing over the channels (channel 2 for Models 3 through 6, and channel D or F for Model 8) and available devices.

For example, suppose you were running a number of jobs that required a substantial amount of I/O and you found, by using the system activity monitor, that a high percentage of WAIT time occurred when running these jobs. One reason for the high percentage of WAIT time might be that a large number of jobs in your system were all requesting the use of the same I/O device. In this case, you could use the I/O RATE measurement to determine which particular device is being requested by your jobs. By mixing the jobs that require the same device with other jobs, you can decrease WAIT time and thus process your jobs faster. Another possibility would be to use an alternate device for some of the concurrent jobs.

#### 4.2.2.2. QUED

The QUED value tells you the percent of I/O commands (for the specified interval) that were delayed because a specific I/O device was busy. A high percentage of I/Os queued for a particular device may mean that a large number of jobs in the system require that device. In such a case, a high percentage of WAIT time will result, since each job remains inactive while waiting for its predecessor to finish using the particular device. By running different job mixes and taking QUED readings for each run, you can determine which job mix is the most efficient. The most efficient job mix will produce the lowest QUED value. You may also want to consider obtaining more devices to decrease the demand for a particular device.

#### 4.2.2.3. ERRC

The ERRC value tells you the number of times that error recovery was required. If this value is consistently observed for a particular device, some marginal hardware error may be adversely affecting your system's performance.

### 4.2.3. DSK Class Items

The item readings under the disk class (DSK) measure the activity of specific disk devices. They are DISK, SEEK, and CYLS.

The items under the disk class can be referenced by the device ID (did) with a maximum of 48 devices. You can use the measurement values to analyze the performance of individual disk devices supported by your system.

The seek statistic is associated with a particular device, independent of the access pass. Thus, only the primary device address reports disk class statistics, even in a channeled environment.

#### 4.2.3.1. DISK

The DISK value tells you the number of times per second that I/O requests were issued to a particular disk device during the specified interval. You can use this value to determine whether disk access for all disks is balanced.

If a high I/O rate occurs for a specific disk device, either many jobs are accessing the same file or they are accessing different files that all happen to be on the same disk. The latter case can be distinguished from the former by looking at the SEEK data (see 4.2.3.2). If most of the accesses are for the same file, rescheduling of some of these jobs may be in order. If many different files are being accessed on the same device, relocating some of these files to another disk device with a low access rate should improve performance.

#### 4.2.3.2. SEEK

The SEEK value tells you the percent of I/O requests that required arm positioning; you should attempt to minimize this value for optimum disk performance. A high percentage for this item indicates file accessing contention. For example, suppose JOB1 and JOB2 are running concurrently and using FILEA and FILEB, respectively. FILEA is located at cylinder address 10-50 and FILEB is located at cylinder address 300-360. JOB1 and JOB2 accesses create a ping-pong effect when repositioning the disk read/write heads from FILEA to FILEB. Better performance could be obtained by reallocating FILEA and FILEB as close together as possible on the disk or allocating FILEA on one disk and FILEB on another disk. Eliminating the high SEEK rates can significantly improve disk I/O time since mechanical movements of the arm are very time consuming when compared with the electronic transfer of data to or from the disk.

#### 4.2.3.3. CYLS

The CYLS value tells you the average number of cylinders moved per seek operation. This item may point out possible file allocation problems on a given disk volume. In some situations, seeking for a given disk device may not be high, but when the disk arm does move, it may have to travel a great distance. It may just be that two heavily accessed files on that volume are quite a distance apart. Either allocating these files closer together or placing them on different volumes should reduce the seek time under these circumstances.

#### 4.2.3.4. Using DSK Class with DSK Trace

If the DSK class values show cause for concern, you can obtain more specific data by using the trace mode feature of the system activity monitor. The DSK or I/O trace can be used to correlate accesses to the requesting job. More specifically, the DSK trace produces the histogram plot reports showing frequency of accesses per cylinder address. This cylinder access report, when used with a volume table of contents of the disk (VTOC), enables you to evaluate and realign file allocations on your disks (see 4.3.1 and 4.3.2).

#### 4.2.4. MEM Class Items

The item readings under the MEM class measure the activity of the system's main storage facilities. They are:

- |        |        |        |        |
|--------|--------|--------|--------|
| ■ NOAJ | ■ TRRQ | ■ TRRU | ■ LGOF |
| ■ NOJS | ■ TRRL | ■ MEMS | ■ LGON |
| ■ TRRC | ■ TRUC | ■ HITR |        |

##### 4.2.4.1. NOAJ

The NOAJ value tells you the number of jobs that were active in the system at the expiration of the specified interval.

