digital

Software Services

AMAR-10 REFERENCE MANUAL VERSION 4.1

AMAR-10
REFERENCE MANUAL
VERSION 4.1

Digital Equipment Corporation

AMAR-10 REFERENCE MANUAL

RELEASE 4.1

TABLE OF CONTENTS

blank page

PREFACE

The AMAR (Automatic Measurement, Analysis, and Reporting) software monitor looks at computer system performance and resource use on a continuous basis and maintains an historical database. It provides periodic reports which are useful for problem detection and analysis, load balancing, and capacity planning.

Currently there are two parts to the AMAR software monitor - System AMAR and Workload AMAR. System AMAR monitors the activity of the computer as a whole and the activity of individual devices such as tape drives and disk packs. Workload AMAR (also called the Workload System) monitors the corresponding activity of individual jobs.

The AMAR-10 Reference Manual provides an overview of how AMAR works on the DECsystem-10. It describes both System AMAR and Workload AMAR. It provides an explanation of sample reports, instructions for obtaining special reports, and a description of normal processing procedures.

A companion volume, AMAR-10 Error Messages, documents all error messages produced by AMAR programs, their likely cause, program action, and recommended user action.

blank page

INTRODUCTION

The AMAR (Automatic Measurement, Analysis, and Reporting) software monitor monitors computer performance and utilization on a continuous basis. It maintains a history of the computer's activity in a set of databases and provides periodic reports on the status of the system.

AMAR is intended to be used to detect bottlenecks and trends in computer usage. It provides the opportunity for you to avoid poor performance by directing efforts and resources to the appropriate problem area. It reduces the lead time required to diagnose problems and provides data which can be used for forecasting and other planning purposes.

AMAR consists of two parts. The first part, which is referred to as System AMAR, collects data continuously on a set of system performance and utilization variables. A database exists which maintains several summary levels of historical data. In addition, there is a menu of reports which can be generated on a daily, weekly, or monthly basis. Reports fall into five categories as follows:

1. System Utilization Summary Reports: Reports characterizing a day's, week's or month's activity. These reports are primarily used for performance and utilization problem analysis and tracking. They provide a graph of CPU utilization and overhead; a summary of problem periods and resources; and a summary of system availability.

2. 'Typical Day' Reports: Reports on system utilization and problem identification for the "average" workday and "average" weekend day of a week or month. These reports aid load balancing by highlighting the typical busy periods.

3. Trend Analysis Reports: Reports which indicate both short term and long term trends in computer utilization. They also project periods when resource consumption may become critical. These reports aid forecasting and capacity planning. They are available both weekly and monthly.

4. Disk Reports: Reports which summarize utilization of the disk subsystem. All disk related information, such as mount time, time in use, average data transfer rate, etc., is presented in a single place.

5. Tape Reports: Reports which summarize utilization of the tape subsystem. All tape related information, such as assigned time, time in use, average data transfer rate, etc., is presented in a single place.

The second part of AMAR is referred to as Workload AMAR. Job specific data, such as job number, PPN, program name, core size, CPU utilization, I/O activity, scheduling class, and so forth, is collected at 5 minute intervals. It is then summarized and, depending on summary level, may be reported in intervals of 5 minutes to a day, week, or month.

The data, also retained in an historical database, is used to locate and solve problems associated with excessive utilization. This data is also expected to provide information for use in a number of areas among which are - forecasting; justifying hardware acquisitions; and providing workload descriptions to be used in benchmarking hardware and software in order to determine the optimal configurations for use by data centers.

Workload characterization reports are generated by a program which allows user definition of report contents. You may specify items around which the report is to be summarized (for example, program name, PPN, charge number) and sort order (for example, largest users of CPU first).

Reports for both parts of AMAR are designed to be as self-explanatory as is technically feasible. Most reports can be obtained automatically. Moderately flexible data inquiry and report generating capabilities are also provided to address special needs.

This manual describes how to use AMAR on the DECsystem-10. It is divided into two main chapters plus several appendices. Chapter 1 describes how System AMAR works, it's standard reports, instructions for obtaining special reports, and normal processing procedures. Chapter 2 provides comparable information on Workload AMAR. The appendices contain detailed information on several topics presented in Chapters 1 and 2. They are intended primarily for reference.

GLOSSARY

CLASS WIDTH - The length of the intervals into which the frequency distribution for a variable is divided and into which the individual sample values are grouped. For example, CPUn Idle Time (CPIn) has a class width default of 5 which means individual sample values are grouped into the following intervals: 0%; 0.01-5.00%; 5.01-10.00%; 10.01-15%; etc. A separate class is always set up for 0 values regardless of the class width specified.

DATA COLLECTION - The process of sampling and recording on disk the selected System AMAR items and workload information.

DATA INPUT - The process of incorporating the raw records from the monitored system into the database.

GRANULARITY - Refers to various summary levels of data; lowest level, or finest granularity, is a raw record, highest level, or coarsest granularity, is a summary record for the longest rolled-up period, which is a usually a fiscal month.

HIGH THRESHOLD - The limit above which an item value is considered to be unacceptable or beyond the range of expected or customary values.

ITEM - A specific variable, which can be measured, related to the utilization or performance of a system. For example, idle time, overhead, swapping rate, etc. are items.

KEY ITEM - Any item appearing in a System AMAR report subsection which has been labeled as "key" in the RFD file.

LOW THRESHOLD - The limit below which an item value is considered to be unacceptable or lower than the range of expected or customary values.

METERED - Refers to the type of measurement of an item in which the exact change, since the last measurement, is known or can be computed. The measurement is accurate and is unaffected by the length of the sample interval or number of samples taken.

MONITORED SYSTEM - A computer system on which an AMAR data collection program collects raw performance data.

RAW DATA - Refers generally to the values stored in the raw records, whether in core or on the disk. Raw data may also refer to the finest level of granularity of data on the system which is being monitored.

RESOURCE - An item, regardless of whether or not it represents a physical "resource".

ROLLUP - The AMAR process of consolidating detail data into the appropriate summary level according to the fiscal calendar.

SAMPLED - Refers generally to the measurement of an item (either metered or snapped).

SAMPLE GROUP INTERVAL - The frequency at which the data collection program writes raw records onto the disk.

SAMPLING INTERVAL - Frequency at which the data collection program takes samples or measures raw data.

SNAPPED - Refers to the type of measurement of an item in which the value of the item is a 'snapshot' of conditions existing at that moment in time and does not reflect other values for that item which may have occurred since the last sample. The accuracy of this form of measurement is dependent on the number of samples taken during a given period of time.

UNDEFINED STATE - If the system crashes or the input or rollup programs fail while the System AMAR database is being updated, errors could be introduced into the database records. Further use of the database may result in erroneous information being obtained. The jobstream will first attempt to recover by using the backup copy on disk. If this fails, restore the database from a tape backup copy and resume processing.

WORKLOAD - The collection of user programs running on the system which is being monitored.

CHAPTER 1

SYSTEM AMAR

1.1 MAJOR FEATURES

Features of System AMAR include:

1. Low overhead, continuous data collection.
2. An historical database with:
   a. Data summarized at the hourly, daily, weekly, monthly, and "typical day" levels.
   b. Data values stored in frequency distributions.
   c. Flexible retention periods for data with monthly values typically kept at least a year.
   d. Automatic deletion of old data.
3. A menu of standard reports which:
   a. Track utilization and performance over the period of a day, week, month or year.
   b. Automatically check for and flag problem items and time periods.
   c. Analyze and report on short term (up to 13 weeks) and long term (up to 12 months) trends.
   d. Consolidate all disk and tape information onto separate reports.
4. Special reporting programs which:
   a. Are user runnable at a terminal or via a batch stream.
   b. Provide for ad-hoc report generation.

c. Enable you to directly access any piece of information contained in the database.

d. Provide data in human readable form or, alternately, in a form useable by other programs.

5. A single daily batch stream which will:

a. Automatically produce daily, weekly, and monthly reports according to a fiscal calendar.

b. Maintain the System AMAR database.

c. Prevent buildup of data files on disk.

1.2 OVERVIEW OF SYSTEM AMAR

The three functions of System AMAR are data collection, database management, and reporting. These functions are performed by seven programs which are described briefly below. Refer to Figure 1-1 for an overview of program and data flow. The bold portions of Figure 1-1 refer to functions that are normally performed automatically.

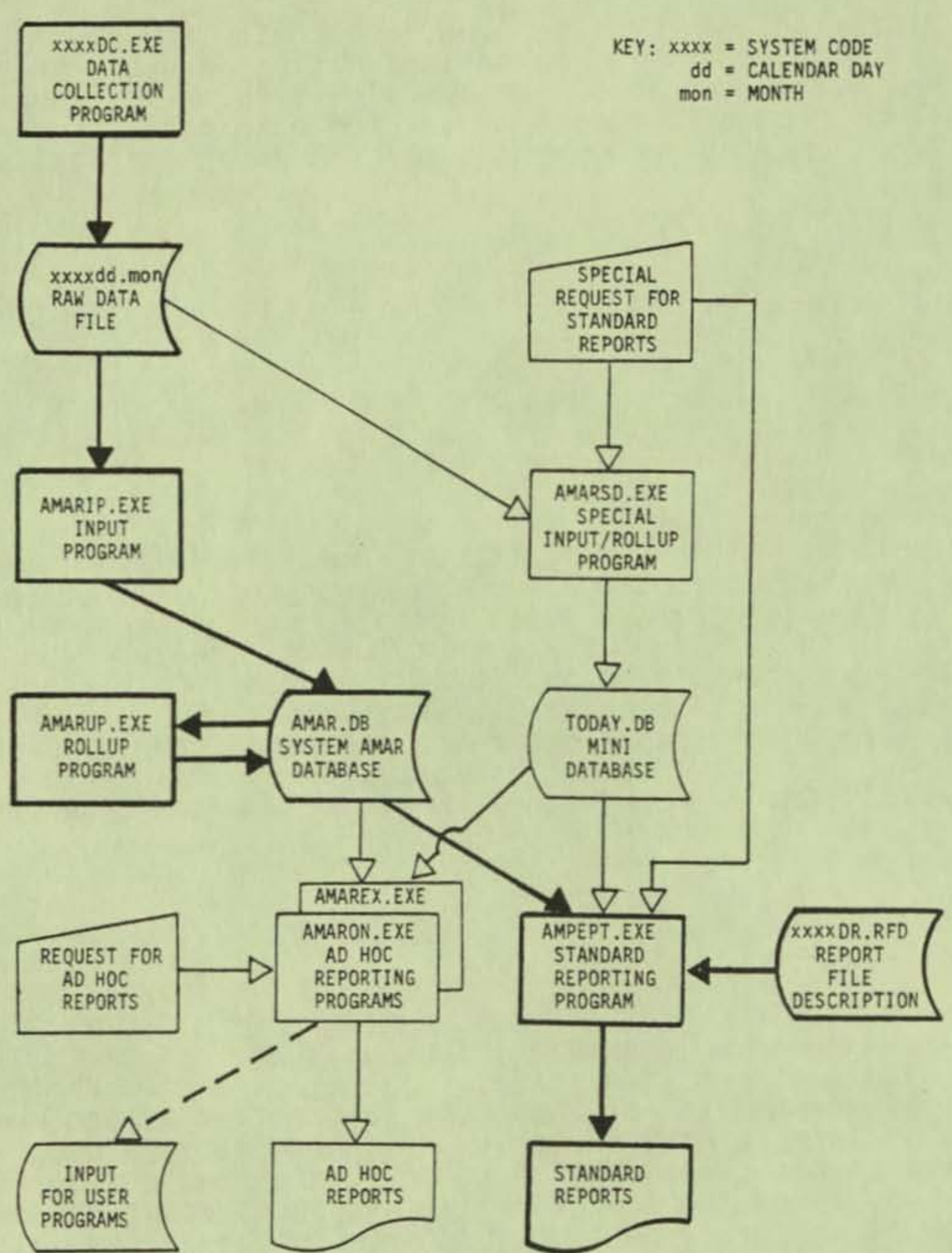
1.2.1 Data Collection

The data collection program, named xxxxDC where xxxx is a 4-character system code, collects data related to the general utilization and performance of a CPU and its associated subsystems - memory, disk and tape.

xxxxDC interfaces with the monitor, and does the actual sampling and measuring of the various System AMAR items which may be specified by the user. The sample data are collected in core. At the end of a sample group interval (1 hour) raw records are created from the sample data and are written to disk. Raw records contain a frequency distribution of the sample data values for a given item. Each record represents a separate predefined range of values, contains the average for the range, and the number of samples whose values fall into that range. xxxxDC runs 24 hours a day, preferably as an OPSER subjob.

1.2.2 Data Input

The data input program, AMARIP, performs the first half of the database management function. AMARIP inserts the raw records into the database. It keeps track of when data was last input and uses a grace period to determine how many days worth of data should be input into the database at one time. AMARIP requests a separate raw data file for each day since the last raw file was input up to the current day or last day of the grace period, whichever comes first. The grace period is equal to the number of days that hourly data is kept in the database. It prevents unnecessary processing in the case where the operator may have incorrectly set the machine date too far in advance.



SYSTEM AMAR OVERVIEW
FIGURE 1-1

1.2.3 Rollup

The rollup program, AMARUP, performs the second half of the database management function. It rolls hourly data up into daily, weekly, monthly, and "typical day" records. AMARUP also deletes any data records which have expired their retention periods.

1.2.4 Report Generation

Automatic report generation is performed by AMREPT. AMREPT produces a menu of standard reports, described in the Section called "Annotated Sample Reports". These reports provide information on system utilization and performance, problem resources and periods, system availability, and usage trends. You may elect to generate only a subset of these reports or the entire menu. Reports may be produced on a daily, weekly or monthly basis according to a fiscal calendar. Weekly and monthly reports are triggered by the passing of the end of the fiscal period to be reported upon. Reports are automatically generated only once per fiscal period by AMREPT. AMREPT, however, may be run interactively to recreate reports for past time periods.

In addition to automatic report generation, System AMAR provides you with the capability to generate special reports. There are three programs in this category, AMARSD, AMARON, and AMAREX.

AMARSD is a variation of the input program AMARIP. AMARSD makes a separate mini database from the raw daily file, by-passing the System AMAR database. The summary file is then forwarded to AMREPT for output in a Daily System Utilization Report or Disk or Tape Report. AMARSD enables you to generate a report from any raw file including the one currently being created, thus obtaining up-to-the-hour information on system utilization.

AMARON is an online inquiry program which can be used to report on any individual item contained within the System AMAR database or the mini-database output by AMARSD. AMARON may output the data as either tables or histograms. Reports may be sent to the terminal or stored on disk for later printing or further program use. Headings may also be stripped from the reports to enable easy access by statistical packages or by your own program.

AMAREX is an online inquiry program which can be used to extract records from the database. The records can be displayed at the terminal or sent to a file. AMAREX records are primarily intended for further processing by user programs.

1.2.5 Operation

The data collection program (xxxxDC) should be run continuously. This program should be set up as an OPSER subjob startable by the OPR.ATO file. Except for data collection, System AMAR operation is accomplished through the use of a single job stream, called AMAR.CTL, which will usually be run after midnight. The stream will perform housekeeping on old files, run the input, rollup, and automatic report program, and produce a tape backup of the database if desired. A sample jobstream is provided in the Appendix called "System AMAR Batch Stream - AMAR.CTL". This stream is usually modified slightly to conform to a site's individual requirements.

1.3 ANNOTATED SAMPLE REPORTS

System AMAR is currently designed to generate the following five standard types of reports:

1. System Utilization Reports - Can be generated daily, weekly, or monthly. They provide a picture of the past day, week, or month's activity, primarily to aid in problem identification and solution.

2. Trend Analysis Reports - Can be generated weekly or monthly. They highlight both short and long term trends in utilization and performance to aid in forecasting and capacity analysis.

3. 'Typical Day' Reports - Can be generated weekly or monthly. They aid the load balancing process by providing an indication of "typical" busy/non-busy periods.

4. Disk Reports - Can be generated daily, weekly, or monthly. They combine all disk related information, such as pack utilization, I/O rate, and free space, in one place.

5. Tape Reports - Can be generated daily, weekly, or monthly. They combine all tape related information, such as drive utilization and I/O rate, in one place.

All of the above reports can be produced automatically by the AMREPT program which is run as part of the nightly batch stream, AMAR.CTL. Alternately, the same reports may be produced on demand by running AMREPT at the terminal. Two programs, AMARON and AMAREX are also provided which enable you to generate special reports. AMARON allows you to obtain averages or histogram data for any item contained in the AMAR database. Results can be printed at the terminal or written to a file for later processing by your own program or by statistical packages. AMAREX allows you to extract any record from the database. The records are written to a sequential file for later processing by your own program.

The following sections describe the System AMAR reports and show annotated examples of each.

1.3.1 System Utilization Reports

Utilization Summary Reports can be generated for a day, week, or month. They are designed to be used together. The report formats are similar. Problems showing up on a monthly report can be easily traced back to specific hours within a day. Utilization Summary Reports show activity on the system as a whole. They should be used in conjunction with those Workload AMAR reports which show corresponding activity by user jobs. See the Workload AMAR chapter in this manual. Report similarities and differences are described in the following subsections.

1.3.1.1 Daily System Utilization Report -

The Daily System Utilization Report provides the following features:

A graph of CPU utilization and overhead by hour which denotes busy periods.

A prime/non-prime time summary of key utilization items such as idle time, overhead, swapping rate, disk I/O rate, and so forth, for quick scanning by data center management.

Disk free space by pack comparing yesterday with today and giving an indication of trend.

A summary of problem periods.

A summary of problem resources (also called "items").

A summary of system availability for prime/non-prime time.

A list of periods of downtime.

Hour-by-hour averages for key items as well as problem items. This feature allows most problem periods and problem resources to be pinpointed and analyzed without the need to generate any other special reports. In addition, it is designed to allow easy comparison with the workload reports which produce hour-by-hour summaries of user jobs.

The Daily System Utilization Report is separated into three sections - a Summary Report, a Problem Report, and a Detail Report. Refer to Figure 1-2. The format of the report header is as follows:

Lines 1 and 2 - Specify the date for which the report is generated, the name of the report, and the page number. The report period begins at approximately 00:01 AM and ends at approximately 23:59 PM.

Line 3 - Specifies the Data Center Name.

Line 4 - Specifies the 4 character system code and the period of time during the day which is considered prime time.

<u>Summary Report (Page 1)</u> - The purpose of this report section is to give management a quick overview of system utilization and peak processing periods during the previous day.

The graph at the top of page 1 (A) represents processor 0. It shows CPU utilization (*) and overhead (#) for each hour of the day. The difference between the CPU utilization (B) and overhead (C) lines represents CPU time consumed by user jobs, lost time, and some embedded priority interrupt (PI) time.

Below the graph are prime and non-prime averages for ten "key" items (D). These averages give a quick picture of the usage during the day. You have the option of selecting any number of key items from the entire list of items that AMAR collects. A default set of key items is supplied with AMAR. You can change this set by editing the xxxxDR.RFD File. See the Section called "How to Tailor the Report Contents". The items and their definitions are listed in the Appendix called "System AMAR Item Definitions". The last column in the row of key items gives the total number of hours (E) each key item exceeded or equaled the thresholds. Hours where the averages exceeded the thresholds are denoted by asterisks (*) on pages 3 and 4 of the sample report.

The last section on page 1 contains disk free space by pack. PERCENT FREE SPACE LEFT ON DISK PACKS is an average of samples taken throughout the day rather than just the reading at the end of the day. The current day's average (F), the previous day's average (G) and the difference (H) between the two are given. If the pack has filled up significantly since yesterday (large negative difference), some housekeeping may be in order.

<u>Problem Report (Page 2)</u> - The purpose of this report section is to give management an overview of potential problem periods and problem items. Problem periods (I) are indicated graphically in the lefthand section of this report. For each hour, the number of items whose averages exceeded the watchdog limits is indicated by a row of asterisks (for key items) and plus signs (for other than key items) (J). Periods of the day which are followed by several asterisks or plus signs should be investigated more thoroughly - generally by first looking at the Detail Report (page 3).

Problem resources (K) are listed in the righthand section of the report. The number of hours the resource values exceeded watchdog limits is indicated graphically by a row of P's (for prime-time hours) and N's (for non-prime-time hours) (L). A brief comment suggests a possible cause or an approach to solving the problem (M). The comments are intended only as a pointer to a possible problem area or to some further course of action. There may be more than one possible cause for a given problem (for example, such as high overhead) and there is often more than one solution which can be applied. The alternatives must always be evaluated carefully. The comments, in and of themselves, are not intended to recommend a solution to a given problem. Comments are defined in the xxxxDR.RFD file. You may modify or delete them.

The SYSTEM AVAILABILITY SUMMARY (N) at the bottom of the page expresses system uptime (P) for prime and non-prime time as a percentage of wall clock time. AMAR measures uptime from time of system reload to within one minute before a system crash.

The percent of time measured by AMAR (Q) may be somewhat less than system uptime, because AMAR measurements (other than uptime) taken in the partial hour before a system crash are not recorded on disk.

A reload (R) will be counted if the system stays up long enough for AMAR to write its reload record.

Periods of downtime are listed (S). The beginning of a period of downtime should be accurate to within a minute. The end of the period of downtime coincides with a system reload.

Detail Report (Pages 3-4) - The main purpose of this report section is to give the user investigating a problem period or problem item an hour by hour report of item utilization. Problems are flagged with asterisks to make them stand out. This report also provides summaries of item use for prime and non-prime time, with indications of how often the problem items exceed watchdog limits. The detail report consists of a table (T) with a row for each hour of the day and a column for each of ten key items. Under each item name is listed the average value for each hour (U). To see corresponding usage by individual jobs, refer to the Workload AMAR "Hourly Report Showing Major CPU Users". Note that usually averages and percentages are expressed as whole numbers - except in the case of items whose values are usually less than 1 such as for ACT SWAP RATIO.

Any element of this table which exceeds the watchdog limits is flagged with an asterisk (V). A row with several asterisks is likely to represent a problem period. A column with several asterisks usually indicates an overused item. If less than 45% (27 minutes) of the hour was measured, the data values are shown as "---", usually indicating system downtime (or, in the case of disk packs, that the pack was not mounted).

After the row-per-hour table, there are summaries for prime and non-prime time which give average value (W), watchdog limit (X), percent of time over limit (Y), and number of hours (Z) when the average was over or equal to the watchdog limit.

A maximum of ten key items are contained on the first page of this report. Key items are meant to be always printed. Items other than key items may also be always printed or be printed only if at least one hourly average exceeds or equals the threshold that has been defined for the item or the threshold is exceeded or equaled 10% of the time. If items other than key items are over limits, or more than ten key items are selected for daily reporting, these items will be included in additional pages of detail report.

For purposes of the Daily Report, watchdog limits may be set differently for each item, including individual disk packs. For example, this will allow different treatment for a pack which is 95% full, but stable and a pack which should average 30% free space to accommodate peak usage.

To indicate when it may be useful to check the other pages of the detail report, the column labeled ALL PAGES (AA) indicates the total number of problem items during the period -- if it is different from the next to last column labeled THIS PAGE (BB), the total difference is the number of items on other pages which were over limits during the period.

DATE: 15-FEB-82 (MONDAY)

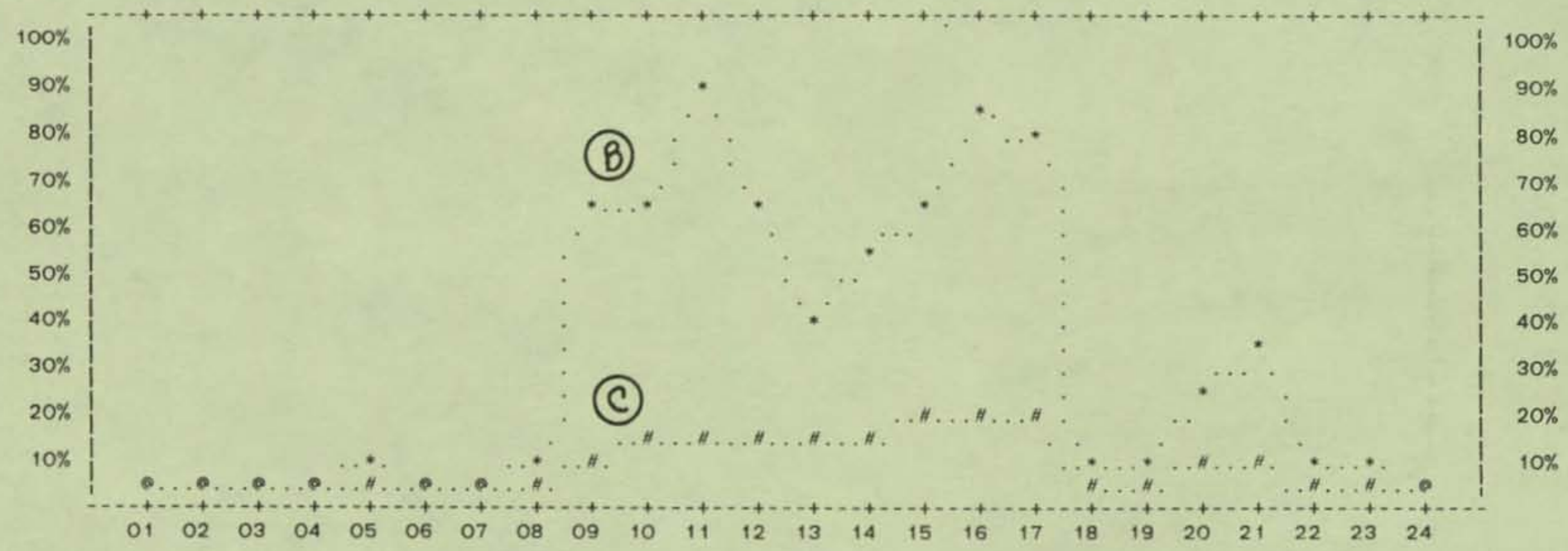
- AMAR -
DAILY SYSTEM UTILIZATION SUMMARY REPORT

PAGE: 1

PUT ANY TITLE HERE
SYSTEM: PATH   PRIME TIME: 0800 - 1700

A

--------- CPU UTILIZATION (*) OVERHEAD (#) AND BOTH (@) ---------



------------------------ HOUR ----------------------------

D

------------------ SUMMARY OF KEY UTILIZATION ITEMS -----------------

-------AVERAGE-----	# JOBS LOGGED IN	# LINES IN USE	% IDLE TIME	% LOST TIME	% OVHD TIME	ACT SWAP RATIO	SWAPPING BLKS/SEC	USER DSK BLKS/SEC	# MTAS ASSIGNED	PGS USER MEMORY	NO. OF HOURS KEY ITEMS OVER LIMITS
----PRIME TIME-----	58	35	34	1	14	.2	35	60	0	1737	1 E
---NON-PRIME TIME--	23	4	91	0	5	.0	0	14	0	1736	

---------- PERCENT FREE SPACE LEFT ON DISK PACKS ----------

-----PACK NAME-----	CASEO	COREO	DEVPO	DSKEO	DSKWO	HRS20	PAWSO	PENSO	PER60	PLT10	STARO	TRNGO	USRSO
---% FREE TODAY----	42	46	16	30	78	44	49	13	23	42	41	74	19 F
-% FREE YESTERDAY--	49	46	3	29	80		49	13			41		26 G
----DIFFERENCE-----	- 7	+ 0	+13	+ 1	- 2		+ 0	+ 0			+ 0		- 7 H

\+ = MORE THAN YESTERDAY   - = LESS THAN YESTERDAY   PACK DISMOUNTED ( )   UNMEASURED (__)   WRONG DAY (?)   BAD FILE (!)

Figure 1-2

DATE: 15-FEB-82 (MONDAY)

- AMAR -
DAILY SYSTEM UTILIZATION PROBLEM REPORT

PAGE: 2

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

I

PROBLEM PERIODS

EACH * = 1 KEY ITEM OVER WATCHDOG LIMIT
EACH + = 1 OTHER ITEM OVER LIMIT

(SEE THE FIRST PAGE OF DETAIL REPORT)
(FOR HOURLY AVERAGES OF KEY ITEMS)

K

PROBLEM RESOURCES

EACH P = 1 PRIME HOUR WHEN THE ITEM WAS OVER THE WATCHDOG LIMIT
EACH N = 1 NON-PRIME HOUR WHEN THE ITEM WAS OVER LIMIT

HOUR	NO. OF ITEMS
00:00 - 01:00	
01:00 - 02:00	
02:00 - 03:00	
03:00 - 04:00	
04:00 - 05:00	
05:00 - 06:00	
06:00 - 07:00	
07:00 - 08:00	
08:00 - 09:00	
09:00 - 10:00	
10:00 - 11:00	*+++
11:00 - 12:00	+
12:00 - 13:00	+
13:00 - 14:00	+++
14:00 - 15:00	+++
15:00 - 16:00	++++
16:00 - 17:00	+
17:00 - 18:00	
18:00 - 19:00	
19:00 - 20:00	
20:00 - 21:00	
21:00 - 22:00	
22:00 - 23:00	
23:00 - 24:00	

J

ITEM	NO. OF HOURS	COMMENTS	
USER DSK BLKS/SEC	P	WARNING	DISK I/O RATE HIGH: CHK PACK I/O
USER UUOS/SEC	PPPP	SERIOUS	USER PGM PROBLEM: CHK WORKLD DATA
% SWAP SPC LEFT	P	WARNING	SWAP SPACE LOW: ALLOCATE MORE
# JOBS TTY IO Q	PPPP	SERIOUS	INTERACTIVE USE HIGH: CHK RESPONSE
%PK DEVPO FREE SPC	PPPPPP	CRITICAL	DELETE UNNECESSARY FILES
PK DEVPO BLKS/SEC	P	WARNING	I/O RATE HIGH: CHK FOR CONTENTION

L M

N

SYSTEM AVAILABILITY SUMMARY

P Q R

	SYSTEM UPTIME	% TIME MEASURED	RELOADS	PERIODS OF DOWNTIME
PRIME TIME :	100.0%	100.0%		17:28 - 17:30
NON-PRIME TIME:	99.8%	96.5%	1	

S

Figure 1-2 (continued)

DATE: 15-FEB-82 (MONDAY)

- AMAR -
DAILY SYSTEM UTILIZATION DETAIL REPORT

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

T

U

PERIOD	# JOBS LOGGED IN	# LINES IN USE	% IDLE TIME	% LOST TIME	% OVHD TIME	ACT SWAP RATIO	SWAPPING BLKS/SEC	USER DSK BLKS/SEC	# MTAS ASSIGNED	PGS USER MEMORY	# ITEMS OVER LIMITS THIS PAGE	# ITEMS OVER LIMITS ALL PAGES
	KEY UTILIZATION ITEMS											
00:00 - 01:00	21	4	96	0	3	.0	0	3	0	1737		
01:00 - 02:00	22	4	96	0	3	.0	0	1	0	1737		
02:00 - 03:00	22	4	96	0	3	.0	0	0	0	1737		
03:00 - 04:00	22	4	96	0	3	.0	0	0	0	1737		
04:00 - 05:00	22	4	94	0	4	.0	0	3	0	1737		
05:00 - 06:00	22	4	96	0	3	.0	0	0	0	1737		
06:00 - 07:00	22	4	96	0	3	.0	0	0	0	1737		
07:00 - 08:00	24	5	90	0	4	.0	2	6	0	1737		
08:00 - 09:00	37	19	39	4	10	.1	48	51	0	1737		
09:00 - 10:00	55	32	38	1	11	.2	17	45	0	1737		
10:00 - 11:00	62	37	10	0	14	.2	18	159 *	0	1737	1	4
11:00 - 12:00	55	32	35	0	11	.1	6	21	0	1737		1
12:00 - 13:00	58	34	61	1	12	.1	7	45	1	1737		1
13:00 - 14:00	61	38	48	1	15	.2	19	35	0	1737		3
14:00 - 15:00	62	39	38	1	18	.2	22	52	0	1737		3
15:00 - 16:00	71	46	19	3	19	.2	136	43	1	1737		4
16:00 - 17:00	59	35	22	1	16	.2	43	87	1	1737		1
17:00 - 18:00	24	6	92	0	5	.0	3	27	0	1735		
18:00 - 19:00	24	5	92	0	5	.0	1	4	0	1735		
19:00 - 20:00	24	4	78	0	7	.0	0	61	0	1735		
20:00 - 21:00	24	4	68	0	10	.0	0	91	1	1735		
21:00 - 22:00	24	6	90	0	5	.0	1	12	1	1735		
22:00 - 23:00	23	5	93	0	4	.0	0	3	0	1735		
23:00 - 24:00	22	4	95	0	4	.1	0	0	0	1735		
----PRIME TIME-----												
AVERAGE VALUE:	58	35	34	1	14	.2	35	60	0	1737		
SHORT TERM LIMIT:	NONE	NONE	<5%	>5%	>20%	>.7	>200	>150	NONE	<768		
% TIME OVER LIMIT:			31.1%					7.5%				
# HOURS OVER LIMIT:								1			1	17
--NON-PRIME TIME---												
AVERAGE VALUE:	23	4	91	0	5	.0	0	14	0	1736		
SHORT TERM LIMIT:	NONE	NONE	<5%	>5%	>20%	>.7	>200	>150	NONE	<768		
% TIME OVER LIMIT:												
# HOURS OVER LIMIT:												

* = OVER LIMITS   > = GREATER THAN OR EQUAL TO   < = LESS THAN OR EQUAL TO

---------- CONTINUED NEXT PAGE --------

Figure 1-2 (continued)

DATE: 15-FEB-82 (MONDAY)

- AMAR -

DAILY SYSTEM UTILIZATION DETAIL REPORT

PAGE: 4

PUT ANY TITLE HERE

SYSTEM: PATH PRIME TIME: 0800 - 1700

-----------------------------OTHER UTILIZATION ITEMS --

PERIOD	USER UUOS/SEC	# JOBS BLK IO Q	%PK DEVPO FREE SPC	PK DEVPO BLKS/SEC	% SWAP SPC LEFT	# JOBS TTY IO Q	# ITEMS OVER LIMITS THIS PAGE	# ITEMS OVER LIMITS ALL PAGES
00:00 - 01:00	7	.0	24	0	89	0		
01:00 - 02:00	9	.0	24	0	89	0		
02:00 - 03:00	8	.0	24	0	89	0		
03:00 - 04:00	8	.0	24	0	89	0		
04:00 - 05:00	15	.1	24	0	89	0		
05:00 - 06:00	8	.0	24	0	89	0		
06:00 - 07:00	10	.0	24	0	89	0		
07:00 - 08:00	49	.2	24	0	85	1		
08:00 - 09:00	144	.9	24	5	67	8		
09:00 - 10:00	141	.4	24	2	49	17		
10:00 - 11:00	201	1.0	10 *	101 *	39	20 *	3	4
11:00 - 12:00	173	.4	3 *	0	43	13	1	1
12:00 - 13:00	205	.3	3 *	11	38	16	1	1
13:00 - 14:00	597 *	.9	3 *	0	32	20 *	3	3
14:00 - 15:00	573 * V	1.0	3 *	0	35	20 *	3	3
15:00 - 16:00	470 *	.9	3 *	0	19 *	27 *	4	4
16:00 - 17:00	358 *	.7	12	20	33	17	1	1
17:00 - 18:00	58	.1	13	0	83	2		
18:00 - 19:00	50	.1	13	0	83	1		
19:00 - 20:00	182	.3	13	0	84	0		
20:00 - 21:00	248	.6	13	0	85	0		
21:00 - 22:00	42	.4	13	0	85	1		
22:00 - 23:00	24	.1	13	0	87	1		
23:00 - 24:00	17	.2	13	0	87	0		
----PRIME TIME-----								
AVERAGE VALUE:	318	.7	9	16	39	17 W		
SHORT TERM LIMIT:	>300	>2.0	<10%	>100	<25	>20 X		
% TIME OVER LIMIT:	32.2%	18.8%	68.9%	2.4%	13.8%	39.1% Y		
# HOURS OVER LIMIT:	4		6	1	1	4 Z	16	17
--NON-PRIME TIME---								
AVERAGE VALUE:	49	.1	19	0	87	0		
SHORT TERM LIMIT:	>300	>2.0	<10%	>100	<25	>20		
% TIME OVER LIMIT:								
# HOURS OVER LIMIT:								

* = OVER LIMITS   > = GREATER THAN OR EQUAL TO   < = LESS THAN OR EQUAL TO

Figure 1-2 (continued)

1.3.1.2 Weekly Utilization Report -

The Weekly Utilization Report provides the following features:

A graph which shows the average CPU utilization and overhead for each 2 hour period for each day in the week.

A prime/non-prime time summary for the week and each day in the week of utilization and performance items such as idle time, overhead, swapping rate, disk I/O rate, system uptime, etc.

A prime/non-prime time summary of key items and problem items during the week.

A summary of problem days in the week.