##### 4.2.4.2. NOJS

The NOJS value tells you the number of jobs that started during the specified interval.



#### 4.2.4.3. TRRC

The transient requests call (TRRC) value tells you the number of times per second that transients were requested during the interval. You can use this value to determine whether the jobs in your system require a high number of transient routines. In a job mix containing jobs that require a substantial number of transients, many of the transient calls may be queued because of insufficient transient areas in main storage. If the rate of transient calls is high, you may want to mix the jobs requesting transients with other jobs not requiring transient routines in order to process your jobs faster, or you may want to regenerate your system to make more transient areas available.

#### 4.2.4.4. TRRQ

The transient requests queued (TRRQ) value tells you the percent of transient calls that could not be serviced immediately because a transient area was not available. A high percentage of transient queueing decreases processing efficiency because jobs requiring transients must wait for main storage transient areas to become available. Therefore, if a high percentage of transient calls are being queued, you may want to assign more transient areas in your system. As a result, more transients can be serviced, and your jobs can be processed faster.

#### 4.2.4.5. TRRL

The transient requests loaded (TRRL) value tells you the percent of transient calls that were loaded during the specified interval.

#### 4.2.4.6. TRUC

The transient reuse calls (TRUC) value tells you the percent of transient calls (for the specified interval) that were for reusable transient routines. This can be used with the percent transients reused value to determine the number of times reusable transients were loaded.

#### 4.2.4.7. TRRU

The transients reused (TRRU) value tells you the percent of transients that were actually reused during the specified interval. Any difference between the values for TRUC and TRRU indicates the percent of times that reusable transients had to be reloaded. You may want to increase the number of transient areas to reduce the amount of reusable transient loading.

#### 4.2.4.8. MEMS

The main storage (MEMS) items show you the main storage mapping characteristics of your system. They are:

- BUFP
- SYMB
- FREE
- CBUF
- SHCD
- JOBS
- MAXF

Their values are based on the system map at the end of each monitoring interval. You can use these values to determine the main storage resources required for the work load and to determine available main storage for:

- allocating transient areas;
- running more or larger jobs;
- allocating larger user and system buffers; or
- allocating cache buffer sizes.

#### 4.2.4.9. HITR

The input/output hit ratio (HITR) tells you the percent of disk cache hits for the monitoring interval. This ratio is based on a comparison of the number of I/O requests satisfied from the cache main storage buffers to all I/O requests issued. Because only read I/O requests are candidates for the cache feature, the upper bound for any environment is, therefore, the total number of read requests divided by the total number of I/O requests. The cache effectiveness can be gauged further by comparing the resultant value to the number of disk reads from an I/O trace report.

#### 4.2.4.10. LGOF

The LGOF value tells you the number of logoffs for the workstations during the specified interval.

#### 4.2.4.11. LGON

The LGON value tells you the number of logons for the workstations during the specified interval.

#### 4.2.5. SVC Class Items

The item readings under the SVC class measure the occurrences of individual types of supervisor calls. They are:

- SVCR
- SCVP

##### 4.2.5.1. SVCR

The supervisor call rate (SVCR) value tells you the number of supervisor calls by type that were issued during the specified interval. You can use this value to check for abnormally high numbers of specific supervisor calls. For example, suppose a high number was generated for SVC△01, which is a WAIT function type of supervisor call. A high number of SVC△01 calls will occur if the jobs in your system are I/O bound. In this situation, supervisor calls are generated because jobs requesting I/O in the system were waiting for other I/O operations to be completed. By running different job mixes and taking SVCR readings for each run interval, you can determine which job mix is the most efficient. Identification of individual SVCs can be found in a SUPEQU listing. More detailed descriptions of many of these SVCs can be found in the current version of the OS/3 supervisor programmer reference, UP-8241.

##### 4.2.5.2. SCVP

The spooled I/O calls (SCVP) value is the percent of EXCP type SVC calls to a spooled device file. It can be used to correlate the ratio of EXCP calls to real and virtual devices with I/O rates from class or trace data.

#### 4.2.6. COM Class Items

The item readings under the COM class measure the activity of your communication system (ICAM). They are:

- |        |        |        |        |
|--------|--------|--------|--------|
| ■ CIMR | ■ CSEN | ■ CNOT | ■ CPOL |
| ■ COMR | ■ CERR | ■ CINT | ■ CBYT |

The COM class statistics are supported for terminals and devices by using the UNISCOPE terminal protocol. The I/O rates are based on the count of first line buffers (a message may be more than one line buffer) associated with a communications message.

The COM class does not monitor activity in ICAM-IMS workstation environments, since ICAM does not provide the physical support for workstation I/O activity.

#### 4.2.6.1. CIMR

The input message rate (CIMR) value tells you, by line, the number of first line buffer interrupts that occurred per second because of input message traffic during the specified period. You can use this value as a measure of input messages on each line. For example, suppose that your communications system is configured with six terminals on each line of your system; a high value for a line may mean that the communications load is unbalanced. The rate could be lowered by distributing the communication work load evenly across the existing lines or by adding more lines to reduce the terminal load per line. Conversely, with the same six-terminals-per-line configuration, a low value for a line may indicate a lack of message traffic or inactive terminals. Again, it is important to balance the work load if possible. A low value may also indicate that large messages are being transmitted (i.e., 1920 characters on a 9600-baud line take 1.6 seconds of transmit time). Message sizes should be kept as small as possible.

#### 4.2.6.2. COMR

The output message rate (COMR) value tells you, by line, the number of first line buffer interrupts that occurred per second because of output message traffic during the specified period. You can use this value as a measure of output messages on each line. The conclusion drawn from the COMR would be the same as for the CIMR (see 4.2.6.1).

#### 4.2.6.3. CSEN

The sense command rate (CSEN) value tells you the number of sense commands that were issued by ICAM regarding an interrupt marked as a bad status completion. Sense commands are issued in response to error conditions and request status information from the system's communication hardware. Therefore, a high number of sense commands indicates malfunctioning communication hardware.

#### 4.2.6.4. CERR

The error rate (CERR) value tells you the number of error conditions (time-outs, negative acknowledgments, or reply requests) by line that occur during the specified interval. These error conditions indicate transmission problems resulting from hardware malfunctions associated with the respective line.

#### 4.2.6.5. CNOT

The no-traffic rate (CNOT) value tells you the number of no-traffic responses received per line during the specified interval. A no-traffic response is received by ICAM when it polls a device and the device has nothing to send. This value can be used to determine whether the terminals on your system are being used efficiently. For example, a high number of no-traffic responses from a particular line may mean certain terminals on that line are not being used during much of the specified interval. In this case, you may want to reschedule extra work for this time interval to ensure a more efficient use of your system.

#### 4.2.6.6. CINT

The CINT value tells you the number of other interrupts, excluding those identified by the previous COM items, that were serviced from the respective line during the specified interval. This value is basically the rate of line-buffer traffic excluding the first buffer associated with a message. If this item appears quite high in a period, it may indicate the need to increase line buffer sizes to reduce these interrupts. This can be tested by changing the ICAM generation parameters for the respective line. The line buffer size is specified by using the LBL parameter of the line definition as described in the ICAM network definitions and operations user guide, UP-8947 (current version).

#### 4.2.6.7. CPOL

The CPOL value tells you the rate per second of polling interrupts. A poll is sent to each device to: avoid contention, determine operational status, and to determine readiness for sending or receiving data. For an idle period with a 1-second polling rate, this value and the no-traffic (CNOT) value should both equal a rate of 1.

#### 4.2.6.8. CBYT

The CBYT value tells you the number of bytes per second transmitted over the communications line. This value can be used for determining line utilization. That is, the ratio of total bytes transferred divided by the lines baud rate (bits per second). The CBYT value includes the number of message bytes and all control and command bytes presented to or by the host. This excludes all block check characters (BCC) and input end-of-text characters (ETX).

### 4.3. TRACE EVENT MEASUREMENTS

The trace events discussed in the following sections are disk (DSK) and input/output (I/O). The system activity monitor provides input/output counts by device or disk distributions by cylinder address. Trace mode can be used to supplement class data to obtain a more detailed profile of your OS/3 operating system; events can be correlated with job names.

#### 4.3.1. I/O Event Trace (T=I/O)

Four types of information are available through the I/O event trace:

1. a summary table of I/Os by command code;
2. a summary count of all I/Os by device ID or device type;
3. a summary count of the number of bytes transferred by device ID or device type;  
and
4. a summary report coinciding with a particular job's elapsed time interval.

This data is helpful in maximizing your system's activity. Knowing how many read and write operations your system performs over a period of time, for example, may help you adapt your inputting and outputting techniques to optimize your system's performance.

With this type of measurement, you can confirm the I/O requests by command for a specified job to each device utilized by the job. Command codes are usually unique to device classes. The meanings of the command codes can be found in the current version of the hardware and software summary, UP-8868. Appendix A lists some of the more typical command codes for Models 3 through 6.