The Weekly Utilization Report is separated into three sections - a Summary Report, a Detail Report, and a Problem Report. Refer to Figure 1-3. The format of the report header is the same as for the Daily System Utilization Report.

<u>Summary Report (Page 1)</u> - This report section gives management a quick overview of system utilization and peak processing periods during the previous week. It is often used in conjunction with the Workload AMAR "Program Name Report", "Batch Vs. Timesharing Report", or the "PPN Report".

The graph at the top of page 1 (A) represents processor 0. It is a bar chart which shows CPU utilization (*) and overhead (#) for each 2 hour period in the day starting at midnight. The difference between the top of the CPU utilization (B) and overhead (C) bars represents CPU time consumed by user jobs, lost time, and some embedded priority interrupt (PI) time.

Below the graph are prime and non-prime time averages for eleven key items (D). These averages are intended to give a quick picture of the usage during the week. You have the option of selecting key items from the entire list of items that AMAR collects. The items and their definitions are listed in the Appendix called "System AMAR Item Definitions". In addition to key items, the percent of system uptime (E), the percent of time that AMAR measured the system (F), and the total number of system reloads (G) during the week is presented.

<u>Detail Report (Pages 2-3)</u> - This report section lets you compare the total resource usage for the week with the resource usage of each of the days in the week. This report section, in conjunction with the Problem Report, allows you to track back through the week to determine on which days the most resource consumption and/or problems occurred. You can then refer to the Daily Utilization Summary Report for more detail. Averages for key utilization items (E) are presented first, followed by averages for other utilization items. Other utilization items are those whose averages have exceeded the long term limit as listed on the Problem Report and as defined in the xxxxDR.RFD Report File Description. Each average exceeding the limit is flagged by an asterisk (*) (H). The weekly average is given first (I) followed by

the daily averages (J). Prime time data for both key and other items is given before non-prime time data.

Problem Report (Page 4) - This report section gives you an overview of problem items and problem periods. Only those items whose averages (K) have exceeded the long term limit or whose values have exceeded the long term limit more than 10% of the time (L), are reported upon. The average values for both prime and non-prime time (M), the long term limit (N) and the percent of time over the limit (P) are all given followed by a comment line (Q), intended to point you to an area for further investigation. As in the Daily System Utilization Problem Report, the comments are intended only as a pointer to a possible problem area or to recommend some further course of action. There may be more than one possible cause for a given problem (for example, such as high overhead) and there is often more than one solution which can be applied. The alternatives must always be evaluated carefully. The comments, in and of themselves, are not intended to recommend a solution to a given problem. Comments are defined in the xxxxDR.RFD file. You may modify or delete them.

In addition to reporting which items had problems over the past week, this section also reports which days experienced the most time in a problem state. Again the item is given along with the percent of time that the item's values were over the long term limit during each day (R) in the week. The prime time summary is given before the non-prime time summary.

FROM: 14-FEB-82 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

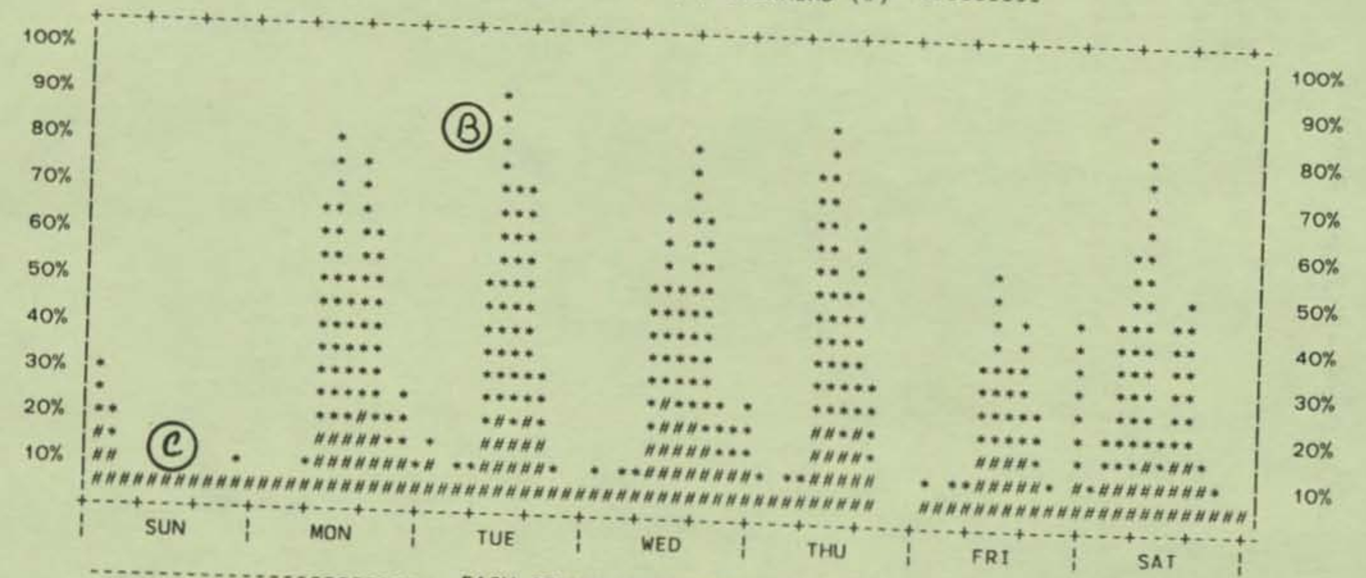
- AMAR -
WEEKLY UTILIZATION SUMMARY REPORT

PAGE: 1

PUT ANY TITLE HERE
SYSTEM: PATH   PRIME TIME: 0800 - 1700

A

--------- CPU UTILIZATION (*) OVERHEAD (#) ---------



---------------------- EACH COLUMN REPRESENTS 2 HOURS ----------------------

------------------ SUMMARY OF KEY UTILIZATION ITEMS ----------------

D

-------AVERAGE-----	% CPU UTIL	# JOBS LOGGED IN	# LINES IN USE	% IDLE TIME	% LOST TIME	% OVHD TIME	ACT SWAP RATIO	SWAPPING BLKS/SEC	USER DSK BLKS/SEC	# MTAS ASSIGNED
----PRIME TIME-----	59%	59	35	41%	1%	15%	.2	33	79	0
---NON-PRIME TIME--	13%	24	5	87%	0%	5%	.0	1	30	0

-------AVERAGE-----	PGS USER MEMORY	% SYSTEM UPTIME	% AMAR CLK TIME	# SYSTEM RELOADS
----PRIME TIME-----	1735	99.9%	98.8%	1
---NON-PRIME TIME--	1735	96.4%	95.0%	4

E F G

Figure 1-3

FROM: 14-FEB-82 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
WEEKLY UTILIZATION DETAIL REPORT

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

I

--- PRIME TIME ---

J

KEY UTILIZATION ITEMS	AVERAGE OF CURRENT -WEEK-	-SUN-	-MON-	-TUE-	-WED-	-THU-	-FRI-	-SAT-
		DAILY AVERAGE						
% CPU UTIL	59%		66%	64%	62%	63%	39%	
# JOBS LOGGED IN	59		58	56	60	64	59	
# LINES IN USE	35		35	36	36	37	33	
% IDLE TIME	41%		34%	36%	38%	37%	61%	
% LOST TIME	1%		1%	1%	1%	1%	2%*	
% OVHD TIME	15%		14%	15%	17%	16%	12%	
ACT SWAP RATIO	.2		.2	.2	.2	.2	.1	
SWAPPING BLKS/SEC	33		35	23	33	45	28	
USER DSK BLKS/SEC	79		60	82	80	114 *	61	
# MTAS ASSIGNED	0		0	0	1	0	1	
PGS USER MEMORY	1735		1737	1735	1735	1735	1732	
% SYSTEM UPTIME	99.9%		100.0%	100.0%	99.6%	100.0%	100.0%	
% AMAR CLK TIME	98.8%		100.0%	100.0%	94.0%	100.0%	100.0%	
# SYSTEM RELOADS	1		0	0	1	0	0	

--- PRIME TIME ---

OTHER UTILIZATION ITEMS	AVERAGE OF CURRENT -WEEK-	-SUN-	-MON-	-TUE-	-WED-	-THU-	-FRI-	-SAT-
		DAILY AVERAGE						
USER UUOS/SEC	249 *	H	318 *	219 *	296 *	238 *	179	
# JOBS BLK IO Q	1.0		.7	1.5 *	1.0	1.1	.6	
%PK DEVPO FREE SPC	9%*		9%*	8%*	4%*	10%*	11%	
%PK HRS20 FREE SPC	13%		---	18%	11%	11%	6%*	
# JOBS IN RUN Q	2		2	3	3	3	2	
% SWAP SPC LEFT	33		39	51	33	18 *	25 *	
# JOBS TTY IO Q	17 *		17 *	16 *	16 *	18 *	17 *	

--- NON-PRIME TIME ---

KEY UTILIZATION ITEMS	AVERAGE OF CURRENT -WEEK-	-SUN-	-MON-	-TUE-	-WED-	-THU-	-FRI-	-SAT-
		DAILY AVERAGE						
% CPU UTIL	13%	7%	9%	8%	15%	8%	11%	28%
# JOBS LOGGED IN	24	21	23	24	25	26	26	27
# LINES IN USE	5	4	4	5	5	6	5	5
% IDLE TIME	87%	93%	91%	92%	85%	92%	89%	72%
% LOST TIME	0%	0%	0%	0%	0%	0%	0%	0%
% OVHD TIME	5%	5%	5%	4%	5%	4%	5%	7%
ACT SWAP RATIO	.0	.0	.0	.0	.0	.0	.0	.0
SWAPPING BLKS/SEC	1	0	0	1	2	2	0	1
USER DSK BLKS/SEC	30	6	14	7	52	5	27	74
# MTAS ASSIGNED	0	0	0	0	0	0	0	0
PGS USER MEMORY	1735	1737	1736	1735	1735	1735	1732	1732

* = OVER LONG TERM LIMITS > = GREATER THAN OR EQUAL TO < = LESS THAN OR EQUAL TO

---------- CONTINUED NEXT PAGE -------

Figure 1-3 (continued)

FROM: 14-FEB-82 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
WEEKLY UTILIZATION DETAIL REPORT

PAGE: 3

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

--- NON-PRIME TIME ---

KEY UTILIZATION ITEMS	AVERAGE OF CURRENT -WEEK-	DAILY AVERAGE -SUN-	-MON-	-TUE-	-WED-	-THU-	-FRI-	-SAT-
% SYSTEM UPTIME	96.4%	100.0%	99.8%	99.8%	99.8%	71.4%	100.0%	100.0%
% AMAR CLK TIME	95.0%	100.0%	96.5%	97.6%	99.6%	64.9%	100.0%	100.0%
# SYSTEM RELOADS	4	0	1	1	1	1	0	0

--- NON-PRIME TIME ---

OTHER UTILIZATION ITEMS	AVERAGE OF CURRENT -WEEK-	DAILY AVERAGE -SUN-	-MON-	-TUE-	-WED-	-THU-	-FRI-	-SAT-
%PK DEVPO FREE SPC	9%*	3%*	19%	9%*	5%*	2%*	11%	11%
%PK HRS20 FREE SPC	15%		---	27%	11%	10%*	7%*	---

* = OVER LONG TERM LIMITS > = GREATER THAN OR EQUAL TO < = LESS THAN OR EQUAL TO

Figure 1-3 (continued)

FROM: 14-FEB-82 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
WEEKLY UTILIZATION PROBLEM REPORT

PUT ANY TITLE HERE
SYSTEM: PATH   PRIME TIME: 0800 - 1700

SUMMARY OF PROBLEM RESOURCES

ITEM		--- PRIME TIME ---			--- NON-PRIME TIME ---			COMMENTS
		M	N	P	M	N	P	Q
		AVG. VALUE	LONG TERM LIMIT	% TIME OVER LIMIT	AVG. VALUE	LONG TERM LIMIT	% TIME OVER LIMIT	
% IDLE	TIME	41%	<30%	37.9%	87%	<30%	3.2%	CPU PRESSED: CHK WORKLD DATA FIRST
% LOST	TIME	1%	>2%	17.7%	0%	>2%	.7%	MEMORY SHORTAGE OR SWAP DEVICE SLOW
% OVHD	TIME	15%	>20%	14.3%	5%	>20%	.0%	OVERHEAD TOO HIGH: INVESTIGATE
USER	UUOS/SEC	249	>200	49.3%	48	>200	7.2%	USER PGM PROBLEM: CHK WORKLD DATA
# JOBS	BLK IO Q	1.0	>1.5	25.5%	.3	>1.5	4.4%	BACKUP OF I/O JOBS: CHK CONTENTION
%PK DEVPO	FREE SPC	9%	<10%	46.7%	9%	<10%	41.0%	DELETE UNNECESSARY FILES
%PK HRS20	FREE SPC	13%	<10%	26.0%	15%	<10%	35.9%	DELETE UNNECESSARY FILES
# JOBS	IN RUN Q	2	>5	11.9%	1	>5	.2%	CPU BOTTLENECK OR SCHEDULER SLOW
% SWAP	SPC LEFT	33	<30	46.1%	78	<30	.1%	SWAP SPACE LOW: ALLOCATE MORE
# JOBS	TTY IO Q	17	>16	64.1%	1	>16	.0%	INTERACTIVE USE HIGH: CHK RESPONSE
USER DSK	BLKS/SEC	79	>100	26.0%	30	>100	9.2%	DISK I/O RATE HIGH: CHK PACK I/O
% CPU	UTIL	59%	>70%	37.9%	13%	>70%	3.2%	CPU PRESSED: CHK WORKLD DATA FIRST

SUMMARY OF PROBLEM PERIODS

--- PRIME TIME ---

------- ITEM -------		LONG TERM LIMIT	SUN	MON	TUE	WED	THU	FRI	SAT
			% TIME OVER LIMIT						R
% LOST	TIME	>2%						29.6%	
USER	UUOS/SEC	>200		49.9%	44.7%	68.9%	57.4%		
# JOBS	BLK IO Q	>1.5			40.2%				
%PK DEVPO	FREE SPC	<10%		68.9%	56.1%	100.0%	11.7%		
%PK HRS20	FREE SPC	<10%						100.0%	
% SWAP	SPC LEFT	<30					93.5%	78.8%	
# JOBS	TTY IO Q	>16		63.1%	62.6%	58.5%	69.1%	66.7%	
USER DSK	BLKS/SEC	>100					43.0%		

SUMMARY OF PROBLEM PERIODS

--- NON-PRIME TIME ---

------- ITEM -------		LONG TERM LIMIT	SUN	MON	TUE	WED	THU	FRI	SAT
			% TIME OVER LIMIT						
%PK DEVPO	FREE SPC	<10%	89.5%		45.3%	78.5%	82.4%		
%PK HRS20	FREE SPC	<10%					17.6%	100.0%	

* = OVER LONG TERM LIMITS   > = GREATER THAN OR EQUAL TO   < = LESS THAN OR EQUAL TO

Figure 1-3 (continued)

1.3.1.3 Monthly Utilization Report -

The Monthly Utilization Report provides the following features:

A graph which shows the average CPU utilization and overhead for each day in the month split out by prime and non-prime time.

A prime/non-prime time summary for the month and each week in the month of utilization and performance items such as idle time, overhead, swapping rate, disk I/O rate, system uptime, etc.

A prime/ non-prime time summary of key items and problem items during the month.

The Monthly Utilization Report is separated into three sections - a Summary Report, a Detail Report, and a Problem Report. Refer to Figure 1-4. The format of the report header is the same as for the Daily System Utilization Report.

<u>Summary Report (Page 1)</u> - This report section gives management a quick overview of system utilization and peak processing periods during the previous month. It is often used in conjunction with the Workload AMAR "Program Name Report", "Batch Vs. Timesharing Report", or the "PPN Report". The graph at the top of page 1 (A) represents processor 0. It is a bar chart which shows CPU utilization (*) and overhead (#) for prime and non-prime time periods for each day in the month. The first day typically printed on the prime time graph will be a Monday (B), the second day in the report period (25-JAN-82 in this example). To help you locate other days in the month, a fiscal week ending date is printed followed by the symbol "¦" (C) which points to that date on the graph.

The difference between the top of the CPU utilization (D) and overhead (E) bars represents CPU time consumed by user jobs, lost time, and some embedded priority interrupt (PI) time. Bars approaching 100% indicate days when the CPU was heavily loaded. These days should be examined more closely by looking at the Daily System Utilization Report.

Below the graph are prime and non-prime time averages for eleven key items (F). These averages are intended to give a quick picture of the previous month's usage. You have the option of selecting key items from the entire list of items that AMAR collects. The items and their definitions are listed in the Appendix called "System AMAR Item Definitions". In addition to key items, the percent of system uptime (G), the percent of time that AMAR measured the system (H), and the total number of system reloads (I) during the past month is presented.

<u>Detail Report (Pages 2-3)</u> - This report section enables you to compare the total resource usage for the month with the resource usage during each of the weeks in the month. This report section, in conjunction with the Problem Report, allows you to trace back through the month to determine during which weeks the most resource consumption and/or problems occurred. You can then refer to the Weekly and Daily System Utilization Reports to trace back and find the problem days and hours.

Averages for key utilization items are presented first, followed by averages for other utilization items. Other utilization items are those whose averages have exceeded the long term limit as listed on the Problem Report and as defined in the xxxxDR.RFD Report File Description. Each average exceeding the limit is flagged by an asterisk (*) (J). The monthly average is given first (K) followed by the weekly averages (L). Prime time data for both key and other items is given before non-prime time data.

Problem Report (Page 4) - This section, like that of the Daily System and Weekly Utilization Problem Reports, gives you an overview of problem resources and problem periods. Only those items whose averages (M) have exceeded the long term limit, or whose sample values (N) have exceeded the long term limit more that 10% of the time, are reported upon. The average values for both prime and non-prime time (P), the long term limit (Q), and the percent of time over limit (R) are all given followed by a comment line (S). The comment line is intended to point you to an area for further investigation. As in the Daily System and Weekly Utilization Problem Reports, the comments are intended only as a pointer to a possible problem area or to recommend some further course of action. The comments are not intended to suggest a solution to a problem. Since there may be more than one reason for an item exceeding limits and more than one solution to a given problem, careful analysis and weighing of alternatives is recommended. Comments are defined in the xxxxDR.RFD file. You may modify or delete them.

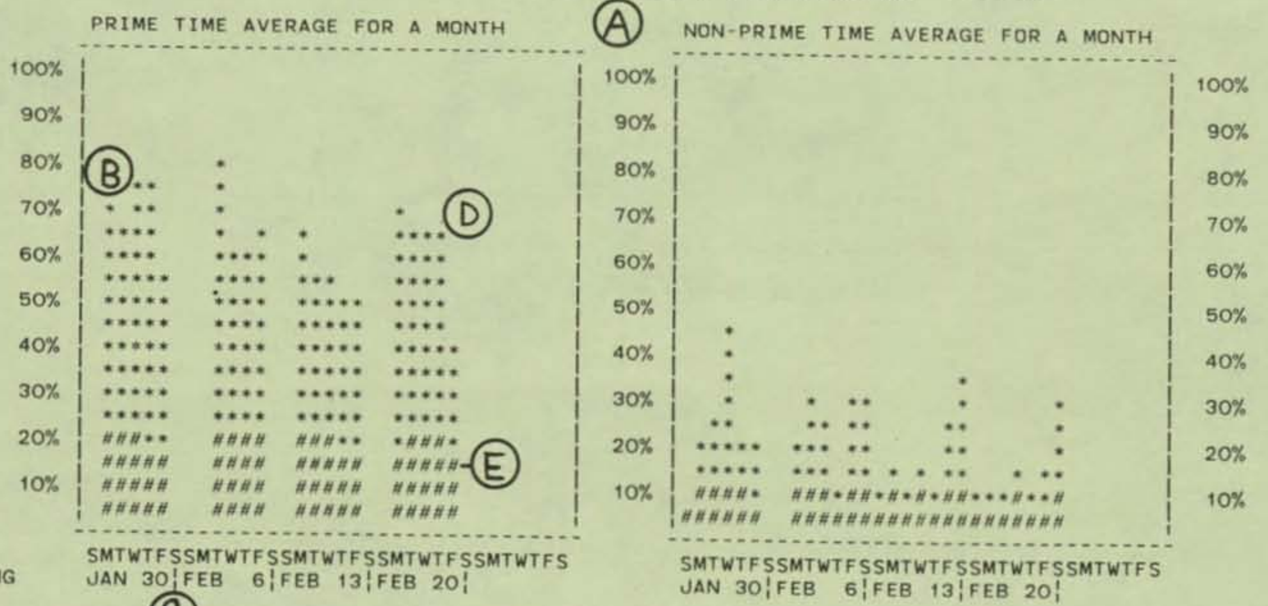
In addition to reporting which items had problems over the past month, this section also reports during which weeks the items spent the most time in a problem state. Again the item is given along with the percent of time that the item's values were over the long term limit during each week (T) in the past month. The prime time summary is given before the non-prime time summary.

FROM: 24-JAN-82 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
MONTHLY UTILIZATION SUMMARY REPORT

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

--------- CPU UTILIZATION (*) OVERHEAD (#) ---------



-------------------------- EACH COLUMN REPRESENTS 1 DAY --------------------------

------------------ SUMMARY OF KEY UTILIZATION ITEMS ------------------ F

-------AVERAGE-----	% CPU UTIL	# JOBS LOGGED IN	# LINES IN USE	% IDLE TIME	% LOST TIME	% OVHD TIME	ACT SWAP RATIO	SWAPPING BLKS/SEC	USER DSK BLKS/SEC	# MTAS ASSIGNED
----PRIME TIME-----	60%	59	36	40%	1%	15%	.2	33	72	0
---NON-PRIME TIME--	18%	24	5	82%	0%	6%	.0	1	32	0

-------AVERAGE-----	PGS USER MEMORY	% SYSTEM UPTIME	% AMAR CLK TIME	# SYSTEM RELOADS
----PRIME TIME-----	1736	94.7%	93.1%	6
---NON-PRIME TIME--	1735	83.9%	82.7%	9

G H I

Figure 1-4

FROM: 24-JAN-82 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
MONTHLY UTILIZATION DETAIL REPORT

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

--- PRIME TIME ---

K L

KEY UTILIZATION ITEMS	AVERAGE OF CURRENT -MONTH-	-JAN 30-	-FEB 6-	-FEB 13-	-FEB 20-
		WEEKLY AVERAGE			
% CPU UTIL	60%	65%	64%	54%	59%
# JOBS LOGGED IN	59	62	62	56	59
# LINES IN USE	36	37	36	35	35
% IDLE TIME	40%	35%	36%	46%	41%
% LOST TIME	1%	1%	1%	1%	1%
% OVHD TIME	15%	15%	17%	15%	15%
ACT SWAP RATIO	.2	.2	.2	.2	.2
SWAPPING BLKS/SEC	33	41	35	25	33
USER DSK BLKS/SEC	72	69	67	71	79
# MTAS ASSIGNED	0	0	1	1	0
PGS USER MEMORY	1736	1737	1733	1736	1735
% SYSTEM UPTIME	94.7%	99.9%	79.0%	99.9%	99.9%
% AMAR CLK TIME	93.1%	98.7%	77.1%	98.0%	98.8%
# SYSTEM RELOADS	6	2	2	1	1

--- PRIME TIME ---

OTHER UTILIZATION ITEMS	AVERAGE OF CURRENT -MONTH-	-JAN 30-	-FEB 6-	-FEB 13-	-FEB 20-
		WEEKLY AVERAGE			
USER UUOS/SEC	253 * J	245 *	287 *	239 *	249 *
# JOBS BLK IO Q	1.3	1.2	1.5 *	1.3	1.0
%PK DEVPO FREE SPC	26%	47%	38%	11%	9%*
# JOBS IN RUN Q	3	3	3	2	2
% SWAP SPC LEFT	39	38	36	47	33
# JOBS TTY IO Q	17 *	17 *	16 *	16 *	17 *

--- NON-PRIME TIME ---

KEY UTILIZATION ITEMS	AVERAGE OF CURRENT -MONTH-	-JAN 30-	-FEB 6-	-FEB 13-	-FEB 20-
		WEEKLY AVERAGE			
% CPU UTIL	18%	19%	24%	16%	13%
# JOBS LOGGED IN	24	24	25	23	24
# LINES IN USE	5	5	6	5	5
% IDLE TIME	82%	81%	76%	84%	87%
% LOST TIME	0%	0%	0%	0%	0%
% OVHD TIME	6%	6%	7%	6%	5%
ACT SWAP RATIO	.0	.1	.0	.0	.0
SWAPPING BLKS/SEC	1	1	1	1	1
USER DSK BLKS/SEC	32	25	43	30	30
# MTAS ASSIGNED	0	0	0	0	0
PGS USER MEMORY	1735	1737	1733	1736	1735
% SYSTEM UPTIME	83.9%	72.6%	70.3%	96.3%	96.4%

* = OVER LONG TERM LIMITS > = GREATER THAN OR EQUAL TO < = LESS THAN OR EQUAL TO

---------- CONTINUED NEXT PAGE --------

Figure 1-4 (continued)

FROM: 24-JAN-82 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
MONTHLY UTILIZATION DETAIL REPORT

PAGE: 3

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

--- NON-PRIME TIME ---

KEY UTILIZATION ITEMS	AVERAGE OF CURRENT -MONTH-	WEEKLY AVERAGE -JAN 30-	-FEB 6-	-FEB 13-	-FEB 20-
% AMAR CLK TIME	82.7%	71.9%	69.6%	94.5%	95.0%
# SYSTEM RELOADS	9	1	1	3	4

--- NON-PRIME TIME ---

OTHER UTILIZATION ITEMS	AVERAGE OF CURRENT -MONTH-	WEEKLY AVERAGE -JAN 30-	-FEB 6-	-FEB 13-	-FEB 20-
%PK DEVPO FREE SPC	25%	51%	36%	13%	9%*

* = OVER LONG TERM LIMITS > = GREATER THAN OR EQUAL TO < = LESS THAN OR EQUAL TO

Figure 1-4 (continued)

FROM: 24-JAN-82 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
MONTHLY UTILIZATION PROBLEM REPORT

PUT ANY TITLE HERE
SYSTEM: PATH   PRIME TIME: 0800 - 1700

SUMMARY OF PROBLEM RESOURCES

ITEM		--- PRIME TIME --- P AVG. VALUE	Q LONG TERM LIMIT	R % TIME OVER LIMIT	--- NON-PRIME TIME --- P AVG. VALUE	Q LONG TERM LIMIT	R % TIME OVER LIMIT	S COMMENTS
% IDLE	TIME	40%	<30%	38.8% N	82%	<30%	4.9%	CPU PRESSED: CHK WORKLD DATA FIRST
% LOST	TIME	1%	>2%	17.0%	0%	>2%	.7%	MEMORY SHORTAGE OR SWAP DEVICE SLOW
% OVHD	TIME	15%	>20%	16.7%	6%	>20%	.3%	OVERHEAD TOO HIGH: INVESTIGATE
USER	UUOS/SEC	253 M	>200	53.0%	51	>200	6.5%	USER PGM PROBLEM: CHK WORKLD DATA
# JOBS	BLK IO Q	1.3	>1.5	33.4%	.3	>1.5	5.5%	BACKUP OF I/O JOBS: CHK CONTENTION
%PK DEVPO	FREE SPC	26%	<10%	26.3%	25%	<10%	25.3%	DELETE UNNECESSARY FILES
# JOBS	IN RUN Q	3	>5	13.1%	1	>5	.3%	CPU BOTTLENECK OR SCHEDULER SLOW
% SWAP	SPC LEFT	39	<30	30.9%	79	<30	.0%	SWAP SPACE LOW: ALLOCATE MORE
# JOBS	TTY IO Q	17	>16	60.7%	1	>16	.0%	INTERACTIVE USE HIGH: CHK RESPONSE
USER DSK	BLKS/SEC	72	>100	23.7%	32	>100	11.3%	DISK I/O RATE HIGH: CHK PACK I/O
% CPU	UTIL	60%	>70%	38.8%	18%	>70%	4.9%	CPU PRESSED: CHK WORKLD DATA FIRST

SUMMARY OF PROBLEM PERIODS

--- PRIME TIME ---

------- ITEM -------		LONG TERM LIMIT	% TIME OVER LIMIT: JAN 30	T FEB 6	FEB 13	FEB 20
USER	UUOS/SEC	>200	54.1%	60.5%	49.8%	49.3%
# JOBS	BLK IO Q	>1.5		41.8%		
%PK DEVPO	FREE SPC	<10%				46.2%
# JOBS	TTY IO Q	>16	65.5%	56.6%	55.7%	64.1%

SUMMARY OF PROBLEM PERIODS

--- NON-PRIME TIME ---

------- ITEM -------		LONG TERM LIMIT	% TIME OVER LIMIT: JAN 30	FEB 6	FEB 13	FEB 20
%PK DEVPO	FREE SPC	<10%				41.0%

* = OVER LONG TERM LIMITS   > = GREATER THAN OR EQUAL TO   < = LESS THAN OR EQUAL TO

Figure 1-4 (continued)

1.3.2 Trend Analysis Reports

1.3.2.1 Weekly Trend Analysis Report -

The Weekly Trend Analysis Report provides the following features:

- Graphs of CPU utilization and overhead for up to the past 13 weeks split out by prime and non-prime time.

- A table of prime and non-prime time averages for key items for up to the past 13 weeks. The table allows for quick comparison of item values. The default list of key items in this report is usually longer than the default list of key itmes in the Daily Utilization Summary Report. The list may be changed by editing the xxxxDR.RFD file.

- A trend analysis section which shows relative usage for each week plus the growth/month during the period and an indication of whether or not a short term linear trend exists.

The format of the report header (refer to Figure 1-5) is as follows:

Lines 1 and 2 - Specify the beginning and ending dates of the report, the name of the report, and the page number. The report period begins at approximately 00:01 AM of the first date and ends at approximately 23:59 PM of the second date.

Line 3 - Specifies the Data Center Name.

Line 4 - Specifies the 4 character system code and the period of time during the day which is considered prime time.

The graphs at the top of page 1 represent processor 0. They show the CPU utilization (*) and overhead (#) averages for the past 13 weeks. Prime time averages are on the left (A) and non-prime on the right (B). At most 13 data points will be plotted for each item. Fewer may be plotted if the database retention period for the weekly granularity records is less than 13. The graphs and the tables which follow are always read from left to right with the most recent week appearing on the right (C). The symbols *, #, and @ represent the actual data point. The dots in between the data points are for visual effect only and do not represent any item values.

By observing the shape and slope of the two graphs, one may obtain an impression of relative utilization of the processor during prime and non-prime time. This will be helpful in determining whether some load balancing of the machine is required. Processing cycles, such as for monthly financial closings, may also begin to show up. It may also be possible to observe some short term linear trend in usage. However, it is difficult to be sure that a trend exists just from observing the data points. For verification purposes, a trend line is computed by the reporting program and the percent of fit of the data points (TREND LINE FIT) is given on page 3 of the sample report.

Below the graph is a table of prime time averages for the past 13 weeks for key utilization items. Key utilization items are defined by you in the xxxxDR.RFD file. You have the option of selecting "key" items from the entire list of items that AMAR measures. The items and their definitions are listed in the Appendix called "System AMAR Item Definitions". The set of key items selected for the Trend Analysis Reports may differ from those selected for any of the other types of reports.

The format for the table header is as follows:

Line 1 (D) - Specifies the fiscal quarter, month and week for which the averages were computed. The earliest week in the period is given first and the most recent week (H) is given last.

Line 2 (E) - Specifies the actual calendar date of the last day in the week (week ending day).

Line 3 (F) - Indicates the distance in time from the most recent week. For example, the week ending DEC 12 (G) was 8 weeks prior to the most recent week listed on the report, FEB 6 (H).

On the left most side of the report are listed the item descriptions. On the right most side (I), are listed the long term limits for each item. The long term limits are specified in the xxxxDR.RFD file and are also user settable. The long term limits apply to daily, weekly and monthly data summary levels. Any average which equals or exceeds the long term limit will be flagged with an asterisk (*) (J). This enables you to immediately spot problem items and periods.

Following the table of prime time averages is the table of non-prime time averages (page 2). Note that there is no table of other items as in the System Utilization Reports. If you want a table of other items, you must define it in the RFD file.

The actual trend analysis information begins on page 3 of the sample report. Prime time data comes first, followed by non-prime time data. Again, the item description is given on the left followed by the averages for the first week (K) in the reporting period (NOV 28 in this example.) Next comes the TABLE OF RELATIVE USAGE PER WEEK (L). There is one column for each week in the reporting period. The order of the columns corresponds to the order of the weeks in the preceeding tables. The table is scaled so that 8 represents the average value for the item. The TABLE OF RELATIVE USAGE PER WEEK enables you to get a quick picture of the amount of variance among the weekly averages. For example, one can see that there was very little variation in average jobsize (M) from week to week while the values for user UUO's per second (N) varied widely. The numbers in the TABLE OF RELATIVE USAGE PER WEEK have meaning only in relation to one another. The numbers from one item cannot be compared with the numbers from another item. For example, an 8 for average jobsize has no relationship to an 8 for user UUO's per second and, in fact, their actual averages will typically be very different.

Following the TABLE OF RELATIVE USAGE PER WEEK is the column of averages for the most recent week (P), FEB 6 in this example. Next comes the growth per month (Q) calculated over the reporting period followed by the percent of trend line fit (R). The growth per month is expressed in the same units as the item's values. The reporting period is typically 13 weeks although it may be shorter if there are fewer than 13 weeks worth of data contained in the database. A negative number in the GROWTH/MONTH column means that the average values are decreasing; a positive number means an increase in growth. The TREND LINE FIT column indicates how much scatter there is around the trend line. For example, a 90% trend line fit means that most of the data points lie very close to the trend line. The TREND LINE FIT will be given only if it is greater than or equal to 70%. There are two comments that may appear in place of the percentages. ERRATIC VALUES means that data was widely scattered around the trend line and no strong linear pattern could be found in the data. The TREND LINE FIT would be less than 70%. UNCHANGING values refers to values that were relatively constant from week to week. The GROWTH/MONTH would be 0 and the TREND LINE FIT would be at least 70%.

There are no predictions given based on weekly data. Predictions are found only on the Monthly Trend Analysis Report.