#### 4.3.2. Disk Event Trace (T=DSK)

The disk event trace lets you analyze your current file accessing patterns, giving you the ability to determine by file which files are used most frequently. SAMRPT produces histograms that show the accesses to your disk pack by cylinder address; when you obtain a VTOC of the current monitor session's disk, you can review the file access pattern. One way to optimize your system's performance would be to make its processing more efficient by placing the most frequently accessed files in the middle of your disk pack and, with the less frequently accessed files added on either side, from the middle outward. Also, the information from the cylinder address accessing report could identify a number of heavily accessed files, one of which might be relocated to a disk that has a lower activity rate.

**PART 4. APPENDIXES**

1944



## Appendix A. I/O Command Codes Subset (Models 3 through 6)

The following is a list of the more typically issued command codes for devices supported by OS/3. This list is not intended to represent every command code supported by the specific devices. Reference should be made to the System 80 (S80) hardware summary, UP-8868 (current version), for a more complete list of all device command codes.

### ■ Printers

04 sense  
X1,X9 print advance  
X5,XD advance print (0770 only)  
X7,XF advance

where: X = advance bits

### ■ Tapes

01 write  
02,12 read  
04 sense  
07 rewind  
0C,1C read backward

### ■ Card readers

02,06 read  
04 sense

### ■ Workstations/S80 console

01 system message write  
09 user write  
02 system message read  
0A user read  
23 enter work area mode  
43 enter system response mode  
32 read event  
07 message waiting  
04 sense

## ■ 8417/19 disks

05 write data  
22 search/read equal  
32 search/read greater than or equal  
0E read ID  
02 read data

## ■ 8420/22 diskettes

23 feed  
31 load track/side/Sector  
21 data set open  
51 data set close  
06 read  
01 write



## USER COMMENT SHEET

We will use your comments to improve subsequent editions.

NOTE: Please do not use this form as an order blank.

---

*(Document Title)*

---

*(Document No.)*

---

*(Revision No.)*

---

*(Update No.)*

### Comments:

**From:**

---

*(Name of User)*

---

*(Business Address)*

Fold on dotted lines, and mail. (No postage stamp is necessary if mailed in the U.S.A.)  
Thank you for your cooperation

CUT

FOLD

NO POSTAGE  
NECESSARY  
IF MAILED  
IN THE  
UNITED STATES

---

**BUSINESS REPLY MAIL**

FIRST CLASS PERMIT NO. 21 BLUE BELL, PA.

---

POSTAGE WILL BE PAID BY ADDRESSEE

**SPERRY CORPORATION**

ATTN.: SOFTWARE SYSTEMS PUBLICATIONS

P.O. BOX 500  
BLUE BELL, PENNSYLVANIA 19424



FOLD



## USER COMMENTS

We will use your comments to improve subsequent editions.

NOTE: Please do not use this form as an order blank.

---

*(Document Title)*

---

*(Document No.)*

---

*(Revision No.)*

---

*(Update Level)*

### Comments:

**From:**

---

*(Name of User)*

---

*(Business Address)*

Fold on dotted lines, and mail. (No postage is necessary if mailed in the U.S.A.)  
Thank you for your cooperation

FOLD



NO POSTAGE  
NECESSARY  
IF MAILED  
IN THE  
UNITED STATES

---

# BUSINESS REPLY MAIL

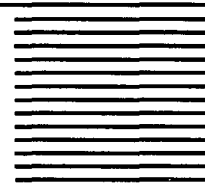
FIRST CLASS PERMIT NO. 21 BLUE BELL, PA.

---

POSTAGE WILL BE PAID BY ADDRESSEE

**SPERRY CORPORATION**

**ATTN: SYSTEM PUBLICATIONS**



P.O. BOX 500  
BLUE BELL, PENNSYLVANIA 19422-9990



FOLD

## USER COMMENTS

We will use your comments to improve subsequent editions.

NOTE: Please do not use this form as an order blank.

---

*(Document Title)*

---

*(Document No.)*

---

*(Revision No.)*

---

*(Update Level)*

### Comments:

**From:**

---

*(Name of User)*

---

*(Business Address)*

CUT

FOLD



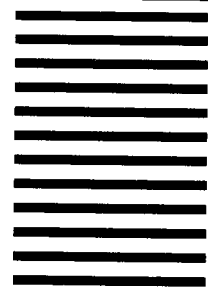
NO POSTAGE  
NECESSARY  
IF MAILED IN THE  
UNITED STATES

**BUSINESS REPLY MAIL**

FIRST CLASS PERMIT NO. 21 BLUE BELL, PA.

POSTAGE WILL BE PAID BY ADDRESSEE

Unisys Corporation  
E/MSG Product Information Development  
PO Box 500 C1-NE6  
Blue Bell, PA 19422-9990



FOLD