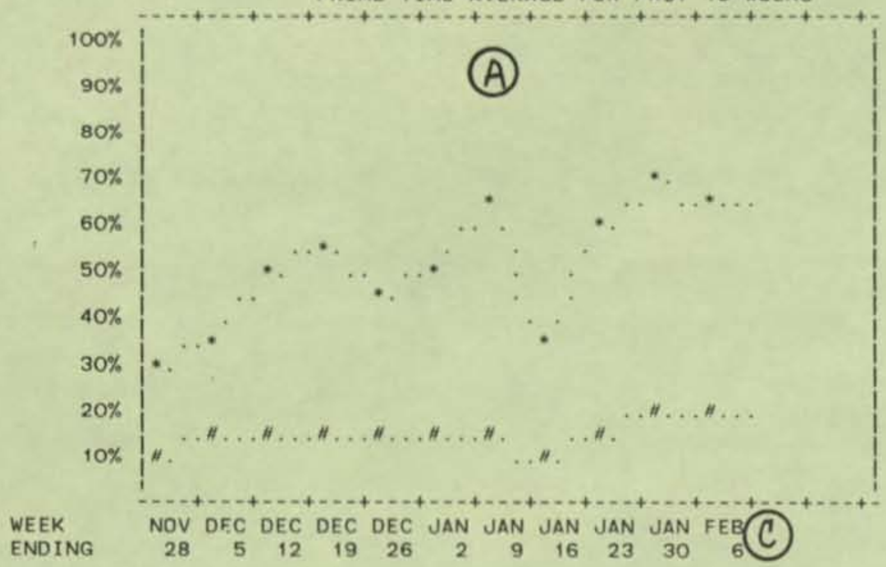
FROM: 22-NOV-81 (SUNDAY)
THRU: 06-FEB-82 (SATURDAY)

- AMAR -
WEEKLY TREND ANALYSIS REPORT

PUT ANY TITLE HERE
SYSTEM: PATH   PRIME TIME: 0800 - 1700

--------- CPU UTILIZATION (*) OVERHEAD (#) AND BOTH (@) ---------

PRIME TIME AVERAGE FOR PAST 13 WEEKS



NON-PRIME TIME AVERAGE FOR PAST 13 WEEKS

PRIME TIME AVERAGES OF KEY UTILIZATION ITEMS

FISCAL WEEK ENDING ------- ITEM -------	Q2M3W1 NOV 28 W-10	Q2M3W2 DEC 5 W-09	G Q2M3W3 DEC 12 W-08	Q2M3W4 DEC 19 W-07	Q2M3W5 DEC 26 W-06	Q3M1W1 JAN 2 W-05	Q3M1W2 JAN 9 W-04	Q3M1W3 JAN 16 W-03	Q3M1W4 JAN 23 W-02	Q3M2W1 JAN 30 W-01	Q3M2W2 D FEB 6 E H W-00 F	I LONG TERM LIMIT
% CPU UTIL	28	32	47	53	45	48	62	31	55	65	64	>70%
% IDLE TIME	72	68	53	47	55	52	38	69	45	35	36	<30%
% OVHD TIME	10	11	12	13	12	13	13	10	15	15	17	>20%
% LOST TIME	2*	1	1	0	0	0	0	1	1	1	1	>2%
USER UUOS/SEC	143	183	193	207*	165	194	935*	166	240*	245*	287*	>200
CONTEXT SWTS/SEC					61	67	66	42	80	80	91	
ACT SWAP RATIO	.1	.1	.1	.1	.1	.1	.1	.1	.2	.2	.2	>.5
ACTV JOB % USR CR					9	12	10	8	18	20	19	
ALL JOBS % USR CR					61	66	71	64	94	100	102	
% RN JOB IN MEM					100	100	100	100	100	100	100	
AVG JOB SIZE	36	32	33	31	34	36	36	35	39	39	40	
# JOBS LOGGED IN	35	36	39	40	42	40	44	40	54	62	62	
PGS USER MEMORY	1228	1230	1339	1743	1741	1736	1739	1740	1737	1737	1733	<768
# JOBS TTY IO Q	7	6	7	8	5	6	6	6	13	17*	16*	>16
USER DSK BLKS/SEC	32	39	53	64	51	59	78	44	59	69	67	>100
# JOBS IN RUN Q	1	1	2	2	2	2	2	1	2	3	3	>5
SWAPPING BLKS/SEC	21	15	18	7	5	6	10	8	35	41	35	>160
PK DSKEO SWPS/SEC	21	15	18	7	5	6	10	8	35	41	35	>160

* = EXCEEDS LONG TERM LIMIT   > = GREATER THAN OR EQUAL TO   < = LESS THAN OR EQUAL TO   ---------- CONTINUED NEXT PAGE ----------

Figure 1-5

FROM: 22-NOV-81 (SUNDAY)
THRU: 06-FEB-82 (SATURDAY)

- AMAR -
WEEKLY TREND ANALYSIS REPORT

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

PRIME TIME AVERAGES OF KEY UTILIZATION ITEMS

FISCAL WEEK ENDING ------- ITEM -------	Q2M3W1 NOV 28 W-10	Q2M3W2 DEC 5 W-09	Q2M3W3 DEC 12 W-08	Q2M3W4 DEC 19 W-07	Q2M3W5 DEC 26 W-06	Q3M1W1 JAN 2 W-05	Q3M1W2 JAN 9 W-04	Q3M1W3 JAN 16 W-03	Q3M1W4 JAN 23 W-02	Q3M2W1 JAN 30 W-01	Q3M2W2 FEB 6 W-00	LONG TERM LIMIT
AVG SCHD RSP TIME	6	4	7	5	3	4	5	4	10	28	10	>200
# MTAS ASSIGNED	0	0	0	0	1	1	0	0	0	0	1	
# LINES IN USE	30	30	32	28	22	18	21	19	31	37	36	
%TY CHNK IN USE					36	39	38	38	40	44	40	
SCN INTR RCVS/SEC					4	9	9	8	11	11	11	
SCN INTR XMTS/SEC					116	239	212	193	257	282	252	
% SYSTEM UPTIME	87.7	98.9	96.6	84.6	100.0	99.7	99.8	77.4	92.3	99.9	79.0	
% AMAR CLK TIME	86.2	90.9	94.1	78.7	96.6	95.1	97.8	61.7	84.8	98.7	77.1	
# SYSTEM RELOADS	3	6	6	3	0	3	3	6	10	2	2	

NON-PRIME TIME AVERAGES OF KEY UTILIZATION ITEMS

FISCAL WEEK ENDING ------- ITEM -------	Q2M3W1 NOV 28 W-10	Q2M3W2 DEC 5 W-09	Q2M3W3 DEC 12 W-08	Q2M3W4 DEC 19 W-07	Q2M3W5 DEC 26 W-06	Q3M1W1 JAN 2 W-05	Q3M1W2 JAN 9 W-04	Q3M1W3 JAN 16 W-03	Q3M1W4 JAN 23 W-02	Q3M2W1 JAN 30 W-01	Q3M2W2 FEB 6 W-00	LONG TERM LIMIT
% CPU UTIL	12	9	13	18	8	13	20	11	17	19	24	>70%
% IDLE TIME	88	91	87	82	92	87	80	89	83	81	76	<30%
% OVHD TIME	5	4	5	5	4	5	5	5	6	6	7	>20%
% LOST TIME	0	0	0	0	0	0	0	0	0	0	0	>2%
USER UUOS/SEC	50	44	49	50	31	50	757*	44	56	45	64	>200
CONTEXT SWTS/SEC					9	16	10	14	23	23	36	
ACT SWAP RATIO	.0	.0	.0	.0	.0	.0	.0	.0	.0	.1	.0	>.5
ACTV JOB % USR CR					2	4	3	3	4	5	6	
ALL JOBS % USR CR					31	30	29	27	28	31	35	
% RN JOB IN MEM					100	100	100	100	100	100	100	
AVG JOB SIZE	25	23	21	22	25	26	25	25	25	25	28	
# JOBS LOGGED IN	23	22	24	22	24	23	22	22	23	24	25	
PGS USER MEMORY	1298	1229	1444	1743	1741	1738	1739	1740	1738	1737	1733	<768
# JOBS TTY IO Q	1	1	1	0	0	0	1	1	1	1	1	>16
USER DSK BLKS/SEC	28	17	27	33	16	31	17	23	37	25	43	>100
# JOBS IN RUN Q	1	1	1	1	1	1	1	1	1	1	1	>5
SWAPPING BLKS/SEC	0	0	0	0	0	0	0	0	0	1	1	>160
PK DSKEO SWPS/SEC	0	0	0	0	0	0	0	0	0	1	1	>160
AVG SCHD RSP TIME	3	1	2	3	2	10	2	2	2	1	3	>200
# MTAS ASSIGNED	0	0	0	0	0	0	0	0	0	0	0	
# LINES IN USE	19	18	18	17	6	5	5	4	4	5	6	
%TY CHNK IN USE					34	35	35	36	36	37	35	
SCN INTR RCVS/SEC					0	0	1	0	0	0	1	
SCN INTR XMTS/SEC					6	8	11	12	17	18	23	
% SYSTEM UPTIME	85.8	96.5	70.4	85.8	97.4	99.9	96.9	82.8	92.3	72.6	70.3	
% AMAR CLK TIME	83.8	94.7	69.8	85.8	96.2	99.4	95.4	80.8	87.7	71.9	69.6	
# SYSTEM RELOADS	4	3	2	1	3	1	3	10	11	1	1	

* = EXCEEDS LONG TERM LIMIT > = GREATER THAN OR EQUAL TO < = LESS THAN OR EQUAL TO

---------- CONTINUED NEXT PAGE ----------

Figure 1-5 (continued)

FROM: 22-NOV-81 (SUNDAY)
THRU: 06-FEB-82 (SATURDAY)

- AMAR -
WEEKLY TREND ANALYSIS REPORT

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

PRIME TIME TREND OF KEY UTILIZATION ITEMS

------- ITEM -------		FIRST WEEKLY AVG	TABLE OF RELATIVE USAGE PER WEEK											LAST WEEKLY AVG	--11 WEEK-- -- TREND -- GROWTH /MONTH	LINE FIT
% CPU	UTIL	28	4	5	7	8	7	7	10	5	9	10	10	64	ERRATIC	VALUES
% IDLE	TIME	72	11	10	8	7	8	8	6	10	7	5	5	36	ERRATIC	VALUES
% OVHD	TIME	10	6	6	7	8	7	8	8	6	9	9	10	17	ERRATIC	VALUES
% LOST	TIME	2	14	9	9	4	3	3	4	6	12	10	10	1	ERRATIC	VALUES
USER	UUOS/SEC	143	4	5	5	6	4	5	25	4	6	7	8	287	ERRATIC	VALUES
CONTEXT	SWTS/SEC	61					7	7	7	4	9	9	10	91	ERRATIC	VALUES
ACT SWAP	RATIO	.1	6	7	8	6	5	6	6	5	10	11	10	.2	ERRATIC	VALUES
ACTV JOB	% USR CR	9					5	6	5	4	10	11	10	19	+8.62	70%
ALL JOBS	% USR CR	61					6	6	7	6	9	9	10	102	+33.28	82%
% RN JOB	IN MEM	100					8	8	8	8	8	8	8	100	-0.35	94%
AVG JOB	SIZE	36	8	7	7	7	7	8	8	8	9	8	9	40	ERRATIC	VALUES
# JOBS	LOGGED IN	35	6	6	7	7	7	7	7	7	9	10	10	62	+11.05	78%
PGS USER	MEMORY	1228	6	6	6	8	8	8	8	8	8	8	8	1733	ERRATIC	VALUES
# JOBS	TTY IO Q	7	6	5	6	6	4	5	5	5	11	14	13	16	ERRATIC	VALUES
USER DSK	BLKS/SEC	32	4	5	7	9	7	8	10	6	8	9	9	67	ERRATIC	VALUES
# JOBS	IN RUN Q	1	5	6	7	8	7	7	8	5	9	10	11	3	ERRATIC	VALUES
SWAPPING	BLKS/SEC	21	8	6	7	3	2	2	4	3	14	17	14	35	ERRATIC	VALUES
PK DSKEO	SWPS/SEC	21	8	6	7	3	2	2	4	3	14	17	14	35	ERRATIC	VALUES
AVG SCHD	RSP TIME	6	5	4	7	4	3	4	5	4	9	25	9	10	ERRATIC	VALUES
# MTAS	ASSIGNED	0	7	5	8	6	12	13	5	3	8	8	14	1	ERRATIC	VALUES
# LINES	IN USE	30	8	8	9	8	6	5	6	5	9	10	10	36	ERRATIC	VALUES
%TY CHNK	IN USE	36					7	7	7	7	8	8	8	40	ERRATIC	VALUES
SCN INTR	RCVS/SEC	4					3	7	7	6	9	9	9	11	+4.42	70%
SCN INTR	XMTS/SEC	116					4	8	7	6	9	9	8	252	ERRATIC	VALUES
% SYSTEM	UPTIME	87.7	7	8	8	7	8	8	8	6	8	8	7	79.0	ERRATIC	VALUES
% AMAR	CLK TIME	86.2	8	8	8	7	9	8	9	5	8	9	7	77.1	ERRATIC	VALUES
# SYSTEM	RELOADS	3	6	12	12	6	0	6	6	12	19	4	4	2	ERRATIC	VALUES

NON-PRIME TIME TREND OF KEY UTILIZATION ITEMS

------- ITEM -------		FIRST WEEKLY AVG	TABLE OF RELATIVE USAGE PER WEEK											LAST WEEKLY AVG	--11 WEEK-- -- TREND -- GROWTH /MONTH	LINE FIT
% CPU	UTIL	12	7	5	7	9	4	7	10	6	9	10	13	24	ERRATIC	VALUES
% IDLE	TIME	88	8	8	8	7	8	8	7	8	7	7	7	76	ERRATIC	VALUES
% OVHD	TIME	5	8	7	8	8	6	8	7	7	9	9	11	7	ERRATIC	VALUES
% LOST	TIME	0	10	8	8	4	2	2	4	8	10	13	19	0	ERRATIC	VALUES
USER	UUOS/SEC	50	3	3	3	3	2	3	48	3	4	3	4	64	ERRATIC	VALUES

PREDICTIONS ARE ONLY MADE USING MONTHLY DATA.
GIVEN THE CURRENT TREND, THERE IS A 90% CHANCE THAT THE ACTUAL VALUES WILL FALL WITHIN THE PREDICTED RANGES.
UNCHANGING VALUES: REFERS TO RELATIVELY CONSTANT VALUES. ERRATIC VALUES: REFERS TO THE LACK OF A STRONG LINEAR PATTERN IN THE DATA.
* = EXCEEDS LONG TERM LIMIT > = GREATER THAN OR EQUAL TO < = LESS THAN OR EQUAL TO ---------- CONTINUED NEXT PAGE ----------

Figure 1-5 (continued)

FROM: 22-NOV-81 (SUNDAY)
THRU: 06-FEB-82 (SATURDAY)

- AMAR -
WEEKLY TREND ANALYSIS REPORT

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

NON-PRIME TIME TREND OF KEY UTILIZATION ITEMS

------- ITEM -------		FIRST WEEKLY AVG	TABLE OF RELATIVE USAGE PER WEEK											LAST WEEKLY AVG	--11 WEEK-- -- TREND -- GROWTH /MONTH	LINE FIT
CONTEXT	SWTS/SEC	9					4	7	4	6	10	10	15	36	+16.77	80%
ACT SWAP	RATIO	.0	8	6	6	8	6	6	5	5	8	9	8	.0	ERRATIC	VALUES
ACTV JOB	% USR CR	2					5	8	6	6	8	10	11	6	+2.08	73%
ALL JOBS	% USR CR	31					8	8	7	7	7	8	9	35	ERRATIC	VALUES
% RN JOB	IN MEM	100					8	8	8	8	8	8	8	100	ERRATIC	VALUES
AVG JOB	SIZE	25	8	7	7	7	8	8	8	8	8	8	9	28	ERRATIC	VALUES
# JOBS	LOGGED IN	23	8	7	8	7	8	8	8	7	8	8	8	25	ERRATIC	VALUES
PGS USER	MEMORY	1298	6	6	7	8	8	8	8	8	8	8	8	1733	ERRATIC	VALUES
# JOBS	TTY IO Q	1	9	7	9	4	2	4	9	11	7	12	13	1	ERRATIC	VALUES
USER DSK	BLKS/SEC	28	8	5	8	9	5	9	5	7	11	7	12	43	ERRATIC	VALUES
# JOBS	IN RUN Q	1	7	7	7	8	7	8	8	7	8	9	9	1	+0.04	99%
SWAPPING	BLKS/SEC	0	9	7	7	4	2	2	4	6	8	16	24	1	ERRATIC	VALUES
PK DSKEO	SWPS/SEC	0	9	7	7	4	2	2	4	6	8	16	24	1	ERRATIC	VALUES
AVG SCHD	RSP TIME	3	10	4	5	9	7	30	4	6	5	4	9	3	ERRATIC	VALUES
# MTAS	ASSIGNED	0	8	4	14	12	4	10	4	5	9	8	9	0	ERRATIC	VALUES
# LINES	IN USE	19	15	14	14	13	4	4	4	3	3	4	5	6	-7.37	71%
%TY CHNK	IN USE	34					7	8	8	8	8	8	8	35	ERRATIC	VALUES
SCN INTR	RCVS/SEC	0					2	2	13	8	8	7	14	1	ERRATIC	VALUES
SCN INTR	XMTS/SEC	6					4	5	6	7	10	10	13	23	+11.48	97%
% SYSTEM	UPTIME	85.8	8	9	6	8	9	9	9	7	8	6	6	70.3	ERRATIC	VALUES
% AMAR	CLK TIME	83.8	8	9	6	8	9	9	9	7	8	7	6	69.6	ERRATIC	VALUES
# SYSTEM	RELOADS	4	8	6	4	2	6	2	6	21	23	2	2	1	ERRATIC	VALUES

PREDICTIONS ARE ONLY MADE USING MONTHLY DATA.
GIVEN THE CURRENT TREND, THERE IS A 90% CHANCE THAT THE ACTUAL VALUES WILL FALL WITHIN THE PREDICTED RANGES.
UNCHANGING VALUES: REFERS TO RELATIVELY CONSTANT VALUES. ERRATIC VALUES: REFERS TO THE LACK OF A STRONG LINEAR PATTERN IN THE DATA.
* = EXCEEDS LONG TERM LIMIT > = GREATER THAN OR EQUAL TO < = LESS THAN OR EQUAL TO

Figure 1-5 (continued)

1.3.2.2 Monthly Trend Analysis Report -

The Monthly Trend Analysis Report is almost identical to the Weekly Trend Analysis Report (refer to the previous section). Major differences are:

All averages are monthly averages.

All dates on the report refer to the fiscal month ending date.

At most 12 months of data are reported.

The format of the first header line (A) for the PRIME TIME AVERAGES OF KEY UTILIZATION ITEMS table indicates the number of fiscal weeks in each particular month. For example, the month ending SEP 26 (B) contained five weeks and was the third month in the first quarter (Q1M3W5).

The trend analysis section of the report, beginning on page 3 of this sample (refer to Figure 1-6), will contain predictions if at least 6 months worth of monthly data is present in the database.

Predictions follow the TREND LINE FIT column. Listed first is a range of values where the item's average is expected to fall within 6 months (C) and then within 12 months (D). An underlying assumption in these predictions is that the current linear trend will continue. If that occurs, there is a 90% chance that the item's actual averages will fall within the predicted ranges. Only the past usage history of the machine is taken into account, not the future plans of the data center or its users. For example, in this sample report, LINES IN USE is expected to grow to between 28-56 (E) by the end of six months. If the data center were to move users to another machine, this number might fall. If several new applications were added, the number might increase more rapidly. Note that some items, such as # MTAS ASSIGNED, show widely varying index numbers (F) while the actual averages are almost always close to or equal to 0. This occurs because the index numbers are computed using higher precision values in the database records. The report values are usually rounded to a lower precision thus masking some of the variation.

The next column, PREDICTED PERIOD WHEN LONG TERM LIMIT REACHED (G), gives the month or range of months when the confidence interval around the trend line exceeds (if it is a high limit) or falls below (if it is a low limit) the long term limit, <u>not</u> when an individual average exceeds or falls below the limit. The confidence interval denotes the edges of a band around the trend line where 90% of the averages fall. Dates in this column (H) are given in the format yymmm where yy is the actual calendar <u>not</u> fiscal year and mmm is the month. The comment, ALREADY BY, means that the long term limit has already been exceeded. NONE LIKELY means that the trend line is below a high long term limit or above a low long term limit and diverging away from it. Predictions are not made further than 2 years out. If the trend line is expected to exceed the long term limit sometime after 2 years have elapsed, the comment AFTER 24 MO will be printed. The final column on this page specifies the long term limits.

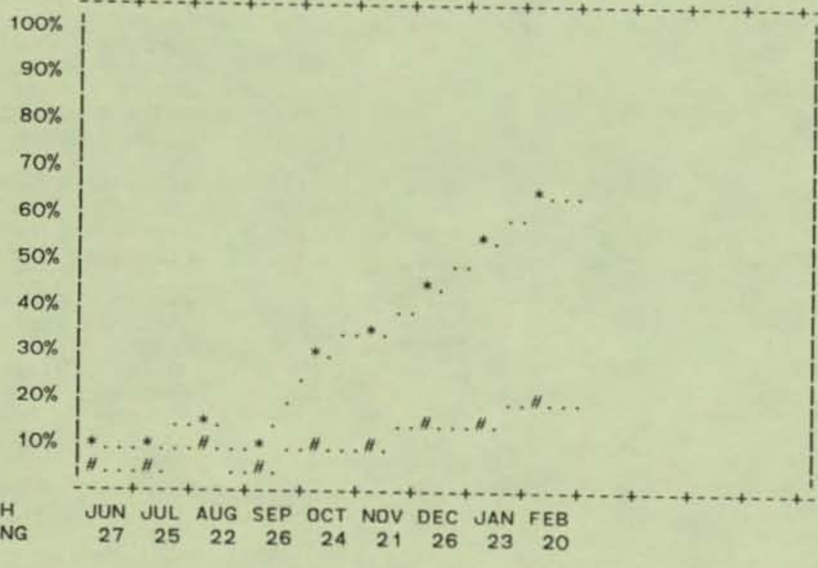
FROM: 22-FEB-81 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
MONTHLY TREND ANALYSIS REPORT

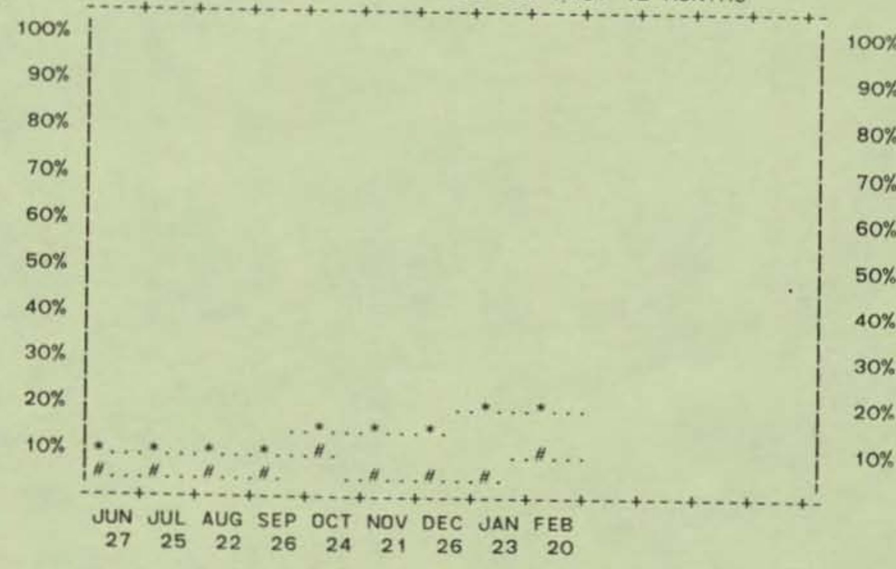
PUT ANY TITLE HERE
SYSTEM: PATH   PRIME TIME: 0800 - 1700

--------- CPU UTILIZATION (*) OVERHEAD (#) AND BOTH (@) ---------

PRIME TIME AVERAGE FOR PAST 12 MONTHS



NON-PRIME TIME AVERAGE FOR PAST 12 MONTHS



PRIME TIME AVERAGES OF KEY UTILIZATION ITEMS

B A

FISCAL MONTH ENDING ------- ITEM -------		Q4M3W5 JUN 27 M-08	Q1M1W4 JUL 25 M-07	Q1M2W4 AUG 22 M-06	Q1M3W5 SEP 26 M-05	Q2M1W4 OCT 24 M-04	Q2M2W4 NOV 21 M-03	Q2M3W5 DEC 26 M-02	Q3M1W4 JAN 23 M-01	Q3M2W4 FEB 20 M-00	LONG TERM LIMIT
% CPU	UTIL	7	8	12	7	27	33	42	51	60	
% IDLE	TIME	93	92	88	93	73	67	58	49	40	>70%
% OVHD	TIME	4	4	5	4	9	10	11	13	15	<30%
% LOST	TIME	0	0	0	0	0	0	1	1	1	>20%
USER	UUOS/SEC	21	29	44	28	79	136 •	182	427*	253*	>2%
CONTEXT	SWTS/SEC							61	65	83	>200
ACT SWAP	RATIO	.0	.0	.0	.0	.0	.1	.1	.1	.2	
ACTV JOB	% USR CR							9	12	18	>.5
ALL JOBS	% USR CR							61	74	100	
% RN JOB	IN MEM							100	100	100	
AVG JOB	SIZE	24	25	33	27	30	33	33	37	40	
# JOBS	LOGGED IN	23	21	28	24	32	34	39	45	59	
PGS USER	MEMORY	1757	1754	1745	1722	1742	1741	1445	1738	1736	
# JOBS	TTY IO Q	2	1	2	1	4	6	7	8	17*	<768
USER DSK	BLKS/SEC	14	15	19	5	29	41	49	62	72	>16
# JOBS	IN RUN Q	1	1	1	1	1	2	2	2	3	>100
SWAPPING	BLKS/SEC	0	0	0	0	2	4	13	15	33	>5
PK DSKEO	SWPS/SEC	0	0	0	0	2	4	13	15	33	>160
											>160

* = EXCEEDS LONG TERM LIMIT > = GREATER THAN OR EQUAL TO < = LESS THAN OR EQUAL TO

---------- CONTINUED NEXT PAGE ----------

Figure 1-6

FROM: 22-FEB-81 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
MONTHLY TREND ANALYSIS REPORT

PUT ANY TITLE HERE
SYSTEM: PATH   PRIME TIME: 0800 - 1700

PRIME TIME AVERAGES OF KEY UTILIZATION ITEMS

FISCAL MONTH ENDING ------- ITEM -------	Q4M3W5 JUN 27 M-08	Q1M1W4 JUL 25 M-07	Q1M2W4 AUG 22 M-06	Q1M3W5 SEP 26 M-05	Q2M1W4 OCT 24 M-04	Q2M2W4 NOV 21 M-03	Q2M3W5 DEC 26 M-02	Q3M1W4 JAN 23 M-01	Q3M2W4 FEB 20 M-00	LONG TERM LIMIT
PK PER10 SWPS/SEC		0								>160
AVG SCHD RSP TIME	40	1	1	0	2	3	5	6	14	>200
# MTAS ASSIGNED	0	0	0	0	0	1	0	0	0	
# LINES IN USE	18	16	21	20	24	28	29	23	36	
%TY CHNK IN USE							36	39	41	
SCN INTR RCVS/SEC							4	9	11	
SCN INTR XMTS/SEC							116	227	261	
% SYSTEM UPTIME	99.8	96.8	84.3	94.7	99.3	95.6	93.5	91.9	94.7	
% AMAR CLK TIME	91.8	95.8	82.0	94.4	98.4	91.2	88.9	84.3	93.2	
# SYSTEM RELOADS	2	5	12	1	3	18	18	22	6	

NON-PRIME TIME AVERAGES OF KEY UTILIZATION ITEMS

FISCAL MONTH ENDING ------- ITEM -------	Q4M3W5 JUN 27 M-08	Q1M1W4 JUL 25 M-07	Q1M2W4 AUG 22 M-06	Q1M3W5 SEP 26 M-05	Q2M1W4 OCT 24 M-04	Q2M2W4 NOV 21 M-03	Q2M3W5 DEC 26 M-02	Q3M1W4 JAN 23 M-01	Q3M2W4 FEB 20 M-00	LONG TERM LIMIT
% CPU UTIL	6	7	8	9	12	11	12	15	18	>70%
% IDLE TIME	94	93	92	91	88	89	88	85	82	<30%
% OVHD TIME	4	4	4	4	6	5	5	5	6	>20%
% LOST TIME	0	0	0	0	0	0	0	0	0	>2%
USER UUOS/SEC	20	22	32	43	47	48	45	232*	51	>200
CONTEXT SWTS/SEC							9	16	24	
ACT SWAP RATIO	.0	.0	.0	.0	.0	.0	.0	.0	.0	>.5
ACTV JOB % USR CR							2	3	4	
ALL JOBS % USR CR							31	29	33	
% RN JOB IN MEM							100	100	100	
AVG JOB SIZE	23	24	29	28	24	24	23	25	26	
# JOBS LOGGED IN	22	19	23	23	24	23	23	22	24	
PGS USER MEMORY	1757	1753	1746	1737	1742	1742	1475	1739	1735	<768
# JOBS TTY IO Q	2	1	1	0	1	1	1	1	1	>16
USER DSK BLKS/SEC	12	14	20	21	23	24	24	27	32	>100
# JOBS IN RUN Q	1	1	1	1	1	1	1	1	1	>5
SWAPPING BLKS/SEC	0	0	0	0	0	0	0	0	1	>160
PK DSKEO SWPS/SEC	0	0	0	0	0	0	0	0	1	>160
PK PER10 SWPS/SEC										>160
AVG SCHD RSP TIME	1	1	2	2	1	1	2	3	2	>200
# MTAS ASSIGNED	0	0	0	0	0	0	0	0	0	
# LINES IN USE	18	15	17	18	19	19	16	5	5	
%TY CHNK IN USE							34	36	36	
SCN INTR RCVS/SEC							0	0	1	
SCN INTR XMTS/SEC							6	12	20	
% SYSTEM UPTIME	97.6	93.6	93.1	81.2	96.3	73.0	86.6	93.1	83.9	
% AMAR CLK TIME	94.6	92.8	92.0	80.7	95.8	69.2	85.5	91.0	82.7	
# SYSTEM RELOADS	6	9	14	10	8	13	13	25	9	

* = EXCEEDS LONG TERM LIMIT   > = GREATER THAN OR EQUAL TO   < = LESS THAN OR EQUAL TO

---------- CONTINUED NEXT PAGE ----------

Figure 1-6 (continued)

FROM: 22-FEB-81 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
MONTHLY TREND ANALYSIS REPORT

PUT ANY TITLE HERE
SYSTEM: PATH   PRIME TIME: 0800 - 1700

PRIME TIME TREND OF KEY UTILIZATION ITEMS

------- ITEM -------		FIRST MONTHLY AVG	TABLE OF RELATIVE USAGE PER MONTH	LAST MONTHLY AVG	--09 MONTH- -- TREND -- GROWTH /MONTH	LINE FIT	PREDICTED RANGE OF VALUES WHERE AVERAGE WILL FALL IN 06 MONTHS C	12 MONTHS D	G PREDICTED PERIOD WHEN LONG TERM LIMIT REACHED	LONG TERM LIMIT
% CPU	UTIL	7	[2 2 3 2 8 10 13 15 18]	60	+7.14	92%	73-100		BY 82JUL	>70
% IDLE	TIME	93	[11 11 10 11 8 8 7 6 5]	40	-7.14	92%	0-26		BY 82JUL	<30
% OVHD	TIME	4	[4 4 5 4 8 9 11 12 15]	15	+1.46	92%	18-28		82MAR-82SEP	>20
% LOST	TIME	0	[0 0 1 1 4 8 19 16 25]	1	+0.15	85%	1-3		BY 82NOV	>2
USER	UUOS/SEC	21	[1 2 3 2 5 9 12 27 16]	253	+41.78	70%	239-863		BY 82JUN	>200
ACT SWAP	RATIO	.0	[3 3 3 3 5 8 13 14 20]	.2	+0.02	99%	.2-.3		83MAY-83JUN	>.5
AVG JOB	SIZE	24	[6 7 9 7 8 9 9 10 11]	40	+1.73	82%	39-58			
# JOBS	LOGGED IN	23	[6 5 7 6 8 8 9 11 15]	59	+4.15	85%	55-95			
PGS USER	MEMORY	1757	[9 9 9 8 9 9 7 8 8]	1736	ERRATIC VALUES					<768
# JOBS	TTY IO Q	2	[3 2 4 2 6 9 10 13 26]	17	+1.55	74%	10-31		BY 83FEB	>16
USER DSK	BLKS/SEC	14	[3 4 5 1 7 10 12 15 18]	72	+7.82	84%	72-151		BY 83JAN	>100
# JOBS	IN RUN Q	1	[6 6 6 6 7 9 10 11 14]	3	+0.18	86%	2-4		82SEP-83DEC	>5
SWAPPING	BLKS/SEC	0	[0 0 0 0 2 4 14 16 36]	33	+3.48	71%	17-68		84JAN-84FEB	>160
PK DSKEO	SWPS/SEC	0	[0 0 0 0 2 4 14 16 36]	33	+3.48	71%	17-68		84JAN-84FEB	>160
AVG SCHD	RSP TIME	40	[62 2 1 1 2 4 8 9 21]	14	ERRATIC VALUES				H	>200
# MTAS	ASSIGNED	0	[4 4 4 4 11 16 9 9 12] F	0	ERRATIC VALUES					
# LINES	IN USE	18	[6 6 7 7 8 10 10 8 12]	36	+1.87	71%	28-56 E			
% SYSTEM	UPTIME	99.8	[9 9 7 8 9 8 8 8 8]	94.7	ERRATIC VALUES					
% AMAR	CLK TIME	91.8	[8 9 7 9 9 8 8 8 9]	93.2	ERRATIC VALUES					
# SYSTEM	RELOADS	2	[2 4 10 1 3 16 16 19 5]	6	ERRATIC VALUES					

NON-PRIME TIME TREND OF KEY UTILIZATION ITEMS

------- ITEM -------		FIRST MONTHLY AVG	TABLE OF RELATIVE USAGE PER MONTH	LAST MONTHLY AVG	--09 MONTH- -- TREND -- GROWTH /MONTH	LINE FIT	PREDICTED RANGE OF VALUES WHERE AVERAGE WILL FALL IN 06 MONTHS	12 MONTHS	PREDICTED PERIOD WHEN LONG TERM LIMIT REACHED	LONG TERM LIMIT
% CPU	UTIL	6	[4 5 6 7 9 8 9 12 13]	18	+1.34	92%	19-29		AFTER 24 MO	>70
% IDLE	TIME	94	[9 9 9 9 8 8 8 8 8]	82	-1.34	92%	71-81		AFTER 24 MO	<30
% OVHD	TIME	4	[7 7 8 8 10 9 8 9 10]	6	ERRATIC VALUES					>20
% LOST	TIME	0	[3 3 3 3 6 6 11 6 22]	0	+0.01	99%	0-0		AFTER 24 MO	>2
USER	UUOS/SEC	20	[3 3 4 6 6 7 6 31 7]	51	ERRATIC VALUES					>200
ACT SWAP	RATIO	.0	[8 8 8 8 8 11 11 8 14]	.0	ERRATIC VALUES					>.5
AVG JOB	SIZE	23	[8 8 10 9 8 8 8 8 9]	26	ERRATIC VALUES					
# JOBS	LOGGED IN	22	[8 7 9 9 9 8 8 8 9]	24	ERRATIC VALUES					
PGS USER	MEMORY	1757	[9 9 9 8 9 9 7 8 8]	1735	ERRATIC VALUES					<768
# JOBS	TTY IO Q	2	[15 9 11 5 7 8 6 7 9]	1	ERRATIC VALUES					>16
USER DSK	BLKS/SEC	12	[5 5 7 8 9 9 9 10 12]	32	+2.14	94%	36-50		AFTER 24 MO	>100
# JOBS	IN RUN Q	1	[8 8 8 8 8 8 9 9 9]	1	+0.02	99%	1-1		AFTER 24 MO	>5

PREDICTIONS ARE ONLY MADE USING MONTHLY DATA.
GIVEN THE CURRENT TREND, THERE IS A 90% CHANCE THAT THE ACTUAL VALUES WILL FALL WITHIN THE PREDICTED RANGES.
UNCHANGING VALUES: REFERS TO RELATIVELY CONSTANT VALUES. ERRATIC VALUES: REFERS TO THE LACK OF A STRONG LINEAR PATTERN IN THE DATA.
* = EXCEEDS LONG TERM LIMIT > = GREATER THAN OR EQUAL TO < = LESS THAN OR EQUAL TO ---------- CONTINUED NEXT PAGE ----------

Figure 1-6 (continued)

FROM: 22-FEB-81 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
MONTHLY TREND ANALYSIS REPORT

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

NON-PRIME TIME TREND OF KEY UTILIZATION ITEMS

------- ITEM -------	FIRST MONTHLY AVG	TABLE OF RELATIVE USAGE PER MONTH	LAST MONTHLY AVG	--09 MONTH- -- TREND -- GROWTH LINE /MONTH	FIT	PREDICTED RANGE OF VALUES WHERE AVERAGE WILL FALL IN 06 MONTHS	12 MONTHS	PREDICTED PERIOD WHEN LONG TERM LIMIT REACHED	LONG TERM LIMIT
SWAPPING BLKS/SEC	0	[1 2 3 3 8 8 10 7 31]	1	ERRATIC VALUES					>160
PK DSKEO SWPS/SEC	0	[1 2 3 3 8 8 10 7 31]	1	ERRATIC VALUES					>160
AVG SCHD RSP TIME	1	[4 6 11 7 6 5 10 15 9]	2	ERRATIC VALUES					>200
# MTAS ASSIGNED	0	[5 5 8 8 14 9 10 10 9]	0	+0.01	99%	0-0			
# LINES IN USE	18	[10 8 10 11 11 11 9 3 3]	5	ERRATIC VALUES					
% SYSTEM UPTIME	97.6	[9 9 9 8 9 7 8 9 8]	83.9	ERRATIC VALUES					
% AMAR CLK TIME	94.6	[9 9 9 8 9 7 8 9 8]	82.7	ERRATIC VALUES					
# SYSTEM RELOADS	6	[4 6 10 7 6 9 9 18 6]	9	ERRATIC VALUES					

PREDICTIONS ARE ONLY MADE USING MONTHLY DATA.
GIVEN THE CURRENT TREND, THERE IS A 90% CHANCE THAT THE ACTUAL VALUES WILL FALL WITHIN THE PREDICTED RANGES.
UNCHANGING VALUES: REFERS TO RELATIVELY CONSTANT VALUES. ERRATIC VALUES: REFERS TO THE LACK OF A STRONG LINEAR PATTERN IN THE DATA.
* = EXCEEDS LONG TERM LIMIT > = GREATER THAN OR EQUAL TO < = LESS THAN OR EQUAL TO

Figure 1-6 (continued)

1.3.3 'Typical Day' Report

The 'Typical Day' Report can be generated weekly or monthly. The format of the report is identical for both the weekly and monthly versions, only the summary level of the data varies (weekly averages on the weekly report; monthly averages on the monthly report.) A monthly report is used as a sample here. 'Typical day' Reports are produced in pairs. The first report in the pair (shown in Figure 1-7) represents an average workday by combining data for Monday thru Friday, excluding holidays, of the week/month. The other report (not shown) combines data for Saturday, Sunday, and any holidays in the week/month. The method of reporting data in the 'Typical Day' Report corresponds most closely with the method of retaining and reporting on data in Workload AMAR. That is, all 24 hours in the day are represented and the breakdown is not by prime vs. non-prime time but by weekday vs. weekend and holiday.

The 'Typical Day' Reports are intended primarily to aid in load leveling machine usage. These reports show "typical" slack periods (where users could get more work done) and "typical" busy periods (where resources are at a premium and performance may be poor.) By using the 'Typical Day' Reports in conjunction with the corresponding workload reports, data center management can select the appropriate work to shift to less busy periods and thus make better use of system resources while improving overall thruput of the machine.

You should be aware, however, that the 'Typical Day' Reports intentionally smooth out the random day-to-day variations. They may also mask certain systematic variations such as a heavy processing day once a week (for example,the day before the payroll is due) or a heavy processing week once a month (for example, a monthly financial closing.) To identify patterns such as these, you should check the Weekly or Monthly Utilization Reports or the Weekly Trend Analysis Report.

The contents of the 'Typical Day' Report differs from that of the Daily System Utilization Report in the following areas (refer to the section on the Daily System Utilization Report for a description of the major report features and the format):

> The 'Typical Day' Report contains a header (A) surrounded by dashes which identifies the report and denotes whether it represents data for weekdays (B) or weekends and holidays.
>
> On the 'Typical Day' Report, there is no list of disk packs showing free space today, yesterday, and the difference; there is no list of periods of downtime; and the number of reloads is not given.
>
> Hourly values on the 'Typical Day' Report are computed by averaging the values of the corresponding time period from each of the days in the reporting interval. For example, on page 3 of the sample report, the value for #JOBS LOGGED IN (C) for 10:00 AM is computed by averaging the 10:00 AM values for each Monday

through Friday (minus holidays) in the month.

NO. OF HOURS KEY ITEMS OVER LIMITS (D) on page 1, # ITEMS OVER LIMITS (E) on page 3, and #HOURS OVER LIMITS (F) also on page 3 refer to the total for the "typical" day not the total during the period measured.

Long term (G) rather than short term thresholds are used.

- AMAR -
MONTHLY 'TYPICAL DAY' SUMMARY REPORT

FROM: 24-JAN-82 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

PAGE: 1

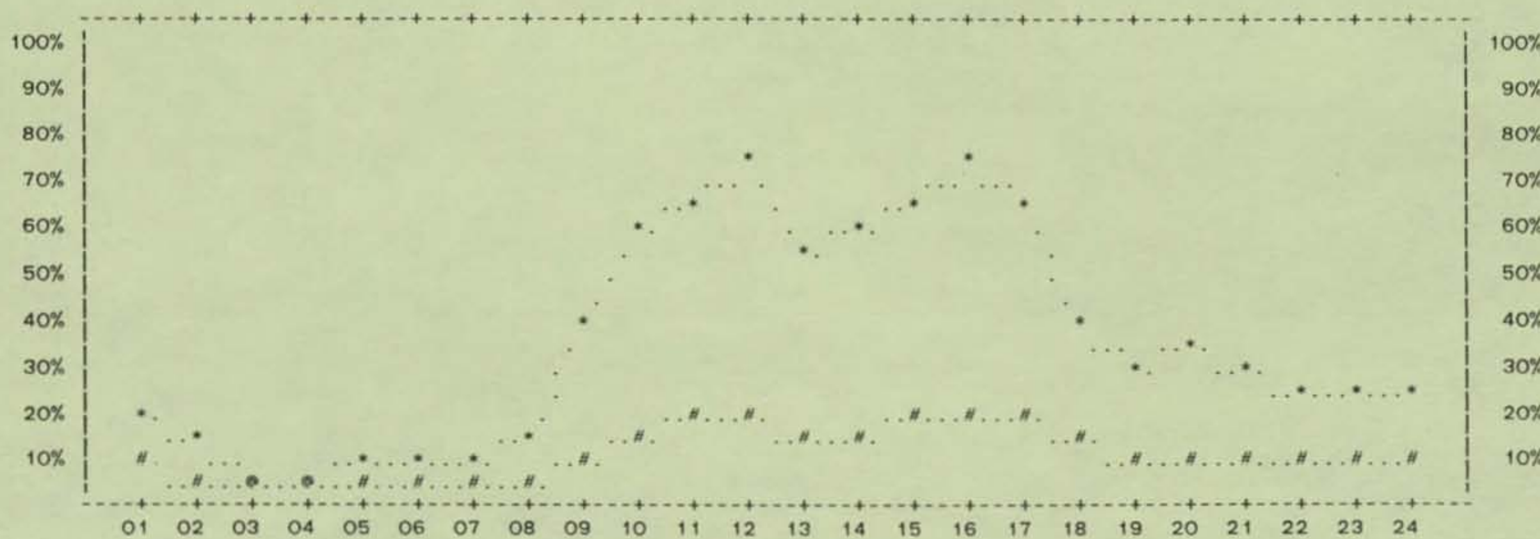
PUT ANY TITLE HERE
SYSTEM: PATH   PRIME TIME: 0800 - 1700

A

THIS REPORT REPRESENTS AN AVERAGE DAY
OBTAINED BY COMBINING DATA FOR
MONDAY THRU FRIDAY EXCEPT HOLIDAYS

B

--------- CPU UTILIZATION (*) OVERHEAD (#) AND BOTH (@) ---------



-------------------- TYPICAL HOUR -----------------------

------------------- SUMMARY OF KEY UTILIZATION ITEMS ----------------

D

-------AVERAGE-----	# JOBS LOGGED IN	# LINES IN USE	% IDLE TIME	% LOST TIME	% OVHD TIME	ACT SWAP RATIO	SWAPPING BLKS/SEC	USER DSK BLKS/SEC	# MTAS ASSIGNED	PGS USER MEMORY	NO. OF HOURS KEY ITEMS OVER LIMITS
----PRIME TIME-----	59	36	40	1	15	.2	33	72	0	1736	4
---NON-PRIME TIME--	24	5	83	0	6	.0	1	27	0	1735	

Figure 1-7

```
                                              - AMAR -
FROM: 24-JAN-82 (SUNDAY)            MONTHLY 'TYPICAL DAY' PROBLEM REPORT                              PAGE:  2
THRU: 20-FEB-82 (SATURDAY)

                                              PUT ANY TITLE HERE
                                     SYSTEM: PATH     PRIME TIME: 0800 - 1700


           PROBLEM PERIODS                                            PROBLEM RESOURCES

EACH * = 1 KEY ITEM OVER WATCHDOG LIMIT           EACH P = 1 PRIME HOUR WHEN THE ITEM WAS OVER THE WATCHDOG LIMIT
EACH + = 1 OTHER ITEM OVER LIMIT                  EACH N = 1 NON-PRIME HOUR WHEN THE ITEM WAS OVER LIMIT

(SEE THE FIRST PAGE OF DETAIL REPORT)
(FOR HOURLY AVERAGES OF KEY ITEMS)

 -----HOUR---- ------NO. OF ITEMS------ --------ITEM-------- ----NO. OF HOURS-------- ------------- COMMENTS --------------------

 00:00 - 01:00                           % IDLE TIME          PP                       SERIOUS   CPU PRESSED: CHK WORKLD DATA FIRST
 01:00 - 02:00                           % LOST TIME          PP                       SERIOUS   MEMORY SHORTAGE OR SWAP DEVICE SLOW
 02:00 - 03:00                           USER UUOS/SEC        PPPPPPPP                 CRITICAL  USER PGM PROBLEM: CHK WORKLD DATA
 03:00 - 04:00                           # JOBS BLK IO Q      PPP                      SERIOUS   BACKUP OF I/O JOBS: CHK CONTENTION
 04:00 - 05:00                           # JOBS TTY IO Q      PPPPPP                   CRITICAL  INTERACTIVE USE HIGH: CHK RESPONSE
 05:00 - 06:00
 06:00 - 07:00
 07:00 - 08:00
 08:00 - 09:00
 09:00 - 10:00 ++
 10:00 - 11:00 ++
 11:00 - 12:00 *+++
 12:00 - 13:00 +
 13:00 - 14:00 ++
 14:00 - 15:00 *++
 15:00 - 16:00 **+++
 16:00 - 17:00 ++
 17:00 - 18:00
 18:00 - 19:00
 19:00 - 20:00
 20:00 - 21:00
 21:00 - 22:00
 22:00 - 23:00
 23:00 - 24:00

                                     SYSTEM AVAILABILITY SUMMARY

                      SYSTEM    % TIME
                      UPTIME    MEASURED

  PRIME TIME     :     94.7%     93.2%
  NON-PRIME TIME:      89.7%     87.7%
```

Figure 1-7 (continued)

FROM: 24-JAN-82 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
MONTHLY 'TYPICAL DAY' DETAIL REPORT

PAGE: 3

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

E

	------------------------------				KEY UTILIZATION ITEMS				------------------------------		# ITEMS OVER LIMITS	
PERIOD	# JOBS LOGGED IN	# LINES IN USE	% IDLE TIME	% LOST TIME	% OVHD TIME	ACT SWAP RATIO	SWAPPING BLKS/SEC	USER DSK BLKS/SEC	# MTAS ASSIGNED	PGS USER MEMORY	THIS PAGE	ALL PAGES
00:00 - 01:00	23	5	84	0	6	.0	0	18	0	1735		
01:00 - 02:00	23	4	89	0	5	.0	0	11	0	1735		
02:00 - 03:00	22	4	96	0	3	.0	0	1	0	1735		
03:00 - 04:00	22	4	96	0	3	.0	0	0	0	1735		
04:00 - 05:00	22	4	93	0	4	.0	0	4	0	1735		
05:00 - 06:00	22	4	94	0	3	.0	0	0	0	1735		
06:00 - 07:00	22	4	92	0	4	.0	0	1	0	1735		
07:00 - 08:00	24	6	89	0	5	.0	2	11	0	1735		
08:00 - 09:00	40 C	20	64	1	10	.1	12	45	0	1735		
09:00 - 10:00	58	36	41	1	15	.2	31	95	0	1735		2
10:00 - 11:00	64	40	35	1	17	.2	36	92	1	1735		2
11:00 - 12:00	64	39	28 *	1	17	.2	38	84	1	1735	1	4
12:00 - 13:00	60	35	46	1	14	.2	11	56	0	1735		1
13:00 - 14:00	62	38	43	1	15	.2	29	57	0	1736		2
14:00 - 15:00	65	40	35	2 *	17	.2	43	69	0	1736	1	3
15:00 - 16:00	67	41	29 *	2 *	18	.2	62	72	1	1736	2	5
16:00 - 17:00	57	31	37	1	16	.2	38	78	0	1736		2
17:00 - 18:00	36	13	60	0	11	.1	3	70	0	1735		
18:00 - 19:00	29	8	71	0	8	.1	3	57	0	1735		
19:00 - 20:00	27	6	66	0	8	.1	1	75	0	1735		
20:00 - 21:00	25	5	70	0	8	.1	1	62	0	1735		
21:00 - 22:00	24	5	77	0	6	.1	2	38	0	1735		
22:00 - 23:00	24	5	75	0	7	.1	1	58	0	1735		
23:00 - 24:00	23	5	75	0	7	.1	2	30	0	1735		
----PRIME TIME-----												
AVERAGE VALUE:	59	36	40	1	15	.2	33	72	0	1736		
LONG TERM LIMIT:	NONE	NONE	<30%	>2%	>20%	>.5	>160	>100	NONE	<768		
% TIME OVER LIMIT:			38.8%	17.0%	16.7%			23.7%				
# HOURS OVER LIMIT: F			2	2							4	21
--NON-PRIME TIME---												
AVERAGE VALUE:	24	5	83	0	6	.0	1	27	0	1735		
LONG TERM LIMIT: G	NONE	NONE	<30%	>2%	>20%	>.5	>160	>100	NONE	<768		
% TIME OVER LIMIT:												
# HOURS OVER LIMIT:												

* = OVER LIMITS   > = GREATER THAN OR EQUAL TO   < = LESS THAN OR EQUAL TO   ----------

CONTINUED NEXT PAGE --------

Figure 1-7 (continued)

FROM: 24-JAN-82 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
MONTHLY 'TYPICAL DAY' DETAIL REPORT

PAGE: 4

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

PERIOD	USER UUOS/SEC	# JOBS BLK IO Q	%PK DEVPO FREE SPC	# JOBS IN RUN Q	% SWAP SPC LEFT	# JOBS TTY IO Q	# ITEMS OVER LIMITS THIS PAGE	# ITEMS OVER LIMITS ALL PAGES
	----OTHER UTILIZATION ITEMS							
00:00 - 01:00	43	.2	29	1	82	1		
01:00 - 02:00	27	.1	28	1	82	1		
02:00 - 03:00	7	.0	28	1	83	0		
03:00 - 04:00	7	.0	28	1	84	1		
04:00 - 05:00	14	.1	28	1	83	1		
05:00 - 06:00	7	.0	28	1	83	1		
06:00 - 07:00	10	.0	29	1	83	1		
07:00 - 08:00	44	.2	29	1	81	2		
08:00 - 09:00	154	.8	29	2	63	9		
09:00 - 10:00	229 *	1.3	28	2	43	16 *	2	2
10:00 - 11:00	259 *	1.4	27	3	37	18 *	2	2
11:00 - 12:00	277 *	1.5 *	26	3	36	18 *	3	4
12:00 - 13:00	241 *	.9	25	2	37	15	1	1
13:00 - 14:00	293 *	.9	24	2	35	18 *	2	2
14:00 - 15:00	303 *	1.4	23	3	31	19 *	2	3
15:00 - 16:00	284 *	1.5 *	23	3	31	20 *	3	5
16:00 - 17:00	244 *	1.5 *	25	3	38	15	2	2
17:00 - 18:00	136	1.0	25	2	60	5		
18:00 - 19:00	81	.5	24	1	69	2		
19:00 - 20:00	79	.6	24	1	72	1		
20:00 - 21:00	91	.5	25	1	75	1		
21:00 - 22:00	64	.3	27	1	78	1		
22:00 - 23:00	91	.4	26	1	79	1		
23:00 - 24:00	89	.3	25	1	80	1		
----PRIME TIME-----								
AVERAGE VALUE:	253	1.3	26	3	39	17		
LONG TERM LIMIT:	>200	>1.5	<10%	>5	<30	>16		
% TIME OVER LIMIT:	53.0%	33.4%	26.3%	13.1%	30.9%	60.7%		
# HOURS OVER LIMIT:	8	3				6	17	21
--NON-PRIME TIME---								
AVERAGE VALUE:	50	.3	27	1	79	1		
LONG TERM LIMIT:	>200	>1.5	<10%	>5	<30	>16		
% TIME OVER LIMIT:			21.9%					
# HOURS OVER LIMIT:								

* = OVER LIMITS > = GREATER THAN OR EQUAL TO < = LESS THAN OR EQUAL TO

Figure 1-7 (continued)

1.3.4 Disk Reports

Disk Reports can be generated daily, weekly, or monthly. The formats are identical, only the length of report period varies. A Monthly Disk Report is used as a sample here. Refer to Figure 1-8. The Disk Report provides the following features:

- All disk related information is presented in one place for ease of analysis.
- System wide information such as system uptime, swapping rate, channel usage, etc., is broken out from individual disk pack information such as mount time, blocks transferred per second, etc.
- Data is organized by logical pack name for ease of reference.
- Mount time and in use time are provided and are expressed both in hours and minutes and as percentages of AMAR measured time.

The Disk Report is separated into three sections - a General Usage Summary and Prime/Non-prime Time Pack Summaries. The format for the report header is the same as that of the Trend Analysis Report. Please refer to that section for a description. The General Usage Summary follows the report header and is located in the upper left hand corner of the first page. The contents of the General Usage Summary are fixed. That is, it always contains the following information:

- The number of hours (A) theoretically available in the reporting period (prime time followed by non-prime time) assuming the system was up 24 hours a day.
- The number of hours and minutes the system was detected by AMAR as being up (B). This time may not be 100% accurate if the system came up and then went down again before AMAR was restarted. This time will always represent the minimum uptime possible, i.e., the actual uptime may be greater than that shown here.
- The number of hours and minutes AMAR measured the system (C). This time should always be accurate.
- The total number of blocks swapped per second system-wide (D).
- The average percent of swap space left (E) during the reporting period.
- The average busy time for each channel (F) expressed as percentages. (KL10 only.)
- The average busy time for each priority interrupt level (G) expressed as percentages. (KL10 only.)

The average channel transfer wait queue length (H) for each disk channel on the system.

The Prime Time Pack Summary follows the General Usage Summary. The first column (I) gives the logical pack name. Packs are listed in alphabetical order, one line per pack.

Following the pack name is a pair of columns, TOTAL TIME (HH:MM) MOUNTED-IN USE, which show the total time, in hours and minutes, that the pack was mounted (J) and "in use" (K). A pack is considered "in use" during a minute if at least one block is read from it or written to it during that minute. The disk report will also include physical unit information if it is collected and specified in the RFD file. The next pair of columns, % OF TIME MOUNTED-IN USE, expresses mounted time (L) and "in use" time (M) as a percentage of the time AMAR measured the system not the system uptime. The values in these two pairs of columns may be anywhere from 1% to 3% low because they are based on an assumed sample count of 60 samples per hour. In actual practice, slightly fewer samples may occasionally be taken , especially if the machine is very heavily used.

The single column, % MOUNTED TIME IN USE (N), gives the percent of mounted time that the pack was actually being used. It is derived by dividing the first "in use" time (K) by the mount time (J). This is a better indication of how heavily a pack is being used than the "in use" percentage (M) which is based on the whole period of measured time regardless of the mount status of the pack. If a pack is mounted a large percentage of the time and in use a relatively small percentage, it is important to know whether that usage occurred in one chunk or was scattered throughout the reporting interval before attempting to free up the spindle. This can be determined by examining the hourly averages for the period in question with the AMARON Online Inquiry Program.

The remaining columns (P) in the Prime Time Pack Summary are items directly measured by AMAR and selected for display on this report via the xxxxDR.RFD file. For a definition of these items please refer to the Appendix called "System AMAR Item Definitions".

The Non-prime Time Pack Summary follows the Prime Time Pack Summary and is identical in format to it.

FROM: 24-JAN-82 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
MONTHLY DISK REPORT

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

GENERAL USAGE SUMMARY

	PRIME TIME	NON-PRIME TIME	
HOURS THEORETICALLY AVAILABLE	180:00	492:00	A
HH:MM SYSTEM WAS UP	170:27	412:59	B
HH:MM AMAR MEASURED THE SYSTEM	167:40	406:59	C
SWAPPING BLKS/SEC	33	1	D
% SWAP SPC LEFT	39	79	E
CHAN 0 % TIM BUSY	1.67%	.22%	
CHAN 1 % TIM BUSY	.22%	.08%	
CHAN 2 % TIM BUSY	2.04%	.46%	
CHAN 3 % TIM BUSY	.44%	.33%	
CHAN 4 % TIM BUSY	.15%	.10%	F
CHAN 5 % TIM BUSY	.00%	.00%	
CHAN 6 % TIM BUSY	.00%	.00%	
CHAN 7 % TIM BUSY	.00%	.00%	
PI 0 % TIM BUSY	.15%	.01%	
PI 1 % TIM BUSY	.00%	.00%	
PI 2 % TIM BUSY	.15%	.14%	
PI 3 % TIM BUSY	.82%	.13%	G
PI 4 % TIM BUSY	1.84%	.47%	
PI 5 % TIM BUSY	.15%	.01%	
PI 6 % TIM BUSY	.00%	.00%	
PI 7 % TIM BUSY	6.69%	3.74%	
CHAN 00 WAIT Q	.0	.0	
CHAN 01 WAIT Q	.0	.0	
CHAN 02 WAIT Q	.0	.0	H
CHAN 04 WAIT Q	.0	.0	

Figure 1-8

PRIME TIME PACK SUMMARY

PACK NAME (I)	TOTAL TIME(HH:MM) MOUNTED (J)	- IN USE (K)	% OF TIME MOUNTED (L)	- IN USE (M)	% MOUNTED TIME IN USE (N)	PACK FREE SPC (P)	PACK BLKS/SEC	PACK SWPS/SEC	PACK WAIT Q
CASEO	166:47	155:26	99%	93%	93%	31%	21		.1
COREO	166:47	79:20	99%	47%	48%	45%	3		.0
DEVPO	166:47	88:44	99%	53%	53%	26%	12		.0
DSKEO	166:47	166:47	99%	99%	100%	33%	3	33	.0
DSKTO	44:46	5:09	27%	3%	12%	100%	0		.0
DSKWO	166:47	135:42	99%	81%	81%	77%	5		.0
HRS20	36:54	6:26	22%	4%	17%	13%	5		.0
PAWSO	65:16	13:36	39%	8%	21%	89%	3		.0
PENSO	120:12	8:06	72%	5%	7%	13%	0		.0
PER10	30:57	6:49	18%	4%	22%	59%	6		.0
PER20	14:16	5:31	9%	3%	39%	26%	18		.0
PER60	144:04	16:05	86%	10%	11%	23%	1		.0
PLT10	165:13	79:15	99%	47%	48%	42%	3		.0
STARO	160:03	79:00	95%	47%	49%	42%	3		.0
TRNGO	57:23	3:45	34%	2%	7%	74%	0		.0
TST10	11:59	0:30	7%	0%	4%	68%	4		.0

---------- CONTINUED NEXT PAGE ----------

FROM: 24-JAN-82 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
MONTHLY DISK REPORT

PUT ANY TITLE HERE
SYSTEM: PATH   PRIME TIME: 0800 - 1700

PRIME TIME PACK SUMMARY

PACK NAME	TOTAL TIME(HH:MM) MOUNTED - IN USE	% OF TIME MOUNTED - IN USE	% MOUNTED TIME IN USE	PACK FREE SPC	PACK BLKS/SEC	PACK SWPS/SEC	PACK WAIT Q
TST20	8:36 - 2:18	5% - 1%	27%	38%	29		.0
TST30	5:22 - 0:07	3% - 0%	2%	71%	0		.0
TST40	7:25 - 0:14	4% - 0%	3%	98%	0		.0
USRE0	64:29 - 7:37	38% - 5%	12%	99%	0		.0
USRS0	166:47 - 165:59	99% - 99%	100%	23%	15		.1
YEAR0	18:05 - 0:51	11% - 1%	5%	21%	0		.0

NON-PRIME TIME PACK SUMMARY

PACK NAME	TOTAL TIME(HH:MM) MOUNTED - IN USE	% OF TIME MOUNTED - IN USE	% MOUNTED TIME IN USE	PACK FREE SPC	PACK BLKS/SEC	PACK SWPS/SEC	PACK WAIT Q
BKP20	23:18 - 12:18	6% - 3%	53%	52%	14		.0
BKP21	23:18 - 13:12	6% - 3%	57%	52%	14		.0
BKP22	23:18 - 13:29	6% - 3%	58%	53%	14		.0
CASE0	405:42 - 59:07	100% - 15%	15%	37%	5		.0
CORE0	405:42 - 10:47	100% - 3%	3%	44%	0		.0
DEVP0	405:42 - 36:09	100% - 9%	9%	25%	3		.0
DSKE0	405:42 - 405:41	100% - 100%	100%	32%	1	1	.0
DSKT0	106:27 - 4:31	26% - 1%	4%	100%	0		.0
DSKW0	405:42 - 101:02	100% - 25%	25%	77%	4		.0
HRS10	20:03 - 0:31	5% - 0%	3%	87%	3		.0
HRS20	74:38 - 1:25	18% - 0%	2%	21%	1		.0
KLAD0	1:52 - 0:17	0% - 0%	15%	39%	0		.0
PAWS0	166:39 - 3:47	41% - 1%	2%	80%	1		.0
PENS0	255:16 - 4:15	63% - 1%	2%	13%	2		.0
PER10	112:40 - 19:25	28% - 5%	17%	65%	5		.0
PER20	84:51 - 13:19	21% - 3%	16%	30%	8		.0
PER30	1:34 - 0:57	0% - 0%	61%	71%	61		.0
PER60	218:57 - 3:37	54% - 1%	2%	23%	1		.0
PER70	1:21 - 0:32	0% - 0%	40%	84%	38		.0
PERS0	5:31 - 1:24	1% - 0%	25%	35%	40		.0
PLT10	323:10 - 7:35	79% - 2%	2%	42%	0		.0
STAR0	261:00 - 7:04	64% - 2%	3%	42%	1		.0
TRNG0	82:30 - 0:46	20% - 0%	1%	74%	1		.0
TST10	0:19 - 0:01	0% - 0%	5%	38%	0		.0
TST20	0:19 - 0:00	0% - 0%	0%	15%	0		.0
TST30	17:01 - 0:08	4% - 0%	1%	97%	0		.0
TST40	34:01 - 0:10	8% - 0%	0%	98%	0		.0
USRE0	106:54 - 2:40	26% - 1%	2%	99%	0		.0
USRS0	405:41 - 211:43	100% - 52%	52%	23%	9		.0
YEAR0	10:31 - 0:00	3% - 0%	0%	21%	0		.0

Figure 1-8 (continued)

1.3.5 Tape Reports

Tape Reports can be generated daily, weekly, or monthly. The formats are identical, only the length of the report period varies. A Monthly Tape Report is used as a sample here. Refer to Figure 1-9. The Tape Report provides the following features:

All tape related information is presented in one place for ease of analysis.

Data is organized sequentially by tape drive number for ease of reference.

Mount time and in use time for individual drives are provided and expressed both in hours and minutes and as percentages of AMAR measured time.

The Tape Report is similar to the Disk Report. The Tape Report is separated into three sections - a General Usage Summary and Prime/Non-prime Time Tape Summaries. The format of the report header is identical to that of the Trend Analysis Report. Please refer to that section for its description. The General Usage Summary follows the report header and is located in the upper left hand corner of the first page. The General Usage Summary always contains the following information:

The number of hours (A) theoretically available in the report period (prime time followed by non-prime time), assuming the system was up 24 hours a day.

The number of hours and minutes AMAR detected the system as being up (B). This time may not be 100% accurate if the system came up and then went down again before AMAR was restarted. This time will always represent the minimum uptime possible, i.e., the actual uptime may be greater than that shown here.

The number of hours and minutes AMAR measured the system (C). This time should always be accurate.

The average number of tape drives assigned (D) during the reporting period.

After the General Usage Summary comes the Prime Time Tape Summary. The first column (E) gives the drive number. Drives are listed sequentially, one line per drive. Following the drive number is a pair of columns, TOTAL TIME (HH:MM) ASSIGNED-IN USE, which shows, in hours and minutes, the total time that tapes were assigned to the drive (F) and actually were "in use" (G) on the drive. A tape drive is considered "in use" during a minute if at least one frame of data is read from it or written to it during that minute. The next pair of columns, % OF TIME ASSIGNED-IN USE, expresses assigned time (H) and "in use" time (I) as a percentage of the time AMAR measured the system <u>not</u> the system uptime. The values in these two pairs of columns may be anywhere from 1% to 3% low because, as in the Disk Report, they are based on an assumed sample count of 60 samples per hour. In actual practice, slightly fewer samples may occasionally be taken, especially if the system is very heavily used.

The column, % ASSIGNED TIME IN USE (J), gives the percent of assigned time that a tape was actually being used. It is derived by dividing the first "in use" time (G) by the assigned time (F). This percentage can quickly indicate how much of the time drives may have been left assigned unnecessarily.

The final column, TAPE BLKS/SEC (K), gives the transfer rate on each drive expressed in equivalent disk blocks (128 words per block) per second. This block size is independent of any actual physical block size on the tape. The data is expressed in equivalent disk blocks to make it easier to relate tape data to workload activity, for example, BACKUP usage where only the disk blocks read/written are given in the workload data.

The set of three columns which are underneath those just described provide information on simultaneous tape usage. The data can be used to determine whether or not more drives are needed or if any excess drives can be eliminated. The first column (L) gives the possible number of drives which could be assigned at one time. This number will vary from 0 to the maximum mumber on the system. The next column (M) gives the percentage when exactly 0, 1, 2, etc. drives were assigned. The last column (N) gives the cumulative percentage of assigned drives, i.e., the percent of time when at most 0, 1, 2, etc. drives were assigned. In this sample, it can be seen that during prime time there were never more than three drives in use at any one time (P). Two drives were always free.

In data centers where tape drives are shared between systems via dual porting, you should check both sets of Tape Reports in order to correctly analyze drive usage.

FROM: 24-JAN-82 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
MONTHLY TAPE REPORT

PAGE: 1

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

GENERAL USAGE SUMMARY

	PRIME TIME	NON-PRIME TIME	
HOURS THEORETICALLY AVAILABLE	180:00	492:00	A
HH:MM SYSTEM WAS UP	170:27	412:59	B
HH:MM AMAR MEASURED THE SYSTEM	167:40	406:59	C
# MTAS ASSIGNED	0	0	D

PRIME TIME TAPE SUMMARY

TAPE DRIVE	TOTAL TIME(HH:MM) ASSIGNED	- IN USE	% OF TIME ASSIGNED	- IN USE	% ASSIGNED TIME IN USE	TAPE BLKS/SEC
E	F	G	H	I	J	K
MTA430	4:09 -	1:14	2% -	1%	30%	62
MTA431	8:55 -	1:18	5% -	1%	15%	28
MTA432	12:52 -	3:50	8% -	2%	30%	29
MTA433	34:53 -	11:17	21% -	7%	32%	20
MTA434	10:20 -	5:16	6% -	3%	51%	40

NUMBER OF DRIVES ASSIGNED SIMULTANEOUSLY	% OF TIME WHEN EXACTLY THIS NO. WERE ASSIGNED	% OF TIME WHEN AT MOST THIS NO. WERE ASSIGNED
L	M	N
0	65%	65%
1	29%	94%
2	5%	99%
3	1% P	100%
4	0%	100%
5	0%	100%

NON-PRIME TIME TAPE SUMMARY

TAPE DRIVE	TOTAL TIME(HH:MM) ASSIGNED	- IN USE	% OF TIME ASSIGNED	- IN USE	% ASSIGNED TIME IN USE	TAPE BLKS/SEC
MTA430	2:28 -	2:16	1% -	1%	92%	80
MTA431	10:53 -	5:35	3% -	1%	51%	41
MTA432	25:30 -	21:21	6% -	5%	84%	87
MTA433	34:13 -	21:09	8% -	5%	62%	55
MTA434	12:01 -	5:57	3% -	1%	50%	65

---------- CONTINUED NEXT PAGE ----------

Figure 1-9

FROM: 24-JAN-82 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
MONTHLY TAPE REPORT

PAGE: 2

PUT ANY TITLE HERE
SYSTEM: PATH   PRIME TIME: 0800 - 1700

NUMBER OF DRIVES ASSIGNED SIMULTANEOUSLY	% OF TIME WHEN EXACTLY THIS NO. WERE ASSIGNED	% OF TIME WHEN AT MOST THIS NO. WERE ASSIGNED
0	83%	83%
1	15%	97%
2	2%	99%
3	1%	100%
4	0%	100%
5	0%	100%

Figure 1-9 (continued)

blank page

1.3.6 Online Inquiry Reports

The AMARON online inquiry program produces two types of reports - Tables of Average Values and Histograms - which can either be displayed at the terminal or stored in a file for later printing or for processing by user programs.

1.3.6.1 Table Of Average Values -

Refer to the report sample in Figure 1-10. For a detailed explanation of the report dialogue shown in the sample, refer to the Appendix called "Online Inquiry Program (AMARON) Dialogue".

The format of the Table of Average Values is as follows:

Line 1 - The first line in the report header contains the 4 character system code (A) and the starting (B) and ending (C) dates of the report period specified by you.

Line 2 - The second line of the report header identifies the date/time column and then gives the 4 character item (D) code or 7 character subitem code positioned over its associated data column. Refer to the Appendix called "System AMAR Item Definitions" for a list of valid items and their codes. Items and subitems are listed in the order specified by you. From 1 to 10 items may be specified on a single report.

Lines 3 - on - Contain:

The ending date (E) of the fiscal period reported in the format yymmdd (yy = normal calendar <u>not</u> fiscal year; mm = month; dd = day);

For hourly level data, the hour (F) in the format hhss (hh = hours; ss = minutes); and

The average value (G) for each item/subitem specified. Note that a value of -1 indicates that no data was gathered during that interval. For example, in the case of disk packs, the pack may not have been mounted.

Lines 1 and 2 (the report header) may be automatically deleted from the report if it is stored in a file rather than printed at the terminal. If the headers are deleted, you must be careful to retain a description of the report contents since there will be no internal identification.

```
.RUN AMARON[,,AMAR]

DATABASE NAME: AMAR.DB
Histogram Function (Y/N): N
Output at (T)erminal or in (F)ile: T
Start Date: 820215
End Date: 820215
Granularity Level: H
Starting Hour: 07
Ending Hour: 17
(P)rime (N)on Prime (B)oth or (W)eekend: B
Item 1: TTYU
Item 2: TIOW
Item 3: CPUC
Item 4: CPIC
Item 5: LUIC?
LUIC          LUIOBKP10   LUIOBKP11   LUIOBKP12   LUIOBKP20   LUIOBKP21
LUIOBKP22     LUIOCASEC   LUIOCCREC   LUIODEVPC   LUIODSKEC   LUIODSKRC
LUIODSKTC     LUIODSKWC   LUIOFOO6C   LUIOFOO7C   LUIOHRS1C   LUIOHRS2C
LUIOKLADC     LUIOPAWSC   LUIOPENSC   LUIOPER1C   LUIOPER2C   LUIOPER3C
LUIOPER6C     LUIOPER7C   LUIOPERSC   LUIOPLT1C   LUIOPLT2C   LUIORP22C
LUIORP23C     LUIOSTARC   LUIOTRNGC   LUIOTST1C   LUIOTST2C   LUIOTST3C
LUIOTST4C     LUIOUSREC   LUIOUSRSC   LUIOYEARC
Item 5: LUIODSKEC
Item 6: LUIODSKRC
Item 7:

                                    A                                   B          C
                                SYSTEM PATH                        FROM 820215 TO 820215
DATE   TIME             TTYU        TIOW         CPUC          CPICLUIODSKEC  LUIODSKRC
 E      F       D
820215:0700             4.00         .00         9.52         96.33         .08        -1.00
820215:0800             5.02         .72        48.84         89.77        1.13        -1.00
820215:0900            19.08        8.16       144.30         38.97        7.71        -1.00
820215:1000            31.95       17.22       140.67         38.17        1.22        -1.00
820215:1100            36.63       20.32       201.19         10.05        1.81        -1.00
820215:1200     G      32.22       12.66       172.91         35.44        1.52        -1.00
820215:1300            34.27       15.54       204.52         61.15        3.74        -1.00
820215:1400            37.78       19.92       597.12         47.59        1.35        -1.00
820215:1500            38.52       19.55       573.49         37.87        2.95        -1.00
820215:1600            46.29       26.95       470.13         18.84        2.64        -1.00
820215:1700            34.54       16.65       357.76         22.02        2.56        -1.00
More Requests (Y/N): N

EXIT
```

Figure 1-10

1.3.6.2 Histograms -

Refer to the report sample in Figure 1-11. For a detailed explanation of the report dialogue shown in the sample, refer to the Appendix called "Online Inquiry Program (AMARON) Dialogue".

The format of the histogram report is as follows:

> Line 1 - The first line in the report header contains the 4 character system code (A) and the 4 character item or 7 character subitem code (B). Refer to the Appendix called "System AMAR Item Definitions" for a list of valid items and their codes. From 1 to 10 items/subitems may be selected in a single report. The histograms for each item/subitem are given in the same order as that specified in the dialogue. Histograms are listed chronologically according to fiscal period specified. That is, if 2 days worth of data for 4 items is requested, all of day 1's data will be given first followed by day 2's data.
>
> Line 2 - Contains the end date (C) of the fiscal period represented by the data and the starting (D) and ending (E) dates of the report period specified by you. All dates are displayed in the format mm/dd/yy (mm = month; dd = day; yy = normal calendar <u>not</u> fiscal year).
>
> Line 3 - Denotes whether the data represents prime or non-prime time (F) and, for hourly data, the hour (G) represented.
>
> Lines 4 - 5 - The last lines of the report header contain the titles for each column.

The first column (H) gives the possible range of values for the item or subitem. The width of the value ranges (normally called classes) are defined in the system AMAR database at installation time. The class width for each item and subitem can be obtained by running the AMRGEN program using the LIST CLASS command. Zero (0) values are always treated as a separate class. This is different from ranges where no sample values fall. Such ranges are called blank ranges (I). Printing of blank ranges may be suppressed by you.

The range of values in the sample report represents user UUO's per second. Again refer to Appendix B for a definition of each item and its data type, i.e., percentage, blocks per second, etc.

The second column (J) lists the number of samples whose values fell within the range.

The third column (K), FREQUENCY % OF TOTAL, gives the percent of samples whose values fell within the range. Since a sample is typically a minute in length, this can be thought of as the percent of time that the values fell within the range.

```
.run amaron[,,amar]

DATABASE NAME: amar
Histogram Function (Y/N): y
Output at (T)erminal or in (F)ile: t
Start Date: 820215
End Date: 820215
Granularity Level: h
Starting Hour: 15
Ending Hour: 15
(P)rime (N)on Prime (B)oth or (W)eekend: p
Item 1: cpu0
Item 2:
Suppress Blank Ranges (Y/N)? y
                 A                B
         SYSTEM: PATH      ITEM: CPU0  D            E
          DAY 02/15/82  C   IN RANGE 02/15/82 TO 02/15/82
   PRIMETIME        TIME: 1500 G
     F
                                NUMBER OF     FREQUENCY %      % OF CUMULATIVE
    RANGE OF VALUES             SAMPLES       OF TOTAL              TOTAL

  50.01 -  H  100.00          J   1         K  1.6%          L     1.6%
 100.01 -     150.00              3            5.0%                6.6%
 150.01 -     200.00              1            1.6%                8.3%
 200.01 -     250.00              4            6.6%               14.9%
 250.01 -     300.00              7           11.6%               26.6% — M
 300.01 -     350.00             14           23.3%               49.9%
 350.01 -     400.00              6           10.0%               59.9%
 400.01 -     450.00              5            8.3%               68.3%
 450.01 -     500.00              4            6.6%               74.9%
 500.01 -     550.00              3            5.0%               79.9%
 600.01 -     650.00              1            1.6%               81.6%
 650.01 -     700.00              2            3.3%               84.9%
 750.01 -     800.00              1            1.6%               86.6%
 800.01 -     850.00              1            1.6%               88.3%
1150.01 -    1200.00 —— I         1            1.6%               89.9%
1350.01 -    1400.00              1            1.6%               91.6%
1550.01 -    1600.00              1            1.6%               93.3%
2050.01 -    2100.00              1            1.6%               94.9%
2100.01 -    2150.00              1            1.6%               96.6%
2750.01 -    2800.00              1            1.6%               98.3%
3500.01 -    3550.00              1            1.6%               99.9%

          AVERAGE VALUE FOR         60 SAMPLES IS        573.49
                                    N                    P
More Requests (Y/N): n

EXIT

.
```

Figure 1-11

The final column (L), % OF CUMULATIVE TOTAL, gives the percent of all samples whose values fell into the previous ranges plus the percent of those samples whose values fell into the current range. For example, on the sample report, values for user UUO's per second were less than or equal to 300 only 26.6% (M) of the time. Conversely, the values for user UUO's per second were greater than 300, 73.4% of the time (100%-26.6%).

At the bottom of the report, the total number of samples (N) and the average value (P) for the report period are given. The average value is the same value that would be displayed if you had elected to use a Table of Average Values instead of a Histogram for this item and time period.

blank page

1.3.7 Data Extraction Records

The Data Extraction Program, AMAREX, is used to extract four types of records from either the System AMAR database (AMAR.DB) or the output file produced by AMARSD (TODAY.DB). The record types are:

Performance Summary
Performance Detail
Granularity
System Calendar

The AMAREX program extracts data for display at the terminal or for input to your own programs. AMAREX eliminates the need for you to have to deal with System AMAR's internal database format.

1.3.7.1 Performance Summary (PS) Record -

The PS records contain mean (or average) values for specific time periods such as hours, days, weeks, etc. The number of samples taken during the period and their mean value are given. This particular type of record will be produced when AMAREX is run with the parameters shown in the example on the next page. Record type is the only critical parameter that must be typed as shown. The others can be varied. Refer to the Appendix called "Data Extraction Program (AMAREX) Dialogue" for further information.

	Field Description	Width
A.	Record format	2
B.	Record type (e.g., PS)	2
C.	Record length (blank)	4
D.	Blank	5
E.	Four-character system code	4
F.	Normal calendar year	2
G.	Normal calendar month	2
H.	Normal calendar day	2
I.	Hour (military time)	2
J.	Minutes	2
K.	Fiscal year	2
L.	Fiscal quarter	1
M.	Fiscal month*	1
N.	Fiscal week*	1
O.	Fiscal day*	1
P.	Granularity (summary level)	1
Q.	System AMAR item (e.g. CPU0, LUI0)	4
R.	System AMAR subitem (i.e., item specified for a particular peripheral device)	9
S.	Prime indicator (P,N or blank)	1
T.	Count of samples	10
U.	Mean value (integer part)	8
V.	Mean value (decimal part)	2

```
.run amarex[,,amar]

DATABASE NAME:amar
OUTPUT:extrc1.txt
RECORD TYPE:ps
DATE:820215:1301-820215:1500
[820215:1301-820215:1500 = 8232421301-8232421500]
GRANULARITY LEVEL:s
RESTRICTING ANY FISCAL PERIOD?no
ITEM:cpi0,cpo0,cpu0,nrjr,luiodske0,luiodskr0
PRIMETIME:all
[ EXTRACTING ]


[ SPECIFY NEXT EXTRACTION CRITERIA ]

OUTPUT:^C

.
.ty extrc1.txt
AAPS             PATH82021514008232421CPI0             P00000000600000004759
AAPS             PATH82021514008232421CPO0             P00000000600000001489
AAPS             PATH82021514008232421CPU0             P00000000600000059712
AAPS             PATH82021514008232421LUIODSKE0        P00000000600000000135
AAPS             PATH82021514008232421NRJR             P00000000600000000194
AAPS             PATH82021515008232421CPI0             P00000000600000003787
AAPS             PATH82021515008232421CPO0             P00000000600000001778
AAPS             PATH82021515008232421CPU0             P00000000600000057349
AAPS             PATH82021515008232421LUIODSKE0        P00000000600000000295
AAPS             PATH82021515008232421NRJR             P00000000600000000259
A B  C   D       E   F G H I J K     Q     R       S    T         U       V
                              LMNOP
```

Figure 1-12

*If a higher granularity level is selected for reporting, these fields will contain the number representing the last day, week, or month (as appropriate) in the fiscal period specified.

1.3.7.2 Performance Detail (PD) Record -

The PD records contain frequency distribution data which can be used to produce histograms showing the percent of samples whose values fell into specific ranges. The number of samples and the mean (average) values for the range are given. The boundaries of the range must be derived from the class widths defined in the System AMAR database. The PD record will be produced when AMAREX is run with the parameters shown in the example on the next page. Record type is the only critical parameter that must be typed as shown. The others can be varied. Refer to the Appendix called "Data Extraction Program (AMAREX) Dialogue" for further information.

	Field Description	Width
A.	Record format	2
B.	Record type (e.g., PD)	2
C.	Record length (blank)	4
D.	Blank	5
E.	Four-character system code	4
F.	Normal Calendar year	2
G.	Normal Calendar month	2
H.	Normal Calendar day	2
I.	Hour (military time)	2
J.	Minutes	2
K.	Fiscal year	2
L.	Fiscal quarter	1
M.	Fiscal month*	1
N.	Fiscal week*	1
O.	Fiscal day*	1
P.	Granularity (summary level)	1
Q.	System AMAR item (e.g., CPU0, LUIO)	4
R.	System AMAR subitem (i.e., item specified for a particular peripheral device)	9
S.	Prime indicator (P,N or blank)	1
T.	Count of samples	10
U.	Mean value (integer part) for the frequency class	8
V.	Mean value (decimal part)	2

*If a higher granularity level is selected for reporting, these fields will contain the number representing the last day, week, or month (as appropriate) in the fiscal period specified.

```
.run amarex[,,amar]

DATABASE NAME:amar
OUTPUT:extrc2.txt
RECORD TYPE:pd
DATE:820215:1401-820215:1500
[820215:1401-820215:1500 = 8232421401-8232421500]
GRANULARITY LEVEL:s
RESTRICTING ANY FISCAL PERIOD?no
ITEM:cpu0,luiodske0
PRIMETIME:all
[ EXTRACTING ]


[ SPECIFY NEXT EXTRACTION CRITERIA ]

OUTPUT:^C


.
.ty extrc2.txt
AAPD         PATH8202151500823242 1CPU0            P00000000010000008516
AAPD         PATH8202151500823242 1CPU0            P00000000030000014276
AAPD         PATH8202151500823242 1CPU0            P00000000010000018792
AAPD         PATH8202151500823242 1CPU0            P00000000040000023163
AAPD         PATH8202151500823242 1CPU0            P00000000070000028168
AAPD         PATH8202151500823242 1CPU0            P00000000140000032897
AAPD         PATH8202151500823242 1CPU0            P00000000060000037070
AAPD         PATH8202151500823242 1CPU0            P00000000050000043445
AAPD         PATH8202151500823242 1CPU0            P00000000040000048792
AAPD         PATH8202151500823242 1CPU0            P00000000030000052562
AAPD         PATH8202151500823242 1CPU0            P00000000010000064512
AAPD         PATH8202151500823242 1CPU0            P00000000020000066909
AAPD         PATH8202151500823242 1CPU0            P00000000010000076099
AAPD         PATH8202151500823242 1CPU0            P00000000010000082005
AAPD         PATH8202151500823242 1CPU0            P00000000010000116729
AAPD         PATH8202151500823242 1CPU0            P00000000010000139049
AAPD         PATH8202151500823242 1CPU0            P00000000010000159924
AAPD         PATH8202151500823242 1CPU0            P00000000010000209549
AAPD         PATH8202151500823242 1CPU0            P00000000010000212768
AAPD         PATH8202151500823242 1CPU0            P00000000010000278830
AAPD         PATH8202151500823242 1CPU0            P00000000010000354622
AAPD         PATH8202151500823242 1LUIODSKE0       P00000000570000000205
AAPD         PATH8202151500823242 1LUIODSKE0       P00000000020000001422
AAPD         PATH8202151500823242 1LUIODSKE0       P00000000010000003189
A B  C    D   E   F G H I J K L M N O P  Q     R   S      T        U     V
```

Figure 1-13

1.3.7.3 Granularity (GR) Record -

The GR records contain a timestamp indicating the fiscal period for which data has been collected. This particular type of record will be produced when AMAREX is run with the parameters shown in the example on the next page. Record type is the only critical parameter that must be typed as shown. The others can be varied. Refer to the Appendix called "Data Extraction Program (AMAREX) Dialogue" for further information.

Field Description	Width
A. Record format	2
B. Record type (e.g., GR)	2
C. Record length (blank)	4
D. Family (type of processor (blank))	5
E. Four-character system code	4
F. Normal calendar year	2
G. Normal calendar month	2
H. Normal calendar day	2
I. Hour (military time)	2
J. Minutes	2
K. Fiscal year	2
L. Fiscal quarter	1
M. Fiscal month*	1
N. Fiscal week*	1
O. Fiscal day*	1
P. Granularity (summary level)	1

*If a higher granularity level is selected for reporting, these fields will contain the number representing the last day, week, or month (as appropriate) in the fiscal period specified.

```
.run amarex[,,amar]

DATABASE NAME:amar
OUTPUT:extrc3.txt
RECORD TYPE:gr
DATE:820215-820215
[820215:0001-820215:2400 = 8232420001-8232422400]
GRANULARITY LEVEL:s
RESTRICTING ANY FISCAL PERIOD?no
[ EXTRACTING ]


[ SPECIFY NEXT EXTRACTION CRITERIA ]

OUTPUT:^C

.
.ty extrc3.txt
AAGR         PATH82021501008232421
AAGR         PATH82021502008232421
AAGR         PATH82021503008232421
AAGR         PATH82021504008232421
AAGR         PATH82021505008232421
AAGR         PATH82021506008232421
AAGR         PATH82021507008232421
AAGR         PATH82021508008232421
AAGR         PATH82021509008232421
AAGR         PATH82021510008232421
AAGR         PATH82021511008232421
AAGR         PATH82021512008232421
AAGR         PATH82021513008232421
AAGR         PATH82021514008232421
AAGR         PATH82021515008232421
AAGR         PATH82021516008232421
AAGR         PATH82021517008232421
AAGR         PATH82021518008232421
AAGR         PATH82021519008232421
AAGR         PATH82021520008232421
AAGR         PATH82021521008232421
AAGR         PATH82021522008232421
AAGR         PATH82021523008232421
AAGR         PATH82021524008232421
|A|B| C |  D  |  E  |F|G|H|I|J|K|||||
                               LMNOP
```

Figure 1-14

1.3.7.4 System Calendar (SC) Record -

The SC records contain the date and time of a system reload and the system uptime. This particular type of record will be produced when AMAREX is run with the parameters shown in the example on the next page. Record type is the only critical parameter that must be typed as shown. The others can be varied. Refer to the Appendix called "Data Extraction Program (AMAREX) Dialogue" for further information.

	Field Description	Width
A.	Record format	2
B.	Record type	2
C.	Blank	13
D.	Date system reloaded (year, month, day)	6
E.	Time system reloaded (hours, minutes)	4
F.	Blank	46
G.	Zeros	10
H.	Blank	34
I.	Date/time of system reload*	12
J.	System up-time (seconds)	12

*The date/time is expressed as the number of seconds since 00:00:00 (midnight) on November 17,1858. If this number is divided by the number of seconds in a day, the quotient will be the left half of a National Bureau of Standards date/timestamp.

```
.run amarex[,,amar]

DATABASE NAME:amar
OUTPUT:extrc4.txt
RECORD TYPE:sc
DATE:820215-820217
[820215:0001-820217:2400 = 8232420001-8232442400]
[ EXTRACTING ]

[ SPECIFY NEXT EXTRACTION CRITERIA ]

OUTPUT:^C

.
.ty extrc4.txt
AASC                 8202112318
0000000000                                                     003889034284000000324627
AASC                 8202151730
0000000000                                                     003889359005000000107284
AASC                 8202162320
0000000000                                                     003889466405000000005689
```

A B C D E F G H I J

Figure 1-15

1.4 HOW TO RUN THE PROGRAMS

1.4.1 Data Collection

The Data Collection Program, xxxxDC (where xxxx is the 4 character system code), should be run as an OPSER subjob. It should also be run under [1,2] in order to collect those variables obtained from the performance meter. See the Appendix called "System AMAR Item Definitions" for a list of such variables. There will be a corresponding subjob to collect data for Workload AMAR.

The following commands should be inserted into the OPR.ATO file to ensure automatic startup and continuous data collection.

```
:SLOGIN 1,2
:DEFINE AMAR=
AMAR-RUN structure:xxxxDC[ppn]
```

These commands may be entered directly to OPSER to get xxxxDC started the first time. The third command line may be used to restart xxxxDC if it has stopped because of disk parity errors or the like.

Each day xxxxDC creates an output file named xxxxdd.mmm where xxxx is the 4 character system code; dd is the day; and mmm is the alpha abbreviation for the month. For example, on the "PATH" system, the raw file created on January 9th would be called PATH09.JAN. Since xxxxDC will write to the raw file throughout the day, the pack used for the raw file must be permanently mounted. Otherwise, data will be irretrievably lost.

1.4.2 Generating Automatic Reports

The AMREPT program, the xxxxDR.RFD file and the System AMAR database are used to generate automatic reports. AMREPT is normally run as part of a nightly batch stream, AMAR.CTL, which is self-submitting. xxxxDR.RFD contains all the commands used to define each report's contents. The System AMAR database keeps track of the records on which AMREPT has already reported. AMREPT checks the database to see when data for a fiscal day, week or month is ready for reporting. It then produces the appropriate daily, weekly and monthly reports. Several report requests are included in the AMAR.CTL stream as defaults. You can specify your own set of automatic reports by commenting in or commenting out the appropriate report codes and associated dialogue.

AMREPT is explained in detail in the Appendix called "Report Program (AMREPT) Dialogue". The appendix also describes how to use AMREPT to generate reports on demand. For quick reference, the dialogue to generate automatic reports is shown in Figure 1-16.

```
.ru amrept

Report Code> du
Input File> amar
Dates> auto
Print File> pathdu.rpt
Report Code> exit

EXIT
```

Report Code> du — Unique 2 character code denoting type of report to be printed.

Input File> amar — AMAR or AMAR.DB specifies the System AMAR database as the input file.

Dates> auto — AUTO indicates that a check will be made to determine the date of the last fiscal period (day, week, or month) for which an automatic report of the same type has already been generated. The next appropriate fiscal period will be used for this report providing the necessary data is in the database.

Print File> pathdu.rpt — Report filename.

Report Code> exit — Terminates the program.

Figure 1-16

* * * * * * * * *

1.4.3 Generating Special Reports - What Program Do I Use?

There are four programs for generating special reports - AMARSD, AMREPT, AMARON, and AMAREX.

Use AMREPT alone to produce standard reports from the database. See Figure 1-17 and the Appendix called "Report Program (AMREPT) Dialogue".

Use AMARSD and AMREPT together if you want to look at today's data which will not be in the database until after midnight. Also use AMARSD and AMREPT to look at any other daily raw file which has not been entered into the database. See Figure 1-18 and the Appendix called "Raw File Preprocessor Program (AMARSD) Dialogue".

Use AMARON if you want to look at selected items from the database in either histogram or tabular format. Data can be examined directly at the terminal or put into files for later processing by your own programs or statistical packages. See Figures 1-10 and 1-11 in the Section called "Online Inquiry Reports" and the Appendix called "Online Inquiry Program (AMARON) Dialogue".

Use AMAREX as an alternative to AMARON for extracting data into sequential format for later processing by your own programs. See Figures 1-12 through 1-15 in the Section called "Data Extraction Records" and the Appendix called "Data Extraction Program (AMAREX) Dialogue".

* * * * * * * * *

```
.ru amrept

Report Code> dd                Daily Disk Report
Input File> amar
Dates> 820215                  System AMAR database
Print File> pathdd.rpx
Report Code> exit              February 15, 1982

EXIT
```

Dialogue to Produce a Standard Report
Figure 1-17

* * * * * * * * *

FROM: 15-FEB-82 (MONDAY)
THRU: 15-FEB-82 (MONDAY)

- AMAR -
DAILY DISK REPORT

PAGE: 1

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

GENERAL USAGE SUMMARY

	PRIME TIME	NON-PRIME TIME
HOURS THEORETICALLY AVAILABLE	9:00	15:00
HH:MM SYSTEM WAS UP	9:00	14:58
HH:MM AMAR MEASURED THE SYSTEM	9:00	14:29
SWAPPING BLKS/SEC	35	0
% SWAP SPC LEFT	39	87
CHAN 0 % TIM BUSY	1.19%	.03%
CHAN 1 % TIM BUSY	.38%	.00%
CHAN 2 % TIM BUSY	1.58%	.30%
CHAN 3 % TIM BUSY	1.14%	.23%
CHAN 4 % TIM BUSY	.00%	.00%
CHAN 5 % TIM BUSY	.00%	.00%
CHAN 6 % TIM BUSY	.00%	.00%
CHAN 7 % TIM BUSY	.00%	.00%
PI 0 % TIM BUSY	.14%	.01%
PI 1 % TIM BUSY	.00%	.00%
PI 2 % TIM BUSY	.14%	.14%
PI 3 % TIM BUSY	.94%	.08%
PI 4 % TIM BUSY	1.25%	.21%
PI 5 % TIM BUSY	.14%	.01%
PI 6 % TIM BUSY	.00%	.00%
PI 7 % TIM BUSY	7.07%	3.65%
CHAN 00 WAIT Q	.0	.0
CHAN 01 WAIT Q	.0	.0
CHAN 02 WAIT Q	.0	.0
CHAN 04 WAIT Q	.0	.0

PRIME TIME PACK SUMMARY

PACK NAME	TOTAL TIME(HH:MM) MOUNTED	- IN USE	% OF TIME MOUNTED	- IN USE	% MOUNTED TIME IN USE	PACK FREE SPC	PACK BLKS/SEC	PACK SWPS/SEC	PACK WAIT Q
CASEO	8:57 -	8:26	99% -	94%	94%	40%	15		.0
COREO	8:57 -	2:56	99% -	33%	33%	46%	1		.0
DEVPO	8:57 -	4:10	99% -	46%	47%	9%	16		.0
DSKEO	8:57 -	8:57	99% -	99%	100%	29%	3	35	.0
DSKWO	8:57 -	7:46	99% -	86%	87%	78%	6		.0
HRS20	1:37 -	1:01	18% -	11%	63%	60%	31		.0
PENSO	8:57 -	0:45	99% -	8%	8%	13%	0		.0
PER60	3:39 -	1:04	41% -	12%	29%	23%	1		.0
PLT10	7:49 -	2:28	87% -	27%	32%	42%	1		.0
STARO	8:57 -	2:06	99% -	23%	23%	41%	0		.0
TRNGO	8:13 -	0:38	91% -	7%	8%	74%	1		.0
USRSO	8:57 -	8:54	99% -	99%	99%	19%	13		.1

---------- CONTINUED NEXT PAGE ----------

Figure 1-17 (continued)

* * * * * * * * *

```
.run amarsd

System ID>path
YYMMDD Date of file>820217
Prime periods>0800-1700

[ARMDSF Data segment full - storing in first free page]
[ARMDSF Data segment full - storing in first free page]

[AMIHDS Hourly data stored for 820217]
[AMIDUD Database up to date]
[Use TODAY.DB as input file to the AMREPT report program]

EXIT

..run amrept

Report Code> du
Input File> today.db
Dates> 820217
Print File> pathxx.rpx
Report Code> exit

EXIT
```

Raw file from February 17, 1982. The raw filename is PATH17.FEB.

Message indicating that preallocated storage has been used up. Secondary database storage is now being automatically used.

Mini-database created by AMARSD above and used as the input file here.

Dialogue to Look at Today's Data
Figure 1-18

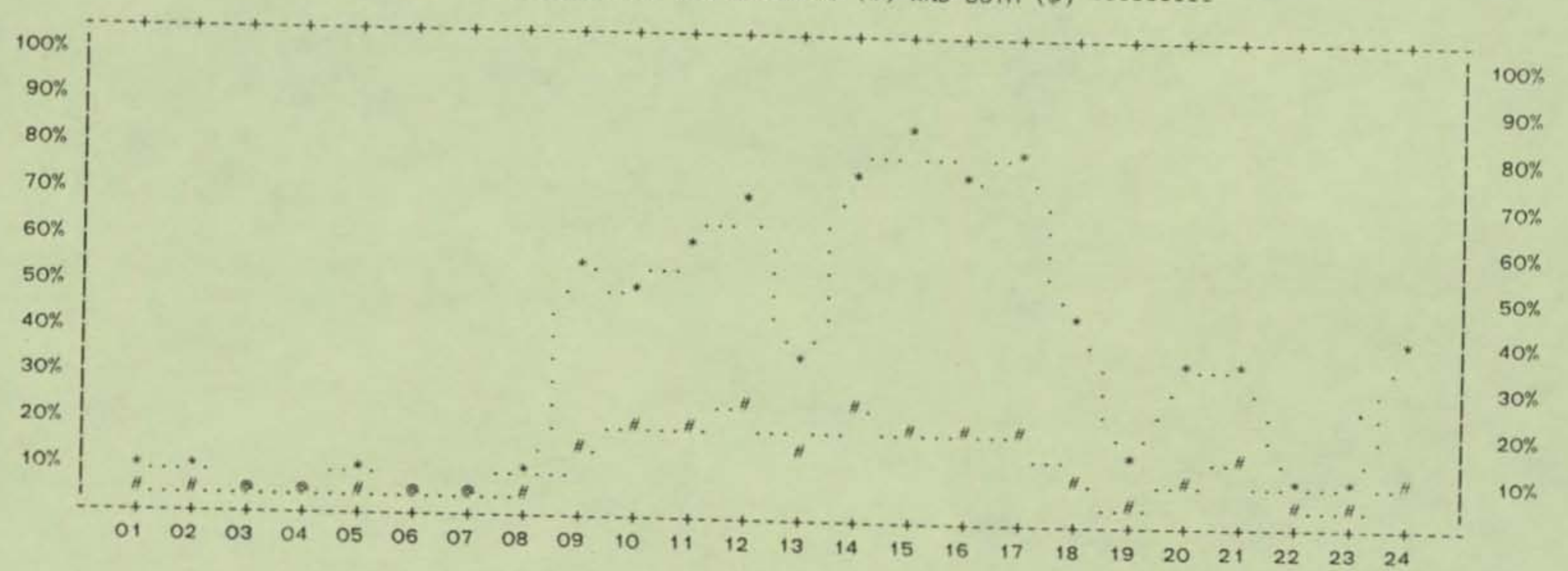
DATE: 17-FEB-82 (WEDNESDAY)

- AMAR -
DAILY SYSTEM UTILIZATION SUMMARY REPORT

PAGE: 1

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

--------- CPU UTILIZATION (*) OVERHEAD (#) AND BOTH (@) ---------



------------------- SUMMARY OF KEY UTILIZATION ITEMS ----------------

-------AVERAGE-----	# JOBS LOGGED IN	# LINES IN USE	% IDLE TIME	% LOST TIME	% OVHD TIME	ACT SWAP RATIO	SWAPPING BLKS/SEC	USER DSK BLKS/SEC	# MTAS ASSIGNED	PGS USER MEMORY	NO. OF HOURS KEY ITEMS OVER LIMITS
----PRIME TIME-----	60	36	38	1	17	.2	33	80	1	1735	2
---NON-PRIME TIME--	25	5	85	0	5	.0	2	52	0	1735	2

---------- PERCENT FREE SPACE LEFT ON DISK PACKS ----------

-----PACK NAME-----	BKP20	BKP21	BKP22	CASE0	CORE0	DEVP0	DSKE0	DSKT0	DSKW0	HRS20	PAWS0	PENS0	PER10	PER20	PER60	PLT10	STAR0
---% FREE TODAY----	55	55	55	38	46	4	31	100	75	11	66	13	46	15	23	41	42
-% FREE YESTERDAY--																	
----DIFFERENCE-----																	

+ = MORE THAN YESTERDAY - = LESS THAN YESTERDAY

---------- CONTINUED NEXT PAGE --------

Figure 1-18 (continued)

blank page

1.4.4 Examining/Changing Database Parameters (AMRGEN)

The AMRGEN program allows you to examine certain parameters within the System AMAR database, notably data retention times, the prime time schedule, and the list of valid items and class widths. It may also be used to change data retention times. AMRGEN should be run before the end of each fiscal year to define the holiday list for the next year. Otherwise, holidays will be treated as normal workdays.

AMRGEN is command driven. It is procedural - certain commands depend on prior commands having been issued.

AMRGEN prompts with an asterisk (*).

Valid commands are:

```
SET RETENTION HOURLY <number of retained periods>
              DAILY
              WEEKLY
              MONTHLY
              COWEEKLY
              COMONTHLY
              LOG
```

Function: To specify retention times for each granularity (summary) level. Note that retention time directly affects the space required for the database. See the Appendix called "Installation and Resource Requirements" for space estimates.

<number of retained periods> = 1-3 digit retention count for a granularity level.

HOURLY = number of days hourly data is to be kept (default=7)

DAILY = number of days daily data is to be kept (default=35)

WEEKLY = number of weeks weekly data is to be kept (default=13)

MONTHLY = number of months monthly data is to be kept (default=12)

COWEEKLY = number of weeks weekly composite data is to be kept (default=5)

COMONTHLY = number of months monthly composite data is to be kept (default=3)

LOG = number of days uptime log records are to be kept (default=91)

LIST NAME

Function: To list the contents of the System Header Record which includes rollup date, retention times, last time automatic reports were generated, and date of last input.

SET WEEKDAY<hhmm-1>-<hhmm-2>,...,<hhmm-7>-<hhmm-8>

Function: To change the prime time schedule for all weekdays. Changes can only be made to dates for which no data has been entered. Up to 4 prime time pairs may be specified.

hh = hours; mm = minutes.

SET HOLIDAY <yymmdd>

Function: To define a given date <yymmdd> as a holiday. In addition to setting holidays for System AMAR, holidays must also be set for Workload AMAR. See the Section called "Examining/Changing the Workload Holidays (WCFIX)".

yy = normal calendar year; mm = month; dd = day.

LIST HOLIDAY

Function: To list the holidays.

DROP HOLIDAY <yymmdd>

Function: To remove a holiday from the holiday list.

yy = normal calendar year; mm = month; dd = day.

LIST PRIMETIMES <yymmdd-yymmdd>

Function: To list the prime times of the days within the date range.

yy = normal calendar year; mm = month; dd = day.

```
LIST CLASS <Item><Subitem>
           <Item>
           ALL
```

Function: To list one or more items and subitems and their class widths.

<Item> - 4 character item code

<Subitem> - 7 character subitem code

ALL - All items and subitems in the database.

```
ADD ITEM <Item><Subitem>
         <Item>
         ALL
```

Function: To enable an item and/or subitem for database storage. Only items already enabled in the data collection program should be named here. Don't make up random item or subitem names!

ALL - The rest of the valid items not yet enabled.

DROP ITEM <Item><Subitem>
<Item>

Function: To delete an item and/or subitem and all its data from the database.

EXIT DATABASE

Function: To gracefully terminate the program.

1.5 HOW TO TAILOR THE REPORT CONTENTS

1.5.1 Editing The Report File Description

The file called xxxxDR.RFD (where xxxx is the 4 character system code) contains information used to control the content, and, to some extent, the format of the System AMAR reports. Refer to Figure 1-19. The RFD file controls which items and subitems get printed on the reports, the titles used to describe these items and subitems, the short and long term thresholds used to test for potential problems, the contents of comment fields, and the paging of subreports. Through editing this file, you can tailor the reports to fit your site's special requirements.

Each field within the RFD file is described below. All fields are separated by tabs. For further information, you can also refer to the RFD.HLP file in your System AMAR area. Following the field descriptions are examples of some common changes to the RFD file. Changes will remain in effect until they are changed again by you -- there is no automatic resetting or reverting feature. Changes will only apply to the way the reports look. You will not be altering any data in the database or data collection programs.

<u>System</u> <u>Description</u> <u>Section:</u> This section contains system and site identification data and test criteria for printing severity codes in the comment field of the Daily System Utilization Report.

A. Command Code. The first 3 characters of each line indicate the purpose of the line. Only these command codes are valid:

.C - Comment
.SD - System Description
.SS - System Specification
.ID - Item Description
.RD - Report Description
.RI - Report Items

B. System Name. The 4 character system code which is used to identify this file, the data collection program, raw data files, etc.

C. System ID. The 20 character system description which appears in all standard report headings. This field must be delimited by underscores (_) and must be exactly 20 characters in length (blank fill if necessary).

D. Plot Graphics. Symbols printed for data points on all graphs produced by AMAR.

First symbol (*)	-	Denotes CPU utilization.
Second symbol (#)	-	Denotes overhead.
Third symbol (@)	-	Denotes where CPU utilization and overhead values are the same. CPU utilization includes overhead.

E. Sample Level Limits. These fields are used for testing the percent of samples over the watchdog limits and for specifying an appropriate code to denote the severity of the situation. The severity code is printed in the comments section on the "Problem Report" page of the "Daily Utilization Summary Report". Each field is parsed as follows:

<Percentage for prime time testing>,<Percentage for non-prime time testing> Severity Code

F. Hourly Level Limits. These fields are used for testing the percent of hourly averages over the watchdog limits and for specifying an appropriate code for the severity of the situation. See "Sample Level Limits" above.

<u>Item Description Section:</u> This section defines the title to be used for identifying an item on a report, the short and longterm watchdog limits, and the comment to be printed when an item or subitem exceeds its limits.

G. Item Code. This code uniquely identifies the item measured by System AMAR. You should never change it. Refer to the Appendix called "System AMAR Item Definitions" for the meaning of each code.

H. Subitem Code. This code uniquely identifies the subitem measured by System AMAR. Only devices whose codes appear in your database should be listed here. Use the AMRGEN program to find out the valid subitem codes. A string of question marks ("??...") represents "all" devices or "all other" devices (if at least one device of the same type has been explicitly listed). Question marks, if used, should always preceed the explicit device names. The device names should be in numerical or alphabetical order.

I. Item/Subitem Title. This field contains a 20 character short title for the item or subitem. The title must have exactly 20 characters, including blanks. It is split into two groups of 10 characters each - preceeded, separated, and followed by an underscore ("_").

J. Picture Specification. This field contains the print format for the item/subitem values. "R" in this field denotes the position at which rounding will occur. The values of all items/subitems are stored in the database with 2 decimal positions regardless of the format shown here.

K. Data Type. This field contains a single character to be printed following the value of the item/subitem. It tells in what units the values have been expressed. Normally the only character used is "%" indicating percentage.

L. Short Term Threshold. This field defines the value for the watchdog limit when testing for the percent of samples (or whether or not the average is) over (">" greater than) or under ("<" less than) the watchdog limit. It is used for testing time periods of an hour or day. If no value is specified, testing will not occur.

M. Long Term Threshold. This field defines the value used for the watchdog limit when testing the percent of samples (or whether or not the average is) over (">" greater than) or under ("<" less than) the watchdog limit. It is used for testing periods of a week or month. If no value is specified, the short term threshold, if one exists, will be used.

N. Comment. This field defines a comment to be displayed on the Utilization Reports whenever the applicable watchdog limit has been exceeded. The comment may contain up to 35 characters. The comment usually describes the probable cause for the limit having been exceeded or a procedure to follow to investigate or correct a problem.

<u>Report Description Section:</u> This section defines which items get printed on the reports and whether or not they are always printed or printed only when watchdog limits are exceeded.

P. Subsection Title. Each System AMAR report contains one or more subsections where items are grouped and printed. This field defines the title which will preceed the subsection. Examples of such default titles are "KEY UTILIZATION ITEMS", "OTHER UTILIZATION ITEMS", and "GENERAL SUMMARY". This field must be exactly 30 characters including blanks.

Q. Before Command. This field controls paging prior to printing the subsection defined here. If the field contains the words "PAGING BEFORE", the printer will eject to a new page before printing the subsection. If the field is blank, there will be no paging prior to printing the subsection.

R. After Command. This field controls paging after printing the subsection defined here. If it contains the words "PAGING AFTER", the printer will eject to a new page after printing the subsection. If the field is blank, there will be no paging after printing the subsection unless the next subsection description for the same report specifies "PAGING BEFORE".

S. Report Code. This field lists the 2 character report codes which define the reports for which the subsection should be printed. The field may contain up to 14 report codes separated by a blank. To receive the trend analysis sections (typically pages 3 - on) of the Weekly or Monthly Trend Analysis Reports, you must follow the applicable report code with the number 03 preceeded by a blank. The example RFD in this Guide would cause the trend analysis sections to be supressed. Note that for databases created before Release 4.1 of AMAR, the trend analysis sections will be produced even if 03 is not specified.

T. Item Code. This field contains the 4 character item code for each item to be printed in this subsection. Items may be listed in any order here and will be printed in that order. To delete an item from a report subsection, just delete the appropriate ".RI" line from that subsection. Conversly, to cause an item to be printed in a report subsection, add the appropriate ".RI" line with the item code to that subsection.

NOTE: Do not list an item more than once in any subsection. This will cause the report program to loop and exhaust your disk quota. Also, in order to get the disk report, at least one item must be specified and this item must have data in the System AMAR database.

U. Subitem Code. This field contains the subitem code (up to 6 characters) for each subitem to be printed in this subsection. Individual device names may be listed here even if they have not been explicitly defined in the "Item Description Section". Subitems, except for disk subitems, may be listed in any order here and will be printed in that order. Disk subitems will always be printed in alphabetical order regardless of their order in the RFD file. Question marks in this field indicate that "all" or "all other" devices should be printed. Question marks, if used, should follow the list of explicitly named devices. On disk reports, all subitems should be explicitly listed or only question marks should be used. Ususally disk subitems will be explicitly listed only if you want to prevent some disks from printing.

NOTE: Do not list a subitem more than once per subsection. This will cause the report program to loop and exhaust your disk quota. Also do not mix item and subitem groups. For example, list all LUFS specifications together, then all LUIO specifications, etc. The order of the item groups controls the print order. In the example just given, LUFS would print before LUIO, etc.

Failure to observe the conventions in this RFD file may have unexpected results.

V. Treatment Code. This field determines whether or not values for the item or subitem will always be printed. "FORCED" means always print the values for the item or subitem. "TESTED" means print the values for the item or subitem only if the appropriate watchdog limit has been exceeded. Watchdog limits are considered to be exceeded if in the case of a high limit, the average value or at least 10% of the samples equal or exceed the limit, or, in the case of a low limit, the average value or at least 10% of the samples equal or fall below the limit.

NOTE: TESTED has meaning only when using the DU, WU, and MU report codes. Any item listed for other reports will always be FORCED even if TESTED is specified.

W. The last line of the xxxxDR.RFD file should always be:

    .RD<tab>_END REPORTS

This tells AMREPT that no further report descriptions have been specified.

```
.C      ******** SYSTEM DESCRIPTION SECTION ********

.SD     PATH     _ PUT ANY TITLE HERE_
.SS     *#@     10,10_WARNING   25,50_SERIOUS   50,80_CRITICAL  10,10_WARNING   25,50_SERIOUS   50,80_CRITICAL

.C      ******** ITEM DESCRIPTION SECTION ********

.ID     ACCR                _ ACTV JOB _ % USR CR _ NNNNNNR.%
.ID     ACSR                _ ACT SWAP _  RATIO   _ NNNNNN.R    >000000.7,>000000.5    CPU/MEMORY SHORTAGE: CHK AMOUNT
.ID     AVJS                _ AVG JOB  _   SIZE   _ NNNNNNR.                           JOB SIZES TOO BIG: CHK WORKLD DATA
.ID     AVRT                _ AVG SCHD _ RSP TIME _ NNNNNNR.    >0000200.,>0000200.    SCHEDULER SLOW: CHK HPQ USE, WORKLD
.ID     CH00                _ CHAN 0 % _ TIM BUSY _ NNNNN.NR%
.ID     CH01                _ SLV CHAN _ 0 % BUSY _ NNNNN.NR%
.ID     CH10                _ CHAN 1 % _ TIM BUSY _ NNNNN.NR%
.ID     CH11                _ SLV CHAN _ 1 % BUSY _ NNNNN.NR%
.ID     CH20                _ CHAN 2 % _ TIM BUSY _ NNNNN.NR%
.ID     CH21                _ SLV CHAN _ 2 % BUSY _ NNNNN.NR%
.ID     CH30                _ CHAN 3 % _ TIM BUSY _ NNNNN.NR%
.ID     CH31                _ SLV CHAN _ 3 % BUSY _ NNNNN.NR%
.ID     CH40                _ CHAN 4 % _ TIM BUSY _ NNNNN.NR%
.ID     CH41                _ SLV CHAN _ 4 % BUSY _ NNNNN.NR%
.ID     CH50                _ CHAN 5 % _ TIM BUSY _ NNNNN.NR%
.ID     CH51                _ SLV CHAN _ 5 % BUSY _ NNNNN.NR%
.ID     CH60                _ CHAN 6 % _ TIM BUSY _ NNNNN.NR%
.ID     CH61                _ SLV CHAN _ 6 % BUSY _ NNNNN.NR%
.ID     CH70                _ CHAN 7 % _ TIM BUSY _ NNNNN.NR%
.ID     CH71                _ SLV CHAN _ 7 % BUSY _ NNNNN.NR%
.ID     CPA0                _ % AMAR TI_ CPU 0 UP _ NNNNNNR.%   <0000090.,<0000095.    TOO MUCH CPU0 DOWNTIME: CHK REASON
.ID     CPA1                _ % AMAR TI_ CPU 1 UP _ NNNNNNR.%   <0000090.,<0000095.    TOO MUCH CPU1 DOWNTIME: CHK REASON
.ID     CPC0                _ CONTEXT  _ SWTS/SEC _ NNNNNNR.
.ID     CPC1                _ SLV CTXT _ SWTS/SEC _ NNNNNNR.
.ID     CPI0                _ % IDLE   _   TIME   _ NNNNNNR.%   <0000005.,<0000030.    CPU PRESSED: CHK WORKLD DATA FIRST
.ID     CPI1                _ SLV IDLE _   TIME   _ NNNNNNR.%   <0000005.,<0000030.    SLAVE CPU PRESSED: CHK WORKLD DATA
.ID     CPL0                _ % LOST   _   TIME   _ NNNNNNR.%   >0000005.,>0000002.    MEMORY SHORTAGE OR SWAP DEVICE SLOW
.ID     CPL1                _ SLV LOST _   TIME   _ NNNNNNR.%   >0000005.,>0000002.    MEMORY SHORTAGE OR SWAP DEVICE SLOW
.ID     CPO0                _ % OVHD   _   TIME   _ NNNNNNR.%   >0000020.,>0000020.    OVERHEAD TOO HIGH: INVESTIGATE
.ID     CPO1                _ SLV OVHD _   TIME   _ NNNNNNR.%   >0000020.,>0000020.    OVERHEAD TOO HIGH: INVESTIGATE
.ID     CPS0                _  CACHE   _ SWEEP/SEC_ NNNNNNR.    >0000100.,>0000080.    CACHE SWEEPS HIGH: INVESTIGATE
.ID     CPS1                _ SLV CACH _ SWEEP/SEC_ NNNNNNR.    >0000100.,>0000080.    CACHE SWEEPS HIGH: INVESTIGATE
.ID     CPU0                _  USER    _ UUOS/SEC _ NNNNNNR.    >0000300.,>0000200.    USER PGM PROBLEM: CHK WORKLD DATA
.ID     CPU1                _ SLV USER _ UUOS/SEC _ NNNNNNR.    >0000300.,>0000200.    USER PGM PROBLEM: CHK WORKLD DATA
.ID     CTWQ    ??          _ CHAN ??  _  WAIT Q  _ NNNNNN.R    >000001.0,>000000.8    CHANNEL CONTENTION: CHK I/O BALANCE
.ID     IOWP                _ # JOBS   _ BLK IO Q _ NNNNNN.R    >000002.0,>000001.5    BACKUP OF I/O JOBS: CHK CONTENTION
.ID     JLOG                _ # JOBS   _LOGGED IN _ NNNNNNR.                           INCREASE JOBMAX OR RESTRICT LOGINS
.ID     LUFS    ?????       _%PK ????? _ FREE SPC _ NNNNNNR.%   <0000010.,<0000010.    DELETE UNNECESSARY FILES
.ID     LUIO    ?????       _ PK ????? _ BLKS/SEC _ NNNNNNR.    >0000100.,>0000080.    I/O RATE HIGH: CHK FOR CONTENTION
.ID     LUSW    ?????       _ PK ????? _ SWPS/SEC _ NNNNNNR.    >0000200.,>0000160.    SWAPPING HIGH: CHK MEMORY AMOUNT
.ID     LUWQ    ?????       _ PK ????? _  WAIT Q  _ NNNNNN.R    >000001.0,>000000.8    FILE/PACK CONTENTION: INVESTIGATE
.ID     MTAU                _ # MTAS   _ ASSIGNED _ NNNNNNR.                           MANY TAPES ASSIGNED: CHK ACTUAL USE
.ID     MTIO    ??????      _ MT ??????_ BLKS/SEC _ NNNNNNR.    >0000150.,>0000100.    TAPE I/O RATE HIGH: INVESTIGATE USE
.ID     NRJR                _ # JOBS   _ IN RUN Q _ NNNNNNR.    >0000006.,>0000005.    CPU BOTTLENECK OR SCHEDULER SLOW
.ID     PFCU                _ % MON FR _ CORE USED_ NNNNNNR.%   >0000075.,>0000060.    FREE CORE LOW: FIND CAUSE/UP AMOUNT
.ID     PHCR                _ ALL JOBS _ % USR CR _ NNNNNNR.%
.ID     PI00                _  PI 0 %  _ TIM BUSY _ NNNNN.NR%
.ID     PI01                _ SLV PI 0 _  % BUSY  _ NNNNN.NR%
.ID     PI10                _  PI 1 %  _ TIM BUSY _ NNNNN.NR%
.ID     PI11                _ SLV PI 1 _  % BUSY  _ NNNNN.NR%
.ID     PI20                _  PI 2 %  _ TIM BUSY _ NNNNN.NR%
.ID     PI21                _ SLV PI 2 _  % BUSY  _ NNNNN.NR%
.ID     PI30                _  PI 3 %  _ TIM BUSY _ NNNNN.NR%
.ID     PI31                _ SLV PI 3 _  % BUSY  _ NNNNN.NR%
```

Figure 1-19

```
.ID     PI40               _  PI 4 %  _ TIM BUSY _ NNNNN.NR%
.ID     PI41               _ SLV PI 4 _  % BUSY  _ NNNNN.NR%
.ID     PI50               _  PI 5 %  _ TIM BUSY _ NNNNN.NR%
.ID     PI51               _ SLV PI 5 _  % BUSY  _ NNNNN.NR%
.ID     PI60               _  PI 6 %  _ TIM BUSY _ NNNNN.NR%
.ID     PI61               _ SLV PI 6 _  % BUSY  _ NNNNN.NR%
.ID     PI70               _  PI 7 %  _ TIM BUSY _ NNNNN.NR%
.ID     PI71               _ SLV PI 7 _  % BUSY  _ NNNNN.NR%
.ID     PRJC               _ % RN JOB _  IN MEM  _ NNNNNNR.%
.ID     PTCU               _ %TY CHNK _  IN USE  _ NNNNNNR.%
.ID     PUFS    ?????      _%UN ????? _ FREE SPC _ NNNNNNR.%     <0000010.,<0000010.    DELETE UNNECESSARY FILES
.ID     PUIO    ?????      _ UN ????? _ BLKS/SEC _ NNNNNNR.      >0000100.,>0000080.    I/O RATE HIGH: CHK FOR CONTENTION
.ID     PUSW    ?????      _ UN ????? _ SWPS/SEC _ NNNNNNR.%     >0000200.,>0000160.    SWAPPING HIGH: CHK MEMORY AMOUNT
.ID     PUWQ    ?????      _ UN ????? _  WAIT Q  _ NNNNNN.R      >000001.0,>000000.8    FILE/PACK CONTENTION: INVESTIGATE
.ID     SRCV               _ SCN INTR _ RCVS/SEC _ NNNNNNR.
.ID     SWIO               _ SWAPPING _ BLKS/SEC _ NNNNNNR.      >0000200.,>0000160.    SWAPPING HIGH: CHK MEMORY AMOUNT
.ID     SWPS               _ % SWAP   _ SPC LEFT _ NNNNNNR.      <0000025.,<0000030.    SWAP SPACE LOW: ALLOCATE MORE
.ID     SXMT               _ SCN INTR _ XMTS/SEC _ NNNNNNR.
.ID     TIOW               _  # JOBS  _ TTY IO Q _ NNNNNNR.      >0000020.,>0000016.    INTERACTIVE USE HIGH: CHK RESPONSE
.ID     TTYU               _ # LINES  _  IN USE  _ NNNNNNR.                             TTY LINE USE HIGHER THAN PLANNED
.ID     UDIO               _ USER DSK _ BLKS/SEC _ NNNNNNR.      >0000150.,>0000100.    DISK I/O RATE HIGH: CHK PACK I/O
.ID     UMEM               _ PGS USER _  MEMORY  _ NNNNNNR.      <0000768.,<0000768.    MEMORY DOWN: MAY CAUSE SWAPPING
.ID     VMPF               _ VIR MEM  _ FAULT/SEC_ NNNNNN.R      >000005.0,>000005.0    VIRTUAL USERS PAGING TOO OFTEN
.ID     XAMT               _ % AMAR   _ CLK TIME _ NNNNNN.R%                            AMAR NOT RUNNING ENOUGH: CHK REASON
.ID     XRLD               _ # SYSTEM _ RELOADS  _ NNNNNNR.                             TOO MANY RELOADS: CHK REASONS
.ID     XUPT               _ % SYSTEM _  UPTIME  _ NNNNNN.R%                            DOWNTIME HIGH: CHK RELOAD REASONS
.ID     _CPU               _  %  CPU  _   UTIL   _ NNNNNNR.%     >0000095.,>0000070.    CPU PRESSED: CHK WORKLD DATA FIRST

.C      ******** REPORT DESCRIPTION SECTION ********

.RD     _       KEY UTILIZATION ITEMS        _   PAGING BEFORE   PAGING AFTER   DU WC MC

.RI     JLOG               FORCED
.RI     TTYU               FORCED
.RI     CPIO               FORCED
.RI     CPLO               FORCED
.RI     CPOO               FORCED
.RI     ACSR               FORCED
.RI     SWIO               FORCED
.RI     UDIO               FORCED
.RI     MTAU               FORCED
.RI     UMEM               FORCED

.RD     _KEY UTILIZATION ITEMS               _   PAGING BEFORE   PAGING AFTER   WU MU

.RI     _CPU               FORCED
.RI     JLOG               FORCED
.RI     TTYU               FORCED
.RI     CPIO               FORCED
.RI     CPLO               FORCED
.RI     CPOO               FORCED
.RI     ACSR               FORCED
.RI     SWIO               FORCED
.RI     UDIO               FORCED
.RI     MTAU               FORCED
.RI     UMEM               FORCED
.RI     XUPT               FORCED
.RI     XAMT               FORCED
.RI     XRLD               FORCED
```

Figure 1-19 (continued)

.RD	_OTHER UTILIZATION ITEMS		_	PAGING AFTER DU WU MU WC MC
.RI	ACCR		TESTED	
.RI	AVJS		TESTED	
.RI	AVRT		TESTED	
.RI	CH00		TESTED	
.RI	CH10		TESTED	
.RI	CH20		TESTED	
.RI	CH30		TESTED	
.RI	CH40		TESTED	
.RI	CH50		TESTED	
.RI	CH60		TESTED	
.RI	CH70		TESTED	
.RI	CH01		TESTED	
.RI	CH11		TESTED	
.RI	CH21		TESTED	
.RI	CH31		TESTED	
.RI	CH41		TESTED	
.RI	CH51		TESTED	
.RI	CH61		TESTED	
.RI	CH71		TESTED	
.RI	CPC0		TESTED	
.RI	CPC1		TESTED	
.RI	CPS0		TESTED	
.RI	CPU0		TESTED	
.RI	CPI1		TESTED	
.RI	CPL1		TESTED	
.RI	CPO1		TESTED	
.RI	CPS1	u	TESTED	v
.RI	CPU1		TESTED	
.RI	CTWQ	??	TESTED	
.RI	IOWP		TESTED	
.RI	LUFS	?????	TESTED	
.RI	LUIO	?????	TESTED	
.RI	LUSW	?????	TESTED	
.RI	LUWQ	?????	TESTED	
.RI	MTIO	??????	TESTED	
.RI	NRJR		TESTED	
.RI	PFCU		TESTED	
.RI	PHCR		TESTED	
.RI	PI00		TESTED	
.RI	PI10		TESTED	
.RI	PI20		TESTED	
.RI	PI30		TESTED	
.RI	PI40		TESTED	
.RI	PI50		TESTED	
.RI	PI60		TESTED	
.RI	PI70		TESTED	
.RI	PI01		TESTED	
.RI	PI11		TESTED	
.RI	PI21		TESTED	
.RI	PI31		TESTED	
.RI	PI41		TESTED	
.RI	PI51		TESTED	
.RI	PI61		TESTED	
.RI	PI71		TESTED	
.RI	PRJC		TESTED	
.RI	PTCU		TESTED	
.RI	PUFS	?????	TESTED	
.RI	PUIO	?????	TESTED	
.RI	PUSW	?????	TESTED	

Figure 1-19 (continued)

```
.RI     PUWQ    ?????   TESTED
.RI     SRCV            TESTED
.RI     SWPS            TESTED
.RI     SXMT            TESTED
.RI     TIOW            TESTED
.RI     VMPF            TESTED

.RD     _KEY UTILIZATION ITEMS      _       PAGING BEFORE   PAGING AFTER    WA MA

.RI     _CPU            FORCED
.RI     CPIO            FORCED
.RI     CPOO            FORCED
.RI     CPLO            FORCED
.RI     CPUO            FORCED
.RI     CPCO            FORCED
.C      CPI1            FORCED
.C      CPO1            FORCED
.C      CPL1            FORCED
.C      CPU1            FORCED
.C      CPS1            FORCED
.C      CPC1            FORCED
.RI     ACSR            FORCED
.RI     ACCR            FORCED
.RI     PHCR            FORCED
.RI     PRJC            FORCED
.RI     AVJS            FORCED
.RI     JLOG            FORCED
.RI     UMEM            FORCED
.RI     TIOW            FORCED
.RI     UDIO            FORCED
.RI     NRJR            FORCED
.RI     SWIO            FORCED
.RI     LUSW    ?????   FORCED
.RI     PUSW    ?????   FORCED
.RI     AVRT            FORCED
.RI     MTAU            FORCED
.RI     TTYU            FORCED
.RI     PTCU            FORCED
.RI     SRCV            FORCED
.RI     SXMT            FORCED
.RI     XUPT            FORCED
.RI     XAMT            FORCED
.RI     XRLD            FORCED

.RD     _GENERAL SUMMARY            _       PAGING BEFORE                   DD WD MD

.RI     SWIO            FORCED
.RI     SWPS            FORCED
.RI     CHOO            FORCED
.RI     CH1O            FORCED
.RI     CH2O            FORCED
.RI     CH3O            FORCED
.RI     CH4O            FORCED
.RI     CH5O            FORCED
.RI     CH6O            FORCED
.RI     CH7O            FORCED
.C      CHO1            FORCED
.C      CH11            FORCED
.C      CH21            FORCED
.C      CH31            FORCED
```

Figure 1-19 (continued)

```
.C     CH41            FORCED
.C     CH51            FORCED
.C     CH61            FORCED
.C     CH71            FORCED
.RI    PI00            FORCED
.RI    PI10            FORCED
.RI    PI20            FORCED
.RI    PI30            FORCED
.RI    PI40            FORCED
.RI    PI50            FORCED
.RI    PI60            FORCED
.RI    PI70            FORCED
.C     PI01            FORCED
.C     PI11            FORCED
.C     PI21            FORCED
.C     PI31            FORCED
.C     PI41            FORCED
.C     PI51            FORCED
.C     PI61            FORCED
.C     PI71            FORCED
.RI    CTWQ    ??      FORCED

.RD    _PACK NAME                   _                      PAGING AFTER   DD WD MD

.RI    LUFS    ?????   FORCED
.RI    LUIO    ?????   FORCED
.RI    LUSW    ?????   FORCED
.RI    LUWQ    ?????   FORCED

.RD    _UNIT NAME                   _                                     DD WD MD

.RI    PUFS    ?????   FORCED
.RI    PUIO    ?????   FORCED
.RI    PUSW    ?????   FORCED
.RI    PUWQ    ?????   FORCED

.RD    _GENERAL SUMMARY             _      PAGING BEFORE                  DT WT MT

.RI    MTAU            FORCED

.RD    _TAPE DRIVE                  _                                     DT WT MT

.RI    MTIO    ??????  FORCED

.RD    _END REPORTS                 _
```

W

Figure 1-19 (continued)

1.5.2 Examples Of Some Common Changes To The RFD File

Changes which are commonly made to the RFD file include adjusting threshold limits (especially for testing for disk free space), revising the comments that get printed when thresholds get exceeded, and forcing certain items or subitems to be always printed.

Example 1 - Changing the Free Space Watchdog Limit:

By default, any pack that has less than ten percent free space will appear on the Utilization Reports as being under the acceptable watchdog limit. Typically, page two of these reports will contain the pack name, the number of hours when the free space was less than 10% (using P's and N's) and the message "DELETE UNNECESSARY FILES". For most packs this 10% free space limit is acceptable. However, if a particular pack, perhaps DSKX, consistently has 5% free space, it is not necessary or useful to see DSKX show up every day with every hour flagged with asterisks. Asterisks generally mean that this is something important to look at or a potential problem to solve. You can make a couple of quick edits to the RFD file to change the free space threshold to perhaps 3%. To do this the "ITEM DESCRIPTION SECTION" is changed to specifically add DSKX with the new limits. The item code for logical unit free space is LUFS. In the RFD file, insert another line immediately after the .ID LUFS line. This new line should have the same format as the LUFS line with the "wild card" question marks in the subitem code field changed to DSKX0. The easiest way to do this is to copy the LUFS line under itself. You will end up with two LUFS lines. Then change the question marks on the second line to the pack name. Next, change the limits. The short term/long term limit fields respectively are "<0000010.,<0000010." on the LUFS line. On the new LUFS DSKX0 line change the fields to "<0000003.,<0000003." This will make the short term/long term limits both 3%. Values for this pack will be flagged now only if it has less than 3% free space. Any number of packs can be added to the RFD in this way. There are only two cautions: if a list of packs is added, they must be in alphabetical order; the question marks on the LUFS line serve as a "wild card" that will allow the line to apply to all packs not specified by name. Do not accidentally delete the "wild carded" line while editing.

Example 2 - Changing the Comment Field:

Another change that can be made is to revise the comment line that appears when an item exceeds its limits. This can be useful, for example, when DSKX is owned by a particular user group that wants to know when their pack has less than 3% free space. After DSKX0 has been specified in the LUFS list, the "DELETE UNNECESSARY FILES" text on the LUFS DSKX0 line can be change to "NOTIFY USER GROUP". With that edit, whenever the hourly average for DSKX is 3% or less, the notify message will appear. For the other packs, the old delete message will still be printed.

Example 3 - Forcing Specific Items/Subitems to Print:

The REPORT DESCRIPTION SECTION of the RFD controls which items are designated as "KEY" in the reports (i.e., always appear) and which are designated as "OTHER" (i.e., appear only if they are flagged as over or under their limits). A simple change, as an example, would be to remove MTAU (number of MTA's assigned) from the key item list and replace it with something more interesting such as CPCO, context switches per second. In Figure 1-19, the second section for key utilization items immediately under the ".C *******REPORT DESCRIPTION SECTION" controls the key items for the Weekly/Monthly Utilization Reports. Simply change the code MTAU to CPCO. The number of context switches will always appear now as a key item.

1.6 PROCEDURE FOR RUNNING AMAR.CTL

1.6.1 Overview Of AMAR.CTL

This stream runs daily. It takes the data which has been collected by xxxxDC, inputs it into the system AMAR database, creates summary records, deletes expired records from the database, performs housekeeping on the AMAR area, and creates the automatic reports. Reference the Appendix called "System AMAR Batch Stream - AMAR.CTL" for a sample of the stream and step descriptions.

1.6.2 Resource Requirements

To run System AMAR on a continuous basis, you will need an OPSER subjob and a permanently mounted disk area with approximately 3.2K blocks for program and raw file storage. The size of your system AMAR database will probably vary from 3K - 6K blocks depending on the type and amount of data retained. You will need to reserve space for the original database plus a backup copy, preferably on permanent storage. See also the Appendix called "Installation and Resource Requirements".

The minimum supporting software required is COBOL 12A.

1.6.3 Submission

The stream normally resubmits itself after running each night. If both the System AMAR database and its backup are corrupt (parity errors, etc.), the stream stops and the operator must restart it after restoring the database from a good copy. The stream should always be restarted from the beginning. There should always be an AMAR.CTL in the submit queue, set to run /AFTER:1:0:0.

1.6.4 Restart Procedure

If a system crash occurs while the stream is running, the stream should automatically restart at the proper checkpoint. If the crash occurs while AMARIP or AMARUP is running, the database will become corrupted. The stream will test for a corrupted database and automatically restore from the disk backup, if necessary. The only time the stream should need manual restarting is if the submit queue were destroyed or if both the primary and backup System AMAR databases are corrupted.

Blank Page

CHAPTER 2

WORKLOAD AMAR

2.1 MAJOR FEATURES

Features of Workload AMAR include:

1. Low overhead continuous data collection.

2. An historical workload database featuring:

   a. Separate files for different time periods to minimize I/O.

   b. Detail data retained in compressed form.

   c. Automatic deletion of old data.

3. Flexible reporting programs featuring:

   a. Grouping of resource usage by one or more items such as PPN, project, charge number, program name, batch vs. timesharing, scheduler class, and individual session.

   b. Sorting of detail lines by above items or more likely by resource usage to highlight heavy users.

   c. Optional suppression of insignificant detail lines.

   d. Selection of time period to be reported and whether to describe it in a single report or a series of interval reports.

4. Single daily batch stream which will:

   a. Provide useful reports automatically (daily, weekly, monthly).

   b. Prevent buildup of data files on disk.

2.2 OVERVIEW OF WORKLOAD AMAR

The four functions of Workload AMAR (also referred to as the workload system) are data collection, preprocessing, database management, and reporting. The four functions are performed by four separate programs which are described briefly below. Refer to Figure 2-1 for an overview of program and data flow.

2.2.1 Data Collection

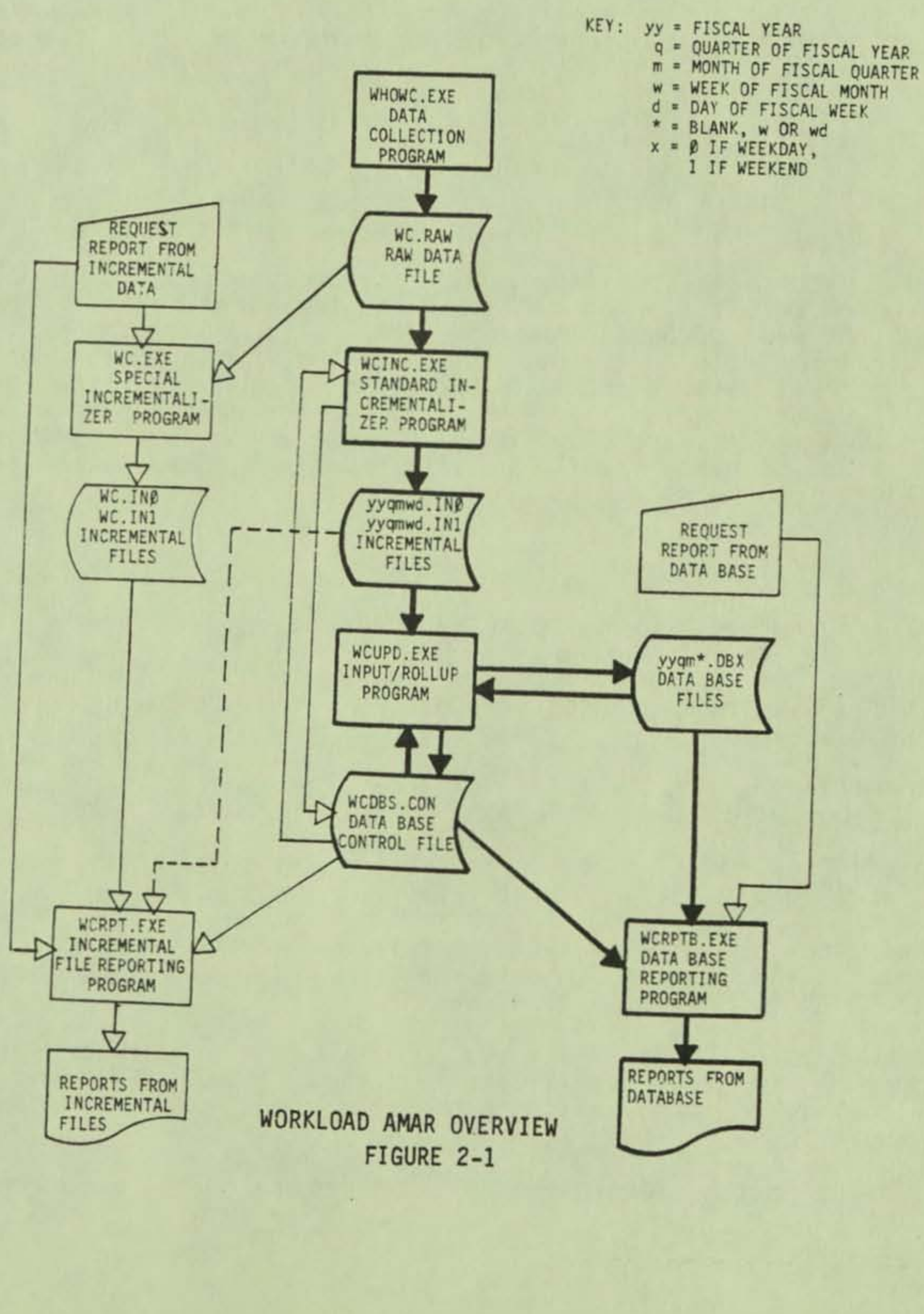
WHOWC collects resource utilization, identification, and state data about each job on the system at user-specified "checkpoint intervals", typically every 5 minutes. WHOWC runs 24 hours a day, preferably as an OPSER subjob.

2.2.2 Preprocessing

WCINC converts the data from "checkpoint records" collected by WHOWC into "incremental usage records" required as input to WCUPD. WCINC actually produces two output files: one which contains incremental usage by job and one which summarizes incremental usage by checkpoint interval. WCINC is normally run once a day. The program WC is a special version of WCINC which will let you preprocess current data without disrupting the normal cycle.

2.2.3 Database Management

WCUPD performs database management including input, rollup, and deletion. Data for each day is included in the database as a separate file with a section for each hour of the day. Daily data for each weekday is rolled up into a weekly weekday file which has a section for each "typical hour". Daily data for each Saturday, Sunday or holiday in a week is rolled into a similar weekly weekend file. Weekly files are rolled into monthly files. When there are more than the desired number of daily, weekly, or monthly files, the oldest files are automatically deleted.



WORKLOAD AMAR OVERVIEW
FIGURE 2-1

2.2.4 Reporting

The main reporting program, WCRPTB, uses the workload database to report on resource utilization during user-specified reporting intervals, which are normally an integral number of hours or "typical hours". Resource usage may be summarized by user-specified identification and state data. It is also sorted by those keys and/or by amount of resources. Weekly and monthly reports are generated automatically by the single daily batch stream. WCRPT is a variation of WCRPTB which is used to report from preprocessed data rather than the database.

2.3 ANNOTATED SAMPLE REPORTS

This section contains four samples of the workload reports produced by the standard daily stream WCRPTB.CTL. The first report is heavily annotated to help you understand the format which is common to all daily workload reports. The minor difference in format for other workload reports (weekly and monthly) is in the first line of the subreport header, which is annotated in the second sample report. Workload AMAR reports are often used in conjunction with the System AMAR Utilization and Trend Analysis Reports. These latter reports show activity on the system as a whole.

2.3.1 Hourly Report Showing Major CPU Users

Figure 2-2 shows the beginning of WCDY0.RP1, the standard daily report with hourly subreports.

This report is the first place to look for workload reasons for problems reported in the System AMAR Daily Utilization Report.

Each detail line shows resource usage by a particular job running a particular program. Major CPU users are at the top of the list.

The first page in Figure 2-2 shows the report header box and the first three subreports. The subreports start at approximately midnight and run to 3:00 AM (00:00:02 to 03:00:02).

The second page of Figure 2-2 shows portions of two subreports from a busy period. The subreports start at approximately 15:00 and run to 17:00. Intervening and trailing subreports have been omitted from this example.

The following notes refer to the circled numbers on the sample report:

REPORT HEADER BOX:

1. This box surrounded by asterisks appears at the top of the first page of each workload report. Look for this box when searching for a particular report in a series of workload reports printed without separator pages.

2. "AMAR WORKLOAD REPORT" always appears in the report header box.

3. Site description (up to 90 characters) comes from the file WCDBS.CON. This description is set up at installation time.

4. System code (4 characters) is used to identify workload data as belonging to a particular system. It comes from WCDBS.CON and is included in all workload database files. The system code should be the same as the code used in the System AMAR database.

5. Report description (up to 90 characters) is entered during WCRPTB dialogue (in the batch stream or on-line).

6. Input filename indicates the fiscal period covered.

7. Parentheses enclose the explanation of the input filename.

SUBREPORT HEADINGS:

8. The first line of this subreport heading is typical for daily reports. The first line of a weekly or monthly subreport heading is different. (See the next sample report for an example.)

9. Start of report interval (time, day of the week, date).

10. End of report interval.

11. Length of report interval (HH:MM:SS).

12. Percent of interval measured for workload characterization. Only measured time is used to compute resource usage rates.

13. List of items whose values are held constant to determine what goes into each line of detail data. In this case, JOB number and LOGIN date/time identify a unique session and PRGRM identifies a portion of that session when a particular program was running.

14. Sort key. In this case, the detail lines have been sorted in descending order by percentage of processor use (CPU%), to draw attention to major CPU users at the top of the list.

15. Cutoff criteria. In this case, each detail line representing less than 0.90% of the CPU was suppressed.

INTERVAL TOTALS LINE (See page 2 of the example.):

16. The INTERVAL TOTALS line is the summary of resource usage during the reporting interval.

17. The total at the top of this column indicates average number of jobs in use during this reporting interval.

18. The average job size in pages in the INTERVAL TOTALS line is a weighted average of job sizes during the period. (Other numbers in the INTERVAL TOTALS line are ordinary totals).

DETAIL LINES:

19. Job number (as would be reported by SYSTAT).

20. Fraction of the reporting interval that the specified program appeared to be in use. Since this example is broken out by program within session, this value will be 1.0 if the session spanned the entire interval and the same program ran throughout the interval. A value of 0.4 would indicate that the program was running during 40% of the interval.

21. 1 = job logged in during interval.

22. 0 = job was in progress.

23. 1 = job logged out during interval.

24. 0 = job remained logged in.

25. Project/programmer number (PPN).

26. Charge number (on systems where it is used).

27. Program name. This is the program that was running at the end of each of the samples rolled into this detail line. It is probably the program that used most of the resources, but it may not be if only one or two samples are represented. In this example, one sample would represent about 0.1 in the AVG JOBS column.

28. The average job size is computed by dividing kilocore seconds by CPU seconds (and multiplying by 2 pages / K). Hence, it is the average size while using the processor. If the job size is as expected for the program, it tends to confirm that it was that program which used the resources. Note that if a detail line represents exactly 0% of the CPU, the average job size will be reported as 0.0 pages.

29. CPU% is CPU seconds charged to this detail line divided

by seconds of measured time during the interval (times 100 to make it a percentage). This measure is based on the way the system charges processor usage to a job, with whatever shortcomings that may have. For example, a KL10 using EBOX/MBOX accounting might only charge users for about 70% of the time they use the processor. Also, resources used between the last time WHOWC checkpoints a job and when the job logs out will not be captured. Thus, all resource totals will usually tend to be somewhat low when compared with the corresponding data extracted from accounting (USAGE) files. On the other hand, resource totals may occasionally be high, because WHOWC checkpoints more frequently than DAEMON. Note also that on a dual processor, users may use the equivalent of more than 100% of a processor.

30. UUO's are Monitor calls. It is common for a job to use one UUO for each disk block read or written, but some jobs use many other UUO's.

31. READS/SECOND is short for disk blocks read per second which is what is actually reported.

32. WRITES/SECOND is similar to READS/SECOND.

33. The scheduler class is normally reported in this column. If the job was running in a high priority queue, the high priority queue number is reported, flagged with a minus sign to distinguish it from an ordinary scheduler class.

34. LOGIN category. Login category may be used to group users and then limit the number of simultaneous users in the group who can log in. Note: This value is displayed in octal.

35. T = timesharing.

36. B = batch.

37. Job state. (SL=sleep, R2=run, TI=teletype input wait, etc. See the SYSTAT description in the Operating System Commands manual.) Note that when the job was observed in more than one state during the interval, question marks show that the state varied.

38. Terminal number where the job was running.

39. Node to which the job was attached.

40. Line number on that node. Note: This value is displayed in octal.

41. Day on which the job logged in, relative to the end of the report interval. For example, 0 means the job logged in

during the day in which the interval ended, -1 means the previous day.

42. Time of day when the job logged in.

SUBTOTAL LINES:

43. When some but not all of the detail lines have been suppressed because of cutoff criteria, this line is included to summarize the detail lines printed.

44. This line is included to summarize any suppressed detail lines.

```
                                              1
*****************************************************************************************
*                                                                                       *
*                                    AMAR WORKLOAD REPORT 2                             *
*                                                                            4          *
*   SITE:  <Put Any Title Here> 3                                    SYSTEM:  PATH      *
*   REPORT DESCRIPTION:  HOURLY REPORT BY PROGRAM AND JOB 5                             *
*   INPUT FILE:  823242.DBO 6 (FISCAL YEAR: 82  QUARTER: 3  MONTH: 2  WEEK: 4  DAY: 2  MONDAY) 7
*                                                                                       *
*****************************************************************************************

                     9                                   10                               11                 12
8 FROM:  0: 0: 2 ON MONDAY   15-FEB-82   TO:  1: 0: 2 ON MONDAY   15-FEB-82   INTERVAL:  0:59:59   MEASURED: 100%
GROUPED BY: 13 JOB   PRGRM LOGIN
SORTED BY: 14 CPU%     CUTOFF:  0.90% OF CPU 15
JOB  AVG  LOG LOG ------PPN----- CHARGE PRGRM  PAGES   CPU%   UUOS  READS WRITES SCD CT B STATE  TTY   NODE--LINE LOGIN AT
 #   JOBS  IN OUT    PROJ  PROG  NUMBER  NAME  (AVG)          /SEC  /SEC   /SEC CLS                               DAY TIME
     21.0   1   0 ******INTERVAL TOTALS******  24.4     0.4    4.4   0.2    0.2


FROM:  1: 0: 2 ON MONDAY   15-FEB-82   TO:  2: 0: 1 ON MONDAY   15-FEB-82   INTERVAL:  0:59:59   MEASURED: 100%
GROUPED BY:   JOB   PRGRM LOGIN
SORTED BY:    CPU%     CUTOFF:  0.90% OF CPU
JOB  AVG  LOG LOG ------PPN----- CHARGE PRGRM  PAGES   CPU%   UUOS  READS WRITES SCD CT B STATE  TTY   NODE--LINE LOGIN AT
 #   JOBS  IN OUT    PROJ  PROG  NUMBER  NAME  (AVG)          /SEC  /SEC   /SEC CLS                               DAY TIME
     22.0   0   0 ******INTERVAL TOTALS******  89.1     1.2   10.5   1.6    1.2


FROM:  2: 0: 1 ON MONDAY   15-FEB-82   TO:  3: 0: 2 ON MONDAY   15-FEB-82   INTERVAL:  1: 0: 0   MEASURED: 100%
GROUPED BY:   JOB   PRGRM LOGIN
SORTED BY:    CPU%     CUTOFF:  0.90% OF CPU
JOB  AVG  LOG LOG ------PPN----- CHARGE PRGRM  PAGES   CPU%   UUOS  READS WRITES SCD CT B STATE  TTY   NODE--LINE LOGIN AT
 #   JOBS  IN OUT    PROJ  PROG  NUMBER  NAME  (AVG)          /SEC  /SEC   /SEC CLS                               DAY TIME
     22.0   0   0 ******INTERVAL TOTALS******  22.7     0.4    7.7   0.1    0.1

                                              •
                                              •
                                              •
```

Figure 2-2

54.5 30 25 ***SUBTOTALS AFTER CUTOFF** 46.6 12.6 137.6 24.2 3.6

FROM: 15: 0: 2 ON MONDAY 15-FEB-82 TO: 16: 0: 3 ON MONDAY 15-FEB-82 INTERVAL: 1: 0: 0 MEASURED: 100%

GROUPED BY: JOB PRGRM LOGIN

SORTED BY: CPU% CUTOFF: 0.90% OF CPU

JOB #	AVG JOBS	LOG IN	LOG OUT	------PPN----- PROJ PROG	CHARGE NUMBER	PRGRM NAME	PAGES (AVG)	CPU%	UUOS /SEC	READS /SEC	WRITES /SEC	SCD CLS	CT	B	STATE	TTY	NODE	LINE	LOGIN AT DAY	TIME
	70.[illegible]	25	24	******INTERVAL	TOTALS******		54.[illegible]	36.[illegible]	418.5	29.[illegible]	18.[illegible]									
59	0.3	0	1	[15323112157]	555267	DIRECT	34.4	6.3	13.8	0.0	0.0	4	0	T	?? ??	TTY37	NODED	37	0	14:40
59	0.5	0	0	[15323112157]	555267	BACKUP	22.0	5.0	27.3	0.0	0.0	4	0	T	??	TTY37	NODED	37	0	14:40
67	0.6	1	0	[15375112420]	555571	1022FO	144.4	2.5	1.8	0.6	0.1	4	0	T	??	TTY54	NODED	54	0	15:23
43	1.0	0	0	[1 2]	133085	DLLSPL	29.9	2.4	161.4	0.1	0.0	0	54	T	?? ??	TTY103	NODED	103	0	8:47
47	0.8	0	0	[12400112167]	750204	1022	49.9	1.4	8.7	2.9	0.0	4	0	T	??	TTY114	NODEM	6	0	12:27
32	0.6	0	0	[20320112161]	555040	DIRECT	28.2	1.3	11.4	2.6	0.4	4	0	T	??	TTY77	NODED	77	0	14:30
7	1.0	0	0	[1 2]		MIC	25.6	1.1	46.0	0.1	0.0	0	0	T	??	DET313		0	-4	23:18
27	0.3	0	1	[15152 45010]	216300	BACKUP	21.9	1.1	5.9	0.0	0.0	5	0	B	??	PTY23		0	0	15:27
52	0.1	0	0	[15152 45010]	216300	BACKUP	23.8	1.1	13.8	0.0	10.8	5	0	B	R2	PTY24		0	0	15:54
44	1.0	0	0	[12400132153]	750240	1022	92.8	1.0	6.6	2.4	0.3	4	0	T	??	TTY75	NODED	75	0	11:11
61	0.1	0	0	[15152 45010]	216300	CBL74	78.9	0.9	1.2	0.6	0.6	4	0	T	R2	TTY16	NODED	16	0	13:16
	6.2	1	2	***SUBTOTALS	THRU CUTOFF***		45.9	24.2	298.0	9.4	12.1									
	64.0	24	22	***SUBTOTALS	AFTER CUTOFF**		72.0	12.7	120.6	20.2	6.7									

FROM: 16: 0: 3 ON MONDAY 15-FEB-82 TO: 17: 0: 2 ON MONDAY 15-FEB-82 INTERVAL: 0:59:59 MEASURED: 100%

GROUPED BY: JOB PRGRM LOGIN

SORTED BY: CPU% CUTOFF: 0.90% OF CPU

JOB #	AVG JOBS	LOG IN	LOG OUT	------PPN----- PROJ PROG	CHARGE NUMBER	PRGRM NAME	PAGES (AVG)	CPU%	UUOS /SEC	READS /SEC	WRITES /SEC	SCD CLS	CT	B	STATE	TTY	NODE	LINE	LOGIN AT DAY	TIME
	58.5	25	50	******INTERVAL	TOTALS******		46.0	41.1	251.1	36.9	52.9									
28	0.5	0	0	[15152 45010]	216300	ISAM	14.0	13.7	20.5	10.1	10.2	4	0	T	R2	TTY55	NODED	55	0	15:03
6	0.3	0	0	[15152112157]	216300	0020CC	119.6	3.5	2.2	1.1	0.3	4	0	T	?? ??	TTY17	NODED	17	0	8:15
52	0.4	0	1	[15152 45010]	216300	BACKUP	24.0	2.8	36.8	0.0	29.4	5	0	B	??	PTY24		0	0	15:54
49	0.2	0	0	[20320112136]	555040	COBOL	57.3	2.5	1.0	0.1	0.1	4	0	T	R2	TTY76	NODED	76	0	9:14
67	0.6	0	1	[15375112420]	555571	1022FO	141.8	1.8	1.3	0.4	0.1	4	0	T	??	TTY54	NODED	54	0	15:23
49	0.8	0	0	[20320112136]	555040	SED	68.0	1.6	9.8	0.9	0.3	4	0	T	??	TTY76	NODED	76	0	9:14
32	0.2	1	1	[275 43226]	133105	DIP	32.0	1.4	14.0	0.1	0.0	4	1	T	??	TTY121	NODEM	23	0	16:21
59	0.4	0	0	[15152112441]	224200	BACKUP	20.6	1.0	7.4	0.1	1.9	4	0	T	?? ??	TTY35	NODED	35	0	16:08
44	1.0	0	0	[12400132153]	750240	1022	92.6	1.0	7.2	2.6	0.2	4	0	T	??	TTY75	NODED	75	0	11:11
	4.4	1	3	***SUBTOTALS	THRU CUTOFF***		45.7	29.4	100.1	15.4	42.5									
	54.1	24	47	***SUBTOTALS	AFTER CUTOFF**		46.8	11.7	151.0	21.4	10.4									

Figure 2-2 (continued)

Blank Page

2.3.2 Program Name Report

Figure 2-3 is a sample of the default report WCWK0.RP2 and shows weekly resource usage by program on the basis of typical 8-hour shifts. This report highlights programs which may be candidates for optimization or rescheduling. The detail lines are sorted by percent of CPU used with the heaviest consumers at the top of the list. For example, during prime time, the program 1022 (K) was the heaviest user of the processor. This program had, on average, 9.4 (L) simultaneous users and accounted for approximately 376 hours of connect time over the five days of prime shift. Connect time equals the average number of jobs (9.4 (L)) multiplied by the interval measured (40.0(N)). A CPU cutoff (J) of 0.1% is used to suppress printing of any lines containing programs which used less than 0.1% of the CPU.

Note the first line of the subreport header (A). It is different from the corresponding line on a daily report. It first tells the start time (B) and end time (C) of the "typical period" described. Then it tells what type of day (D) is included. This should be read "WEEKDAYS MINUS HOLIDAYS". The other possibility is "WEEKENDS (+ HOLIDAYS)". Next it tells the first day (E), last day (F), and number of days (G) included. A glance at a fiscal calendar will tell you if all the days have been included. Finally, it tells how much of the time was measured: first as a percentage (H) of the typical period, then as a total number of hours (I). This number of hours can be used to compute resource totals from the reported rates.

Note: Question marks in the PPN and charge number fields indicate that the values of these items varied; i.e., more than one user ran the program under more than one charge number.

As another example, the ISAM program was the second heaviest user during prime shift, even though it was being run only 20% (M) of the time.

The third heaviest CPU user was COBOL. Note that here it used 2.8% of the CPU although it ran only 20% of the time. That means that while it was running it used about 14% of the CPU (2.8% divided by 20%).

```
**********************************************************************************************************
*                                                                                                        *
*                                       AMAR WORKLOAD REPORT                                             *
*  SITE:  <Put Any Title Here>                                                                           *
*                                                                                                        *
*  REPORT DESCRIPTION:  WEEKLY REPORT BY PROGRAM (TYPICAL 8-HOUR SHIFTS)            SYSTEM:  PATH        *
*                                                                                                        *
*  INPUT FILE:  82324 .DBO    (FISCAL YEAR: 82   QUARTER: 3   MONTH: 2   WEEK: 4   WEEKDAYS)             *
*                                                                                                        *
**********************************************************************************************************
```

A B 0: 0 - C 8: 0 WEEKDAYS D (-HOLIDAYS) FROM: MON E 15-FEB-82 TO: FRI F 19-FEB-82 G (5 DAYS) MEASURED: H 99% = I 39.77 HOURS

GROUPED BY: PRGRM

SORTED BY: CPU% CUTOFF: 0.10% OF CPU

JOB #	AVG JOBS	LOG IN	LOG OUT	PPN PROJ	PPN PROG	CHARGE NUMBER	PRGRM NAME	PAGES (AVG)	CPU%	UUOS /SEC	READS /SEC	WRITES /SEC	SCD CLS	CT	B	STATE	TTY	NODE--LINE	LOGIN AT DAY TIME
	22.2	95	38	******INTERVAL TOTALS******				48.9	1.2	11.3	2.2	0.5							
	0.0	1	1	[1	2]	133085	BACKUP	23.8	0.2	1.8	1.1	0.0	0	54	T	??		NODED	
	0.1	14	0	[124001?2???	]	7502??	1022	64.0	0.2	0.7	0.2	0.0	4	0	T	??		NODE?	
	0.0	5	5	[2104	51335]	133070	5955LS	92.8	0.1	0.3	0.1	0.1	5	1	B	??			
	0.1	20	6	***SUBTOTALS THRU CUTOFF***				55.2	0.5	2.8	1.5	0.1							
	22.1	75	32	***SUBTOTALS AFTER CUTOFF**				44.9	0.7	8.5	0.7	0.4							

8: 0 - 16: 0 WEEKDAYS (-HOLIDAYS) FROM: MON 15-FEB-82 TO: FRI 19-FEB-82 (5 DAYS) MEASURED: 100% = N 40.00 HOURS

GROUPED BY: PRGRM

SORTED BY: CPU% CUTOFF: J 0.10% OF CPU

JOB #	AVG JOBS	LOG IN	LOG OUT	PPN PROJ	PPN PROG	CHARGE NUMBER	PRGRM NAME	PAGES (AVG)	CPU%	UUOS /SEC	READS /SEC	WRITES /SEC	SCD CLS	CT	B	STATE	TTY	NODE--LINE	LOGIN AT DAY TIME
	58.5	1255	1079	******INTERVAL TOTALS******				57.3	30.7	204.6	62.3	23.8							
L	9.4	296	309	[???????????	]	??????	1022 K	59.3	8.0	37.1	23.6	2.5				?? ??		?????	
M	0.2	3	1	[???????????	]	??????	ISAM	15.4	3.4	5.1	3.5	3.4		O		?? ??		?????	
	0.2	5	7	[???????????	]	??????	COBOL	58.9	2.8	1.3	0.3	0.2	4	O	T	?? ??		?????	
	4.1	95	82	[????????????	]	??????	SOS	51.7	1.6	17.2	1.9	1.0	4		T	?? ??		?????	
	1.2	19	15	[???????????	]	??????	DIP	33.8	1.4	27.7	6.0	5.8				?? ??		?????	
	0.7	22	22	[1????112???	]	75????	1022FO	143.5	1.3	1.2	0.6	0.0	4	O	T	?? ??		?????	
	1.5	77	59	[????????????	]	??????	DIRECT	44.3	1.0	6.7	2.4	0.6				?? ??		?????	
	2.6	105	107	[????????????	]	??????	PIP	47.1	0.9	4.7	1.7	0.5				?? ??		?????	
	0.2	2	13	[???????????	]	??????	BACKUP	23.8	0.9	8.8	3.8	1.6				?? ??		?????	
	0.6	24	27	[????????????	]	??????	WHO	59.5	0.6	1.8	0.1	0.0				?? ??		?????	
	0.0	1	1	[15152112157	]	216300	0010DE	64.0	0.6	0.0	0.0	0.0	5	O	B	R2			
	1.1	11	4	[???????????	]	??????	MIC	31.4	0.5	15.5	0.3	0.1			T	?? ??		?????	

Figure 2-3

```
  1.1  14  15 [ ???????????] ?????? SED      46.8   0.5   5.0   0.3   0.1        T ?? ??     ?????
  0.0   1   0 [   275 42203] 462300 0004HR  257.7   0.4   1.1   1.6   0.1   3  2 B R2
  1.0   1   1 [     1     2]        FILDAE   14.0   0.3   1.8   0.2   0.0  -1  0 T ?? ??
  0.1   3   2 [ 1515277????] 216700 COBABY   74.3   0.3   2.7   3.7   0.2      0   ?? ??     ?????
  0.0   0   1 [   275 42203] 216300 0100PT   79.6   0.3   1.9   2.1   1.3   3  2 B ??
  2.7   5   6 [     1     2] 133077 LPTSPL   24.9   0.2   1.7   0.2   0.0   0 54 T ?? ??
  1.6  11  18 [ ???????????] ?????? S        48.5   0.2   2.2   0.2   0.1        T ^C ??     ?????
  1.2  12   8 [ 15323112177] 555267 1005PT  140.6   0.2   1.1   1.0   0.4   4  0 T ?? ??     ?????
  0.6  27  27 [ ???????????] ?????? TECO     39.4   0.2   1.0   0.1   0.0        T ?? ??     ?????
  0.1   3   1 [ 20320112161] 555040 ROWMON   62.5   0.2   1.8   0.2   0.1   4  0 T ?? ??     NODED
  0.2   8  11 [ ???????????] ?????? BASIC    59.6   0.2   0.5   0.1   0.0   4    T ?? ??     ?????
  1.0   1   1 [     1     2] 133025 BATCON   27.4   0.2   4.7   0.1   0.1   0 54 T ?? ??
  0.0   1   0 [ 15152112157] 216300 LOOK     36.0   0.2   0.1   0.1   0.0   4  0 T ??        NODED
  1.0   0   1 [     1     2] 133010 QUASAR   26.8   0.2   1.7   0.0   0.1   0 54 T ?? ??
  2.2  20  36 [????????????] ?????? QUEUE    63.0   0.2   1.2   0.3   0.1          ?? ??     ?????
  0.0   0   0 [ 15152 47???] 216700 CBL74    78.1   0.2   0.3   0.1   0.1   4  0 T ?? ??     NODED
  0.1   1   0 [ 20320112143] 463395 PUSH     78.6   0.2  12.4   0.0   0.0   4  0 T ?? ??     NODED
  0.0   0   0 [ 15152112157] 216300 0020CC  115.2   0.2   0.1   0.1   0.0   4  0 T ?? ??     NODED
  0.0   0   1 [ 15152112403] 555267 0080TN   90.8   0.1   0.9   0.2   1.0   4  0 T ??        NODED
  1.0   1   1 [     1     2]        DAEMON   22.0   0.1   1.0   0.1   0.1      0 T ?? ??
  2.9   6   5 [ ???????????] ?????? OPSER    12.0   0.1   4.7   0.2   0.1        T ?? ??     ?????
  0.1   2   2 [  7700112151] 133100 FRCST    30.0   0.1   0.1   0.0   0.0   4  0 T ??        NODEN
  0.2  34  14 [ ???????????] ?????? UMOUNT   31.7   0.1   1.1   0.3   0.2          ?? ??     ?????

 38.9 811 798 ***SUBTOTALS THRU CUTOFF***    55.7  28.1 176.4  55.4  20.3

 19.5 444 281 ***SUBTOTALS AFTER CUTOFF**    74.0   2.6  28.3   6.9   3.5


16: 0 - 24: 0  WEEKDAYS (-HOLIDAYS)    FROM: MON 15-FEB-82  TO: FRI 19-FEB-82  ( 5 DAYS)    MEASURED:  89% =  35.65 HOURS

GROUPED BY:   PRGRM

SORTED BY:    CPU%    CUTOFF:  0.10% OF CPU

JOB  AVG  LOG LOG ------PPN----- CHARGE PRGRM  PAGES   CPU%  UUOS  READS WRITES SCD CT B STATE  TTY   NODE--LINE LOGIN AT
 #   JOBS  IN OUT   PROJ  PROG   NUMBER  NAME  (AVG)         /SEC   /SEC   /SEC CLS                                DAY TIME

 32.4 261 450 ******INTERVAL TOTALS******    43.0  11.1  94.1  36.6  14.7

  0.2   0   2 [ 1??????????] ?????? ISAM     15.3   2.7   3.8   2.1   2.1   4  0 T ??        NODE?
  0.2  12  14 [ ???????????] ?????? BACKUP   27.4   1.9  24.6  16.7   0.9          ?? ??     ?????
  2.0  36  95 [ ???????????] ?????? 1022     56.9   1.0   6.2   3.2   0.2   4  0 T ?? ??     ?????
  0.0   0   0 [     1     2] 680160 DDRPI    94.6   0.7   4.4   3.6   3.6   4  2 T ??        NODED
  0.0   1   0 [     1     2] 133085 FSCOPY   47.7   0.4   1.3   5.2   5.0   5 54 B ??
  0.1   0   0 [  77001121??] 133085 VTTECO   29.9   0.4   6.9   0.0   0.0   0 54 T ??        NODEM
  0.8  12  22 [ ???????????] ?????? SOS      53.5   0.4   3.4   0.4   0.2        T ?? ??     ?????
  1.0   3   2 [210400 62165] 133000 NTRACK   69.3   0.3   8.6   0.0   0.0   5  0 B ?? ??
  0.7  20  39 [ ???????????] ?????? PIP      82.0   0.3   1.2   0.3   0.1          ?? ??     ?????
  0.3  11  10 [ ???????????] ?????? DIP      36.4   0.2   3.3   0.9   0.6          ?? ??     ?????
  2.9  11   7 [     1     2] 133077 LPTSPL   24.5   0.2   1.0   0.2   0.0   0 54 T ?? ??
  0.1   0   5 [ 1????112???] 555571 1022FO  146.6   0.2   0.3   0.3   0.0   4  0 T ??        NODE?
  0.3  11  17 [ ???????????] ?????? DIRECT   37.7   0.1   1.1   0.4   0.2        T ?? ??     ?????
  1.0   3   2 [     1     2] 133025 BATCON   22.7   0.1   4.6   0.0   0.0   0 54 T ??
  0.1   2   3 [ 20320112153] 555040 VTED     86.6   0.1   1.6   0.0   0.0   4  0 T ??        NODED
  0.2   3   8 [ ???????????] ?????? SED      52.9   0.1   1.2   0.1   0.0        T ?? ??     NODED
  0.0   0   1 [  7700112151] 133100 FRCST    30.0   0.1   0.1   0.0   0.0   4  0 T T?        NODEN
  0.0   1   1 [ ????112????] ?????? COBOL    57.4   0.1   0.3   0.0   0.0   4  0 T ??        NODE?
                                        .
                                        .
                                        .
```

Figure 2-3 (continued)

Blank Page

2.3.3 Batch Vs. Timesharing Report

Figure 2-4 is a sample of the default report WCDY0.RP3 and shows daily resource usage by all batch jobs (A) combined vs. all timesharing jobs combined on an 8-hour (B) shift basis.

The prime shift (8:00-16:00) is about three times busier than the evening shift (16:00 to midnight). In both cases, the bulk of the CPU load comes from timesharing jobs (C).

Note that the average batch job (E) used 2.9% of the CPU during the day. The average timesharing job used .6%. To find the average percent used by a job, divide the CPU% field by the AVG JOBS field. While batch jobs can be heavy CPU users, the amount of CPU time that batch and timesharing jobs are allowed to consume over a given period can be regulated, to some extent, by the system scheduler. By using the scheduler, it is possible to favor timesharing jobs over batch work during the day and vice versa at night.

Also, note that although an average of 28.4 timesharing jobs were reported during the evening shift, a little more than two thirds are [1,2] jobs running DAEMON, LPTSPL, etc. The number of [1,2] jobs can be determined from a PPN report such as WCMN0.RP4 (Figure 2-5).

```
*****************************************************************************************
*                                                                                       *
*                                   AMAR WORKLOAD REPORT                                *
*  SITE:  <Put Any Title Here>                                                          *
*  REPORT DESCRIPTION:  SHIFT REPORT BY BATCH VS. TIMESHARING            SYSTEM:  PATH  *
*  INPUT FILE:  823242.DBO   (FISCAL YEAR: 82  QUARTER: 3  MONTH: 2  WEEK: 4  DAY: 2  MONDAY) *
*****************************************************************************************

FROM:   0: 0: 2  ON  MONDAY   15-FEB-82   TO:   8: 0: 2  ON  MONDAY   15-FEB-82   INTERVAL:   7:59:59   MEASURED: 100%
GROUPED BY:   BATCH A
SORTED BY:    CPU%

JOB  AVG  LOG LOG ------PPN----- CHARGE PRGRM  PAGES   CPU%  UUOS  READS WRITES SCD CT B STATE  TTY   NODE--LINE LOGIN AT
 #   JOBS  IN OUT   PROJ  PROG   NUMBER  NAME  (AVG)         /SEC   /SEC   /SEC CLS                               DAY TIME
     22.1  13   5 ******INTERVAL TOTALS******   52.6    0.8  11.8    0.6    0.4
     20.1   7   1 [????????????] ?????? ??????  30.5    0.6   7.9    0.3    0.2          T ?? ??  ?????? ?????
      2.0   6   4 [?????? ?????] ?????? ?????? 101.3    0.3   3.9    0.3    0.2          B ??     PTY??           0

FROM:   8: 0: 2  ON  MONDAY   15-FEB-82   TO:  16: 0: 3  ON  MONDAY   15-FEB-82   INTERVAL:   B 8: 0: 0   MEASURED: 100%
GROUPED BY:   BATCH
SORTED BY:    CPU%

JOB  AVG  LOG LOG ------PPN----- CHARGE PRGRM  PAGES   CPU%  UUOS  READS WRITES SCD CT B STATE  TTY   NODE--LINE LOGIN AT
 #   JOBS  IN OUT   PROJ  PROG   NUMBER  NAME  (AVG)         /SEC   /SEC   /SEC CLS                               DAY TIME
     56.8 224 183 ******INTERVAL TOTALS******   57.0   37.9 265.7   41.3   18.5
                                                                               C
     55.7 211 171 [????????????] ?????? ??????  56.8   34.5 260.4   40.7   16.9          T ?? ??  ?????? ?????
      1.2  13  12 [????????????] ?????? ??????  59.0    3.4   5.3    0.6    1.7          B ?? ??  PTY??           0  0
                                                                               E

FROM:  16: 0: 3  ON  MONDAY   15-FEB-82   TO:   0: 0: 2  ON  TUESDAY  16-FEB-82   INTERVAL:   7:59:59   MEASURED: 100%
GROUPED BY:   BATCH
SORTED BY:    CPU%

JOB  AVG  LOG LOG ------PPN----- CHARGE PRGRM  PAGES   CPU%  UUOS  READS WRITES SCD CT B STATE  TTY   NODE--LINE LOGIN AT
 #   JOBS  IN OUT   PROJ  PROG   NUMBER  NAME  (AVG)         /SEC   /SEC   /SEC CLS                               DAY TIME
     28.4  63 111 ******INTERVAL TOTALS******   34.8   13.1 103.8   32.4    9.9
                                        .
                                        .
                                        .
```

Figure 2-4

2.3.4 PPN Report

Figure 2-5 is a sample of the default report WCMNO.RP4 and shows monthly resource usage by Project-Programmer Number on an 8-hour shift basis. It can be used to determine which users consume the most resources on the system.

The second detail line (A) of the second subreport shows that there were an average of 21.6 operator [1,2] jobs logged on during prime time. These jobs used a total of 3.3% of the CPU during this period. Question marks in the charge number and program name field indicate that values of these items varied, i.e. there were several programs run by [1,2] jobs under several charge numbers.

It is also possible to get a similar report grouped only by Project Number. In addition, if a user operates under several PPN's it is possible to combine those PPN's into a single PPN group for reporting purposes. This latter feature is described in the Appendix called "Grouping PPN's for Reporting Purposes".

```
**********************************************************************************************************
*                                                                                                        *
*                                        AMAR WORKLOAD REPORT                                            *
*     SITE:  <Put Any Title Here>                                                                        *
*                                                                                  SYSTEM:  PATH         *
*     REPORT DESCRIPTION:  MONTHLY REPORT BY PPN (TYPICAL 8-HOUR SHIFTS)                                 *
*                                                                                                        *
*     INPUT FILE:  8232  .DBO    (FISCAL YEAR: 82   QUARTER: 3   MONTH: 2   WEEKDAYS)                     *
*                                                                                                        *
**********************************************************************************************************


 0: 0 -  8: 0  WEEKDAYS (-HOLIDAYS)     FROM: MON 25-JAN-82  TO: FRI 19-FEB-82  (20 DAYS)     MEASURED:  94% = 151.77 HOURS
GROUPED BY:    PPN

SORTED BY:     CPU%      CUTOFF:  0.50% OF CPU

JOB  AVG  LOG LOG ------PPN----- CHARGE PRGRM  PAGES    CPU%   UUOS  READS WRITES SCD CT B STATE  TTY    NODE--LINE LOGIN AT
 #   JOBS  IN OUT    PROJ  PROG  NUMBER  NAME  (AVG)           /SEC  /SEC   /SEC CLS                                DAY TIME
     22.3 333 194 ******INTERVAL TOTALS******   67.3     3.4   16.5   3.5    1.7

      0.1  10  14 [ 15152112403] ?????? ??????  68.8     1.3    2.0   0.6    0.7       0   ??            ?????
     19.5  82  59 [     1     2] ?????? ??????  39.4     0.8    8.5   0.9    0.3               ?? ??        ?????
      0.0   0   3 [   275 42203] 224300 ??????  93.3     0.6    2.0   1.0    0.3       2 B ??

     19.6  92  76 ***SUBTOTALS THRU CUTOFF***   65.2     2.7   12.5   2.5    1.2

      2.7 241 118 ***SUBTOTALS AFTER CUTOFF**   75.7     0.7    4.0   0.9    0.5


 8: 0 - 16: 0  WEEKDAYS (-HOLIDAYS)     FROM: MON 25-JAN-82  TO: FRI 19-FEB-82  (20 DAYS)     MEASURED:  94% = 151.22 HOURS
GROUPED BY:    PPN

SORTED BY:     CPU%      CUTOFF:  0.50% OF CPU

JOB  AVG  LOG LOG ------PPN----- CHARGE PRGRM  PAGES    CPU%   UUOS  READS WRITES SCD CT B STATE  TTY    NODE--LINE LOGIN AT
 #   JOBS  IN OUT    PROJ  PROG  NUMBER  NAME  (AVG)           /SEC  /SEC   /SEC CLS                                DAY TIME
     58.749164165 ******INTERVAL TOTALS******   63.7    31.3  208.0  57.4   18.4

      0.8 105  86 [ 15152112403] ?????? ??????  56.3     3.3    8.5   5.0    3.0       0   ?? ??         ?????
 (A) 21.6 255 213 [     1     2] ?????? ??????  33.6     3.3   55.6   2.7    1.7           ?? ??         ?????
      0.6  48  38 [ 15152 45010] ?????? ??????  67.2     2.3    8.0   3.7    2.6       0   ?? ??         ?????
      0.7  63  52 [ 20320112161] ?????? ??????  80.1     1.5    8.6   1.8    0.9       0 T ?? ??         ?????
      0.4  72  75 [ 12400112136] ?????? ??????  62.0     1.3    5.8   5.7    0.4       0   ?? ??         ?????
      0.7  31  15 [ 20320112136] ?????? ??????  78.1     1.3    4.5   0.8    0.3       0 T ?? ??         ?????
      1.6 136 153 [   275 43226] ?????? ??????  40.5     1.3   22.8   0.6    0.1           ?? ??         ?????
      0.4 136 128 [ 15152 46013] ?????? ??????  61.5     1.1    2.0   1.7    0.9       0   ?? ??         ?????
      0.5  73  67 [ 15152112424] ?????? ?????? 129.1     1.1    5.6   3.0    0.8       0   ?? ??         ?????
      0.2  50  44 [ 15152112160] ?????? ??????  63.4     0.8    2.9   1.4    0.5       0   ?? ??         ?????
      2.1 110 101 [ 12400132153] ?????? ??????  79.9     0.8    3.5   1.4    0.5       0   ?? ??         ?????
      0.9 109 103 [ 12400112154] ?????? ??????  51.1     0.7    4.7   1.5    0.2       0   ?? ??         ?????
                                                          .
                                                          .
                                                          .
```

Figure 2-5

2.4 HOW TO RUN THE PROGRAMS

2.4.1 Data Collection

The Data Collection program, WHOWC, should be run as an OPSER subjob. There will be a corresponding subjob to collect data for System AMAR.

The following commands should be inserted into the OPR.ATO file to ensure automatic startup and continuous data collection:

```
:SLOGIN ppn
:DEFINE WC=
WC-RUN structure:WHOWC[ppn]
```

These commands may be entered directly to OPSER to get WHOWC started the first time. The last line may be used to restart WHOWC if it has stopped because of disk parity errors or the like.

WHOWC creates an output file named WC.RAW. The output file is closed after each checkpoint interval.

2.4.2 Generating Automatic Reports

There are three programs used in the automatic reporting process - WCINC, WCUPD, and WCRPTB. WCINC and WCUPD massage the raw file output by the data collection program and create the database files. WCRPTB is the report generating program which operates on the database files. These programs are normally run as part of a nightly batch stream, WCRPTB.CTL, which is self-submitting. By using special wild carded filenames (described under the INPUT FILE.EXT = command in the Appendix called, "Report Program (WCRPT, WCRPTB, and WCRPTC) Dialogue"), it is possible to generate daily, weekly, and monthly reports through this one stream without operator intervention. The reporting program WCRPTB, keeps track of the database files on which it has already reported. It also recognizes when a fiscal day, week or month has ended and then produces the appropriate daily, weekly and monthly reports. An exception occurs if the stream is not run every day. Processing will get several days behind. Then only the first fiscal report type for the first fiscal period encountered will be produced. There are six types of fiscal period in Workload AMAR. These periods correspond to the six types of files listed under the "DORPT:" step of WCRPTB.CTL. See the Appendix called "Workload AMAR Batch Stream - WCRPTB.CTL".

Four default reports are supplied with the package. These reports are described in the previous section and may be produced at your option on a daily, weekly, or monthly schedule or not at all. You can define special reports through the WCRPTB program dialogue. The dialogue responses can be added to the daily batch stream. The special reports will then be produced automatically.

2.4.3 Generating Special Reports

2.4.3.1 What Program Do I Use? -

There are four programs for generating special reports - WCRPTB, WCRPTC, WC and WCRPT. For detailed explanations of report program dialogues, see the Appendix called "Report Program (WCRPT, WCRPTB, and WCRPTC) Dialogues".

Use WCRPTB to report from the workload database. See Figure 2-6 for an example of WCRPTB dialogue and the resultant report.

Use WCRPTC to report from the database only if you need more detail groups than WCRPTB can handle, for example, if you wanted to report on every program run by every user over the period of a month.

Use WC and WCRPT to report from yesterday's or today's data for one of two reasons:

1. You want to look at today's data which will not be in the database until after midnight.

2. You want to look at yesterday's data for intervals less than one hour or not starting and/or ending on hour boundaries.

```
.run wcrptb

REPORT DESCRIPTION = primetime node and line usage 1

INPUT  FILE.EXT = 823242.db0 2

OUTPUT FILE.EXT = worknl.rpt  3

PPN GROUPING FILE.EXT =

ENTER DESIRED START AS    HH MM: 8  4
ENTER DESIRED END   AS    HH MM: 16  5
ENTER DESIRED INTERVAL SIZE AS  HH MM:  6

ENTER MAXIMUM DETAIL LINES PER INTERVAL:
ENTER CPU% CUTOFF:                        7

ANY SPECIAL MASKS OR SORT ORDERS? (Y OR N): n  8

ID ITEM     0-0: node  9

ID ITEM     0-1: line  10

ID ITEM     0-2:

SORT ITEM   1-0: node  11

SORT ITEM   1-1: jobs  12

SORT ITEM   1-2:

SORT ITEM   2-0:

MORE REPORTS? (Y OR N): n

END OF EXECUTION
CPU TIME: 1.55  ELAPSED TIME: 14.85
EXIT

.
```

Sample WCRPTB Dialogue

Figure 2-6

1. Free form report description. This report shows the percent of time each line was used on each node. The most heavily used lines are listed first under each node. To determine the percent of time the line was used, look under the AVG JOBS column. 1.0 = 100%; .9 = 90%; etc. 0 means that each time a sample was taken, no one was using that line. Remember samples are taken every 5 minutes.

2. Daily file for February 15,1982 (FY82, third quarter, second month, fourth week, second day.)

3. Report filename.

4. Start the report at 8:00 AM. Minutes, seconds, and day default to 0 if only the hour is specified. If carriage return only had been entered, the report would have started at the beginning of the file .

5. The report will stop at 16:00 PM (4:00 PM). If carriage return only had been entered, the report would have stopped at the end of the file .

6. Since carriage return was entered, the report will cover the entire period between 8:00 AM and 4:00 PM.

7. No special cutoffs will be used.

8. No other special reporting features will be used.

9. and 10. Data is grouped by line within each individual node.

11. The major sort is by node.

12. For each node, details will be sorted by AVG JOBS (jobs). This will have the effect of listing the most heavily used lines for that node first. The "node" and "jobs" mnemonics have a default sort order implied.

```
*************************************************************************************************************************
*                                                                                                                       *
*                                              AMAR WORKLOAD REPORT                                                     *
*                                                                                                                       *
*   SITE:  <Put Any Title Here>                                                                  SYSTEM:  PATH          *
*                                                                                                                       *
*   REPORT DESCRIPTION:  primetime node and line usage 1                                                                *
*                                                                                                                       *
*   INPUT FILE:  823242.DBO 2 (FISCAL YEAR: 82  QUARTER: 3  MONTH: 2  WEEK: 4  DAY: 2  MONDAY)                          *
*                                                                                                                       *
*************************************************************************************************************************
```

4 5 6

FROM: 8: 0: 2 ON MONDAY 15-FEB-82 TO: 16: 0: 3 ON MONDAY 15-FEB-82 INTERVAL: 8: 0: 0 MEASURED: 100%

GROUPED BY: NODE LINE 9 10

SORTED BY: NODE JOBS 11 12

JOB #	AVG JOBS	LOG IN	LOG OUT	------PPN----- PROJ PROG	CHARGE NUMBER	PRGRM NAME	PAGES (AVG)	CPU%	UUOS /SEC	READS /SEC	WRITES /SEC	SCD CLS	CT	B	STATE	TTY	NODE	--LINE	LOGIN AT DAY	TIME
	56.8	224	183	******INTERVAL	TOTALS******		57.0	37.9	265.7	41.3	18.5									
20	1.0	0	0	[12400112206]	750223	1022	60.4	0.0	0.2	0.0	0.0	4	0	T	TI	TTY73	NODED	73	0	7:41
28	1.0	1	1	[15152 4????]	216?00	??????	52.3	1.4	8.3	2.7	1.7	4	0	T	?? ??	TTY55	NODED	55	0	
18	1.0	1	0	[15323112152]	555267	??????	44.8	0.1	1.4	0.5	0.0	4	0	T	?? ??	TTY25	NODED	25	0	8:04
	1.0	1	1	[12400132153]	750240	1022	90.7	0.4	2.3	0.9	0.1	4	0	T	??	TTY75	NODED	75	0	
31	1.0	1	0	[15152112437]	248100	??????	39.4	0.2	3.3	0.2	0.1	4	0	T	?? ??	TTY23	NODED	23	0	8:09
	0.9	9	9	[124001321??]	??????	??????	64.5	0.3	2.2	0.9	0.0		0	T	?? ??	TTY74	NODED	74	0	
17	0.9	0	0	[1 2]	133085	??????	40.0	0.2	1.6	0.1	0.0	0	54	T	?? ??	TTY102	NODED	102	-3	21:59
6	0.9	1	0	[15152112157]	216300	??????	93.0	1.0	5.7	0.7	0.3	4	0	T	?? ??	TTY17	NODED	17	0	8:15
43	0.9	1	0	[1 2]	133085	DLLSPL	29.9	0.3	20.5	0.0	0.0	0	54	T	?? ??	TTY103	NODED	103	0	8:47
49	0.8	1	0	[20320112136]	555040	??????	58.8	13.2	2.8	0.3	0.1	4	0	T	?? ??	TTY76	NODED	76	0	9:14
	0.8	2	1	[15???1121?1]	??????	??????	30.1	0.0	0.3	0.0	0.0	4	0	T	?? ??	TTY13	NODED	13	0	
	0.8	2	3	[20320112161]	555040	??????	40.7	1.2	10.8	2.0	0.7	4	0	T	?? ??	TTY77	NODED	77	0	
	0.7	3	1	[15152??????]	2?????	??????	62.0	0.3	1.5	0.3	0.2	4	0	T	?? ??	TTY16	NODED	16	0	
	0.6	5	4	[1????1?2???]	??????	??????	51.5	0.7	4.8	9.4	7.4	4	0	T	?? ??	TTY22	NODED	22	0	
	0.6	5	5	[12400?721??]	7502??	??????	41.8	0.1	1.2	0.4	0.0	4	0	T	?? ??	TTY106	NODED	106	0	
	0.6	2	1	[12400112206]	750223	????	48.8	0.0	0.3	0.0	0.0	4	0	T	??	TTY71	NODED	71	0	
60	0.4	1	0	[17600112137]	216410	?????	47.0	0.0	0.1	0.0	0.0	4	0	T	?? ??	TTY70	NODED	70	0	12:26
	0.4	2	1	[1????112???]	75????	??????	108.0	0.1	1.0	0.3	0.0	4	0	T	?? ??	TTY107	NODED	107	0	
36	0.4	1	0	[20320112143]	463395	??????	78.5	0.9	64.3	0.1	0.0	4	0	T	?? ??	TTY100	NODED	100	0	13:07
	0.3	1	0	[15152 4????]	216300	??????	22.1	0.1	0.8	0.1	0.1	4	0	T	?? ??	TTY27	NODED	27	0	
	0.3	7	7	[15152??????]	??????	??????	31.8	0.0	0.4	0.0	0.0	4	0	T	?? ??	TTY11	NODED	11	0	
	0.3	5	5	[153231121??]	555267	??????	28.8	1.5	5.9	0.1	0.0	4	0	T	?? ??	TTY37	NODED	37	0	
	0.3	9	10	[15???112???]	???2??	????	30.0	0.1	2.1	0.1	0.0	4	0	T	??	TTY35	NODED	35	0	
	0.2	3	2	[15152112???]	??????	?I????	31.2	0.2	1.5	0.4	0.0	4	0	T	?? ??	TTY5	NODED	5	0	
45	0.2	2	1	[15152??????]	271122	????	76.7	0.2	1.2	0.5	0.0	4	0	T	?? ??	TTY4	NODED	4	0	
	0.2	4	2	[1532311214?]	555267	??????	161.6	0.3	1.4	0.5	0.2	4	0	T	?? ??	TTY36	NODED	36	0	
	0.2	4	2	[15375112 4??]	555571	??????	141.1	0.4	0.6	0.2	0.0	4	0	T	??	TTY54	NODED	54	0	
	0.2	8	7	[???????????]	??????	??????	26.3	0.1	1.0	0.2	0.0		0	T	?? ??	TTY33	NODED	33	0	
	0.2	3	2	[15152???1??]	??????	?????	29.9	0.0	0.1	0.0	0.0		0	T	?? ??	TTY30	NODED	30	0	
	0.1	3	0	[15?????????]	75????	??????	30.6	0.0	0.1	0.0	0.0	4	0	T	?? ??	TTY45	NODED	45	0	
	0.1	2	2	[12400112???]	750???	?????	32.3	0.0	0.1	0.0	0.0	4	0	T	?? ??	TTY32	NODED	32	0	
72	0.1	1	1	[15152112435]	216114	SOS	30.1	0.0	0.1	0.0	0.0	4	0	T	TI	TTY34	NODED	34	0	15:05
39	0.1	1	1	[15152112136]	216119	OPSTAT	41.7	0.1	0.2	0.0	0.0	4	0	T	SL	TTY6	NODED	6	0	12:09
29	0.0	1	0	[1 2]	680160	PIP	31.4	0.0	0.3	0.1	0.0	4	2	T	??	TTY104	NODED	104	0	8:40

Figure 2-6 (continued)

```
37  0.0   1  0 [        1         2] 133085 PIP      19.1   0.0   0.0   0.0   0.0   0 54 T ^C     TTY113 NODED 113  0  8:27
    0.0   2  1 [        2         5]        LOGIN    32.0   0.0   0.0   0.0   0.0   0  0 T SL     TTY26  NODED  26  0
    0.0   2  1 [        2         5]        LOGIN    32.0   0.0   0.0   0.0   0.0   0  0 T SL     TTY7   NODED   7  0
39  0.0   1  0 [        2         5]        SYSTAT   19.3   0.0   0.0   0.0   0.0   0  0 T TO     TTY21  NODED  21  0 11:19
70  0.0   1  1 [  15323112135] 555267 ??????   29.9   0.0   0.1   0.0   0.0   4  0 T ?? ??  TTY140 NODEF  13  0 15:34
52  0.5   1  1 [  12400112206] 750223 1022     59.4   0.1   0.5   0.2   0.0   4  0 T T?     TTY126 NODEH   6  0  9:22
    0.2   2  2 [  12400112154] 750192 ????     46.0   0.1   1.0   0.1   0.0   4  0 T ?? ??  TTY13? NODEH   5  0
 6  0.0   0  1 [210400 62165] 133000 ???      69.6   0.0   0.1   0.1   0.0   4  0 T ??     TTY120 NODEH  54  0  7:58
32  0.0   1  1 [  12400112154] 750192 1022     49.2   0.0   0.1   0.0   0.0   4  0 T T?     TTY130 NODEH   1  0 11:00
45  0.0   1  1 [  12400112170] 750205 1022     46.1   0.0   0.2   0.3   0.2   4  0 T TI     TTY126 NODEH  21  0  9:11
    0.2   2  2 [  12400112150] 750188 ????     39.8   0.0   0.6   0.1   0.0   4  0 T ?I     TTY1?? NODEI   3  0
64  0.1   1  1 [  12400112205] 750222 ????     53.0   0.0   0.4   0.3   0.2   4  0 T ?I     TTY134 NODEI  35  0  9:59
21  1.0   0  0 [        1         2] 133085 OPSER     8.0   0.1   2.6   0.0   0.0   0  1 T ?? ??  TTY115 NODEM  21 -4 23:21
    1.0   1  0 [        1         2] 133085 ??????   21.3   0.0   0.1   0.0   0.0   0 54 T ^C ??  TTY117 NODEM  20
    0.9   3  5 [  1????1?2???] 75???? ??????   83.5   0.2   0.8   0.9   0.0   4  0 T ?? ??  TTY1?? NODEM   0  0
    0.9   2  1 [  1????112???] 75???? ??????  140.5   0.3   0.4   0.1   0.0   4  0 T ?? ??  TTY12? NODEM  37  0
    0.8  11  9 [  ???????????] ?????? ??????   45.5   0.7   3.2   1.8   0.4      0 T ?? ??  TTY1?? NODEM   1  0
    0.8   5  3 [  ??70???????] ?????? ??????   44.7   0.5   1.6   2.1   0.2      0 T ?? ??  TTY1?? NODEM   3  0
    0.8   5  4 [  ????? ?????] ?????? ??????   50.2   0.1   0.8   0.3   0.1        T ?? ??  TTY12? NODEM  25  0
    0.7   8  6 [  ??70???????] ?????? ??????   49.9   0.2   1.2   0.6   0.3      0 T ??     TTY1?? NODEM   2  0
    0.7   3  2 [  ????? ?????] 133105 ??????   40.7   0.1   0.5   0.2   0.1   4  1 T ?? ??  TTY1?? NODEM  24  0
    0.6   3  3 [  12400?721??] 750??? 1022?    71.5   0.3   1.4   0.7   0.0   4  0 T ?? ??  TTY13? NODEM   5  0
    0.6  11  9 [  12400?2????] 750??? ??????   47.9   0.2   1.7   0.7   0.0   4  0 T ??     TTY1?? NODEM   4  0
    0.5   4  3 [  ???????????] ?????? ?????    41.1   1.5  25.0   0.2   0.0        T ?? ??  TTY1?? NODEM  23  0
    0.5   2  2 [  ??70???????] ?????? ??????   56.6   1.1   6.6   4.1   1.0      0 T ?? ??  TTY1?? NODEM   6  0
    0.5   2  0 [   7700112136] 133100 ??????   25.9   0.0   0.2   0.0   0.0   4 74 T ?? ??  TTY13? NODEM  31  0
    0.5   5  3 [  ??70???????] ?????? ??????   65.9   1.5   4.0   2.9   1.4      0 T ?? ??  TTY1?? NODEM  10  0
    0.4   7  6 [  ??70???????] ?????? ??????   46.8   0.3   1.6   1.0   0.6      0 T ?? ??  TTY13? NODEM   7  0
    0.3   3  3 [  12400112154] 750192 1022     52.5   0.1   0.7   0.6   0.4   4  0 T ??     TTY1?? NODEM  30  0
    0.3   4  4 [   ??????????] 133105 ??????   32.7   0.4   7.7   0.1   0.0   4  1 T ?? ??  TTY12? NODEM  22  0
    0.3   2  3 [  12400?721??] 750??? ??????   48.1   0.1   0.6   0.2   0.0   4  0 T ??     TTY1?? NODEM  11  0
    0.2   2  1 [  1????112???] 75???? 1022??  133.9   0.1   0.3   0.3   0.2   4  0 T T? ??  TTY1?? NODEM  14  0
73  0.0   1  0 [  16016112136] 555114 1022C   118.8   0.0   0.3   0.0   0.0   4  0 T TI     TTY137 NODEM  36  0 15:37
70  0.0   1  1 [  12400132150] 750244 ??????   56.3   0.0   0.2   0.2   0.0   4  0 T ?? ??  TTY137 NODEM  13  0 15:18
64  0.1   1  0 [  15323112150] 555267 TIP      26.8   0.0   0.1   0.0   0.0   4  0 T T?     TTY123 NODEN  23  0 14:54
65  0.1   1  0 [  15323112150] 555267 RLPSPL   13.3   0.0   0.2   0.0   0.0   4  0 T ?? ??  TTY136 NODEN  24  0 14:55
32  0.1   1  1 [   7700112151] 133100 ?????    30.0   0.2   0.4   0.0   0.0   4  0 T ??     TTY122 NODEN   2  0 11:46
    0.0   5  4 [  15152??????] 2??700 ??????   31.3   0.0   0.1   0.0   0.0   4  0 T ?? ??  TTY1?? NODEN   4  0
   25.3  23 29 [?????????????] ?????? ??????   47.1   5.9  52.3   1.9   2.2          ?? ??  ??????        0
```

Figure 2-6 (continued)

2.4.3.2 Special Report On Yesterday's Data -

To obtain a special report on yesterday's data down to the 5 minute level, run WCRPT according to the general directions in the Appendix called "Report Program (WCRPT, WCRPTB, and WCRPTC) Dialogue", specifying an output filename of the form yyqmwd.ext where yyqmwd represents yesterday's fiscal date (year, quarter, month, week, day) and ext is not .IN0, .IN1, .RP?, .EXE, or .RAW. Before you run WCRPT, make sure that there are two files for yesterday named yyqmwd.IN0 and yyqmwd.IN1, as these are the input files required by WCRPT. If an hourly or higher level report is needed, the WCRPTB program and the database file for yesterday may be used.

2.4.3.3 Special Report On Today's Data -

To obtain a special report on today's data:

1. Run WC which will produce files WC.IN0 and WC.IN1.

2. Run WCRPT according to general directions in the Appendix called "Report Program (WCRPT, WCRPTB, and WCRPTC) Dialogue", specifying an output filename of the form WC.ext, where .ext is not .IN0, .IN1, .RP?, .EXE, or .RAW.

2.4.4 Examining/Changing The Workload Holidays (WCFIX)

In addition to specifying holidays for Workload AMAR, holidays must also be specified for System AMAR. Although the holidays should be the same throughout AMAR, the processes to set them are separate. The WCFIX program is used to specify holidays for Workload AMAR. For System AMAR, see the Section entitled "Examining/Changing Database Parameters (AMRGEN)". The holiday list in Workload AMAR is currently limited to a maximum of 18 entries. Once the list is full, adding a new entry will cause the earliest date to be deleted from the list. Thus it should never be necessary to explicitly delete holidays unless they have been set by mistake. On the other hand, you should not try to set more than 18 future holidays because the earliest would be deleted from the list before it could do its job of making the corresponding data enter the database as holiday data. WCFIX can also be used to temporarily reset the grace period. The grace period has a threefold purpose:

1. It allows automatic processing to be resumed after as many as 7 days (the standard default) with no special action.

2. It prevents processing of more than 7 days of data at one time which could cause disk problems.

3. It avoids database update when the operator has set the system date more than 7 days into the future. If database update were allowed in that case, dummy data would be entered into the database which could not be replaced by the real data and old data would be deleted prematurely. In an extreme case, if the operator set the date a year ahead and the longest retention in the database was a year, the whole database could be filled with dummy data and it would be impossible to enter any real data for the next year.

Before using WCFIX to temporarily extend the grace period, you should make sure there is enough disk space to process the extra days, make sure the system date is correctly set, and make sure the workload data collection program is collecting data. If the normal batch stream is going to be used, be sure to request enough time and avoid multiple submissions. The next time WCINC runs successfully, it will set the grace period back to the default of 7 days.

WCFIX is command driven. Both lower and upper case are valid. WCFIX prompts with an asterisk (*).

Valid Commands:

HELP

Function: To provide a brief sysnopsis of the valid commands.

S H yymmdd

Function: To set a holiday (yy = the normal calendar year; mm = the month; and dd = the day). A holiday must be set <u>before</u> the corresponding data is entered into the database. Otherwise, setting the day as a holiday will have no useful effect.

D H yymmdd

Function: To delete a holiday (yy = the normal calendar year; mm = the month; and dd = the day). Once the corresponding data has been entered into the database as a holiday, deleting the holiday will have no useful effect other than to free up a slot in the holiday list.

L H

Function: To list the holidays. Holidays will be listed in the following format: fiscal date followed by a 3 character day abbreviation (MON-FRI) followed by the normal calendar date.

S G nnn

Function: To temporarily set the grace period to nnn days.

EXIT
E D

Function: To exit the program and return to monitor level.

2.5 PROCEDURE FOR RUNNING WCRPTB.CTL

2.5.1 Overview Of WCRPTB.CTL

This stream runs daily. It takes the workload data which has been collected by WHOWC since the last time WCRPTB.CTL ran, preprocesses it, updates the database, and produces daily reports (and weekly and monthly reports if appropriate). See the Appendix called "Workload AMAR Batch Stream - WCRPTB.CTL" for a sample of the stream and step descriptions.

2.5.2 Resource Requirements

To run the workload system on a continuous basis, you will need an OPSER subjob to run the WHOWC program and a permanently mounted disk area with at least 5K blocks available for raw data. The average number of simultaneous users is the key factor determining daily file size. The size of your workload database will probably range between 6K blocks and 30K blocks. Size depends partly on the complexity of your workload and partly on the retention periods you specify. This space may be on a pack separate from the raw data, but should also be permanently mounted, if possible. See the Appendix called "Installation and Resource Requirements" for a more complete discussion of resource requirements.

2.5.3 Submission

The stream resubmits itself in most cases, whether completion is successful or not. This is because it is important not to let raw data build up on disk. There should always be a WCRPTB.CTL in the submit queue, set to run /AFTER:0:30:0.

2.5.4 Restart Procedure

If a system crash occurs while the stream is running, the stream automatically restarts at the appropriate checkpoint. The user should never have to resubmit WCRPTB.CTL unless the submit queue entry is destroyed. Each major step is checkpointed.

APPENDIX A

SYSTEM AMAR ITEM DEFINITIONS

This appendix contains a definition for each item which can be monitored by System AMAR. Items are listed in alphabetical order by the title which appears on the System AMAR reports. The title is the first line in each definition below. Preceeding special characters such as # or % are ignored. Following the title line is a line which contains a 4 character code which uniquely identifies the item and its corresponding records in the System AMAR database.

Beneath each item code is a letter, in parentheses, which indicates whether the item is metered (M), snapped (S), or obtained from the performance meter (PM). Metered implies that the value of the item is accurate, regardless of the size of the sample interval. Snapped means that the sample taken is a "snapshot" of the item at sample time. Subsequent or prior to the snapshot, the values could have been very different. Thus, the accuracy of the values' correspondence to what is actually happening on the system depends on the number of samples taken during the reporting period. Performance meter variables are available only on the KL10. For a discussion of the performance meter and how to use it, see the METER monitor call in the DECsystem-10 Monitor Calls Manual.

Item values reported as a "per second" count were obtained by dividing the value observed at sample time by the number of seconds in the sample interval.

DECSYSTEM-10 ITEMS

ACT SWAP RATIO
ACSR - Active Swapping Ratio
(S)

Active Swapping ratio is a ratio of the amount of memory currently in use by runnable jobs to the total amount of user memory available.

ACTV JOB % USR CR
ACCR - User Core Occupied by Active Jobs (Percent)
(S)

User core occupied by active jobs is the amount of physical core (less the monitor low and high segments) divided into the amount of core occupied by jobs not in long term wait states such as SL, HB, etc. or in a terminal input wait state (TI and JB.UOA clear). Shared segments are counted only once.

ALL JOBS % USR CR
PHCR - User Core Occupied by All Jobs (Percent)
(S)

User core occupied by all jobs is obtained by dividing the amount of core occupied by all jobs by the total amount of physical core available (less the monitor low and high segments). Shared segments are counted only once.

% AMAR CLK TIME
XAMT - AMAR Measured Time (Percent of Realtime)
(M)

AMAR measured time is the percentage of time that the Data Collection Program was running.

% AMAR TI CPU n UP
CPA0, CPA1 - CPUn Uptime
(M)

CPUn uptime is the percent of time measured by AMAR that the CPU was up. CPUn uptime is recorded separately for each CPU within the system. For a multiprocessor configuration, CPA0 is recorded for the first CPU, CPA1 for the second, etc.

AVG JOB SIZE
AVJS - Average Job Size (Pages)
(S)

Average job size is the sum of the sizes, measured in pages, of all jobs logged into the system divided by the number of jobs. The null job is not included. Shared segments are counted only once.

AVG SCHD RSP TIME
AVRT - Average Scheduler Response Time (Ms)
(M)

Average scheduler response time is the average number of milliseconds that elapsed between the issuance of a command to execute a program or remove a job from a TTY input wait state and selection of the job by the scheduler. This is derived from two accumulators maintained by the monitor, one which counts milliseconds and one which counts the number of responses by the CPU.

CACHE SWEEP/SEC
CPS0, CPS1 - CPUn Cache Sweep Rate (Sweeps/Sec)
(M)

Cache sweep rate is the total number of cache sweeps executed on the CPU per second. It is obtained from an accumulator maintained by the monitor. Cache sweeps are memory updates by cache initiated primarily by direct memory accesses, or context switches.

CPUn cache sweep rate is recorded separately for each CPU within the system. For a multiprocessor configuration, CPS0 is recorded for the first CPU, CPS1 for the second, etc. Cache sweeps are only recorded for KL10's.

CHAN n % TIM BUSY
CHn0, CHn1 - Channel Busy Time (Percent)
(PM)

Channel busy time is the percent of real time this channel was busy. n = 0 to 7.

CHAN nn WAIT Q
CTWQ - Channel Transfer Wait Queue Length
(S)

The transfer wait queue length is the count of commands awaiting channel action.

The length of a disk channel transfer wait queue is recorded separately for each channel.

CONTEXT SWTS/SEC
CPC0, CPC1 - Context switches per second
(M)

Context switches per second are recorded separately for each CPU. In a multiprocessor configuration, CPC0 is recorded for the first CPU, CPC1 for the second, etc.

% CPU UTIL
_CPU - CPU Utilization

CPU utilization is derived by the reporting program by subtracting idle time (the average of CPI0 and CPI1) from 100%. It includes lost and overhead time, user program time, and some embedded priority interrupt (PI) processing time. Since this is a derived item, it cannot be obtained directly from the database.

See CPU% in the Appendix called "Valid Grouping and/or Sort Items" for the variable representing individual job CPU utilization.

% IDLE TIME
CPI0, CPI1 - CPUn Idle Time (Percent)
(M)

Idle time is the amount of CPU time the null job was running minus lost time.

At the sample group interval, CPUn idle time is recorded as a percentage of wall clock time over which the item is measured.

CPUn idle time is recorded separately for each CPU within the system. For a multiprocessor configuration, CPI0 is recorded for the first CPU, CPI1 for the second, etc.

JOBS BLK IO Q
IOWP - Jobs in Noninteractive IO Queues
(S)

The number of jobs in noninteractive I/O queues is the count of jobs waiting for paging (PI state), disk I/O (DI state), or input and output to a device (IO state). The number of jobs in the several states are summed.

JOBS IN RUN QUEUE
NRJR - Number of Jobs in the Run Queue
(S)

The number of jobs in the run queue is the number of jobs available for selection (RN state) to run in the CPU. A job will be counted whether it is core resident or swapped.

JOBS LOGGED IN
JLOG - Jobs Logged in Concurrently
(S)

The number of jobs logged in concurrently is the total number of jobs logged into the system not counting the null job.

JOBS TTY IO Q
TIOW - Jobs in Interactive IO Queue
(S)

The number of jobs in the interactive IO queue is the number of jobs waiting for terminal input or output (TI state).

LINES IN USE
TTYU - Terminal Lines in Use Concurrently
(S)

The number of terminal lines in use is a count of all lines (CTY, remote, local, slave, etc.) in use except PTY's.

% LOST TIME
CPL0, CPL1 - CPUn Lost Time (Percent)
(M)

Lost time is the amount of CPU time a runnable job was waiting to be swapped in while the null job ran.

At the sample group interval, CPUn lost time is recorded as a percentage of wall clock time over which the item is measured.

CPUn lost time is recorded separately for each CPU within the system. For a multiprocessor configuration, CPL0 is recorded for the first CPU, CPL1 for the second, etc.

% MON FR CORE USED
PFCU - Free Core in Use (Percent)
(S)

Free core in use is the percentage of 4-word free core blocks in use by the monitor. This is determined by examining two accumulators maintained by the monitor; one contains the current amount of core in use and the other contains the total amount of core allocated to the monitor for use.

MT name BLKS/SEC
MTIO - Magtape I/O (Blocks/Second While Assigned)
(M)

Magtape I/O rate is the number of equivalent disk blocks read and written per second from a drive and is calculated based on the length of time the drive was assigned not the amount of time AMAR was sampling. Internally the monitor maintains the number of frames read and written from a tape unit on a mount per job basis, from the time the tape unit was mounted to the time the tape unit was unloaded. A frame consists of 7 or 9 bits of information (depending on the number of channels on the drive) with 1 bit written in each channel. Frames are converted by the data collection program into blocks of 128 words each.

Magtape I/O rate is recorded separately for each magnetic tape unit active.

While Magtape IO rate is a metered item in so far as accurate counts are maintained by the monitor, it can be sensitive to sample interval. A long sample interval could result in job overlapping between sampling, while a shorter interval would more likely catch the appropriate values before they were reset for a new job.

MTAS ASSIGNED
MTAU - Magtape Units in Use Concurrently
(S)

Magtape units in use is the total number of magnetic tape units assigned to jobs at sample time.

% OVHD TIME
CP00, CP01 - CPUn Overhead Time (Percent)
(M)

Overhead time is the amount of CPU time spent in clock queue processing, short command processing, swapping and scheduling, context switching and interrupt servicing.

At the sample group interval, CPUn overhead time is recorded as a percentage of wall clock time over which the item is measured.

CPUn overhead time is recorded separately for each CPU within the system. For a multiprocessor configuration CP00 is recorded for the first CPU, CP01 for the second, etc.

PGS USER MEMORY
UMEM - User Memory Available (Pages)
(S)

User memory available is the total amount of physical memory less the amount used by the monitor high and low segments and locked (LK) jobs.

PI n % TIME BUSY
PIn0, PIn1 - Priority Interrupt Level Busy Time (Percent)
(PM)

Priority interrupt level busy time is the percent of real time spent at the level (n = 0 to 7). This measurement uses the the monitor round robin mechanism specific to version 7 of the monitor.

PK name BLKS/SEC
LUIO - Logical Unit I/O Rate (Blocks/Second)
(M)

I/O rate is the number of disk blocks read and written per second by user jobs. Swapping and monitor overhead are omitted from this count.

I/O rate is computed separately for each individual logical disk pack.

See READS and WRITE in the Appendix called "Valid Grouping and/or Sort Items" for the variables representing individual job disk reads and writes.

%PK name FREE SPC
LUFS - Logical Unit Free Space (Percent)
(S)

Logical unit free space is the percentage of unused blocks on a logical pack.

Free space is recorded separately for each logical pack. Both public and private structures are sampled.

PK name SWPS/SEC
LUSW - Logical Unit Swapping Rate (Blocks/Second)
(M)

Logical unit swapping rate is the number of disk blocks read and written per second to the pack while swapping user jobs.

Swapping rate is computed separately for each logical disk pack which has swapping space.

PK name WAIT Q
LUWQ - Logical Unit Position Wait Queue
(S)

The position wait queue is the count of jobs awaiting disk control.

The length of the position wait queue is recorded separately for each logical pack. Both public and private disk structures are sampled.

% RN JOB IN MEM
PRJC - Runnable Jobs in Memory (Percent)
(S)

Runnable jobs in memory is the percent of jobs in the run queues (R1, R2 and HPQ) which are resident in core.

SCN INTR RCVS/SEC
SCRV - SCNSER Interrupts Received/Second
(M)

The number of interrupts received by SCNSER per second.

SCN INTR XMTS/SEC
SXMT - SCNSER Interrupts Transmitted/Second
(M)

The number of interrupts transmitted by SCNSER per second.

SLV CACH SWEEP/SEC - See CACHE SWEEP/SEC.

SLV CHAN n % BUSY - See CHAN n % TIM BUSY.

SLV CTXT SWTS/SEC - See CONTEXT SWTS/SEC.

SLV IDLE TIME - See % IDLE TIME.

SLV LOST TIME - See % LOST TIME.

SLV OVHD TIME - See % OVHD TIME.

SLV PI n % BUSY - See PI n % TIM BUSY.

SLV USER UUOS/SEC - See USER UUOS/SEC.

% SWAP SPC LEFT
SWPS - Swapping Space Left (Percent)
(S)

Swapping space left is the percentage of 1k blocks of virtual core remaining as swapping space.

SWAPPING BLKS/SEC
SWIO - Swapping IO Rate (Blocks/Sec)
(M)

Swapping IO rate is the number of disk blocks read and written per second while swapping user jobs.

SYSTEM RELOADS
XRLD - Number of System Reloads
(M)

Number of System reloads is the number of times the system was rebooted.

% SYSTEM UPTIME
XUPT - System Uptime (Percent of Realtime)
(M)

System uptime is the percentage of time that the system was known to be operational.

%TY CHNK IN USE
PTCU - TTY Chunks in Use (Percent)
(S)

The percent of monitor TTY chunks in use.

UN name BLKS/SEC
PUIO - Physical Unit IO Rate (Blocks/Second)
(M)

Physical unit I/O rate is the number of disk blocks read and written per second by user jobs. Swapping blocks and monitor overhead are omitted from this count.

I/O rate is computed separately for each physical disk unit.

See READS and WRITE in the Appendix called "Valid Grouping and/or Sort Items" for the variables representing individual job disk reads and writes.

%UN name FREE SPC
PUFS - Physical Unit Free Space (Percent)
(S)

Physical unit free space is the percentage of unused blocks on a disk pack.

Free space is recorded separately for each non-fixed head disk unit. Both public and private structures are sampled.

UN name SWPS/SEC
PUSW - Physical Unit Swapping Rate
(M)

Physical unit swapping rate is the number of disk blocks read and written per second from the pack while swapping user jobs.

Swapping rate is recorded separately for each physical pack which has swapping space.

UN name WAIT Q
PUWQ - Physical Unit Position Wait Queue
(S)

The position wait queue is the count of jobs awaiting disk control.

The length of the position wait queue is recorded separately for each non-fixed head disk unit. Both public and private disk structures are sampled.

USER DSK BLKS/SEC
UDIO - User Disk IO Rate (Blocks/Sec)
(M)

User disk I/O rate is the number of disk blocks read and written per second by user jobs. Swapping blocks and monitor overhead are omitted from this count.

See READS and WRITE in the Appendix called "Valid Grouping and/or Sort Items" for the variables representing individual job disk reads and writes.

USER UUOS/SEC
CPU0, CPU1 - CPUn UUO Execution Rate (UUOs/Sec)
(M)

UUO execution rate is the total number of monitor calls, in both executive and user mode, executed on the CPU per second.

CPUn UUO execution rate is recorded separately for each CPU utilized within the system. For a multiprocessor configuration, CPU0 is recorded for the first CPU, CPU1 for the second, etc.

See UUOS in the Appendix called "Valid Grouping and/or Sort

Items" for the variable representing the individual job UUO count.

VIR MEM FAULT/SEC
VMPF - Virtual Memory Paging Rate (Faults/Sec)
(M)

Virtual memory paging rate is the number of page faults per second.

Virtual memory page fault rate is only recorded for systems with the virtual memory option enabled.

APPENDIX B

RAW FILE PREPROCESSOR PROGRAM (AMARSD) DIALOGUE

To obtain special reports on the current System AMAR raw file (today's data), the AMARSD program must first be run. AMARSD always names its output file TODAY.DB. This file may then be input to the AMREPT program to obtain a Daily System Utilization, Disk, or Tape Report. It may also be examined via the AMARON or AMAREX programs. Any System AMAR raw file, including the current day's file, may be run through AMARSD.

.RUN AMARSD

System ID:

Requests the ID of the system whose raw file is to be examined.

Valid Response:

xxxx - 4 character system code

YYMMDD Date of File:

Requests the date of the raw file to be examined:

Valid Response:

yymmdd

Where yy = the normal calendar <u>not</u> fiscal calendar year; mm = the month; and dd = the day.

Prime Periods for yymmdd:

Requests the time period to be considered prime time. The user may enter any prime time period regardless of the specification in the database.

Valid Response:

Start time-End time

Up to four start time-end time pairs may be entered separated by commas. All time is of the format hhss where hh = the hour and ss = the minutes.

Once processing of AMARSD is completed, the following messages will appear:

[AMIHDS Hourly Data Stored for yymmdd]

[Use TODAY.DB as Input File to the AMREPT Program]

You may then run AMREPT, AMARON, or AMAREX to obtain the appropriate report.

APPENDIX C

REPORT PROGRAM (AMREPT) DIALOGUE

AMREPT may be used to generate automatic reports or reports on demand. In either case, a standard set of preformatted reports is obtained. The contents of the reports may be modified via changes to the xxxxDR.RFD File. Generating automatic reports is described in a previous section by that name.

Control-C (^C) may be used to exit at any time. To generate special reports, AMREPT may be run at the terminal as follows:

```
.RUN AMREPT
```

Report Code>

Requests the 2 character code of the standard report to be generated.

Valid Response:

DU Daily System Utilization Report
WU Weekly Utilization Report
MU Monthly Utilization Report

WA Weekly Trend Analysis Report
MA Monthly Trend Analysis Report

WC Weekly 'Typical Day' Report
MC Monthly 'Typical Day' Report

DD Daily Disk Report
WD Weekly Disk Report
MD Monthly Disk Report

DT Daily Tape Report
WT Weekly Tape Report
MT Monthly Tape Report

EXIT

EXIT may be used with this query only to terminate the program.

Default: On subsequent passes through the dialogue, carriage return <CR> will cause the last valid response to this query to be re-used.

Input File>

Requests the name of the AMAR database or the output file produced by the AMARSD program.

Valid Response:

AMAR
AMAR.DB
TODAY
TODAY.DB

AMAR or AMAR.DB specifies the system AMAR database as the input file. TODAY or TODAY.DB specifies the mini-database created by the AMARSD program as the input file.

Default: On subsequent passes through the dialogue, carriage return <CR> will cause the last valid response to this query to be re-used.

Dates>

Requests the dates of the report period to be used.

Valid Response:

start date-end date
AUTO
AUTO:start date-end date
AUTO=end date
AUTO=?

All dates are of the format yymmdd (where yy = the normal calendar (not fiscal) year; mm = the month; and dd = the day). If no data exists for the time period specified, the query is repeated.

The "start date-end date" response should be the only one used when generating special reports. It denotes the range of dates for generating one or more reports. A report will be generated for each fiscal period (implied by the Report code used in the "Report Code>" query) which ends within the date range. The exception is the Weekly/Monthly Trend Analysis Reports which will contain up to 13 weeks or 12 months of data on one report.

If a single daily report is desired, only one date need be entered.

The AUTO response should only be used in the AMAR.CTL jobstream. It will modify the section of the database that controls the automatic reporting capability. AUTO indicates that a check will be made to determine the date of the last fiscal period (day, week, or month) for which an automatic report of the same type has already been generated. The next appropriate fiscal period will be used for this report providing the necessary data is in the database. If AUTO reporting is behind several periods, the AUTO switch will bring you up-to-date. You will get reports for all intervening fiscal periods.

AUTO:start date-end date specifies a range of dates for which one or more reports are to be generated. In addition, the date of the last fiscal period reported is entered into the database as the date of the most recent report of this type. Further use of the AUTO response to generate similar reports will be based on this new date.

AUTO=end date modifies the database, resetting the date of the last automatic report of the type requested to this new date. No reports are produced.

AUTO=? requests a display of the date of the last automatic report of this type and the range of dates to be reported on next.

Default: On subsequent passes through the dialogue, carriage return <CR> will cause the last valid response to be re-used.

Print File>

Requests the filename of the report to be generated. Each report should have a unique name.

Valid Response: Any unique filename of the format filename.ext. File names reserved for AMAR use (see the Appendix called "Summary of Programs and Filenames") should not be specified.

Default: Carriage return <CR> will cause the query to be repeated.

Blank Page

APPENDIX D

ONLINE INQUIRY PROGRAM (AMARON) DIALOGUE

AMARON is the recommended program for examining either a single item/subitem or groups of items/subitems. AMARON is normally run at a terminal. The output, however, can either be displayed at the terminal or stored in a file for later processing. When output is stored in a file, the user has the option of retaining the report headers or automatically stripping them off. AMARON displays data in either of two formats - Tables of Average Values or Histograms. Refer to the section on Annotated Sample Reports for examples of these formats.

Control-C (^C) may be used at any point to terminate the program. The program will accept responses in either upper or lower case.

.RUN AMARON

DATABASE NAME:

Requests the filename of the database from which item and subitem values are to be examined.

Valid Response:

AMAR
AMAR.DB
TODAY
TODAY.DB

AMAR and AMAR.DB refer to the system AMAR database. TODAY and TODAY.DB refer to the output file produced by the AMARSD program.

Default: Carriage return will cause the query to be repeated.

Histogram Function:

Requests whether histogram data or average values should be reported.

Valid Response:

Y - Produce histogram report.
N - Produce tabular report of average values.

Default: Carriage return will cause the query to be repeated.

Output at (T)erminal or in (F)ile:

Requests whether the output should be displayed at the terminal or stored in a file.

Valid Response:

T - terminal
F - file

Default: Carriage return will cause the query to be repeated.

File ID:

This query is displayed only if "F" is specified in response to the Output at (T)erminal or in (F)ile query. It requests the filename of the output file.

Valid Response:

Filename
Filename.ext

Default: Carriage return will cause the query to be repeated.

Start Date:

Requests the start date of the report period.

Valid Response:

yymmdd

Where yy is the normal calendar not fiscal calendar year; mm is the month; and dd is the day.

Default: Carriage return will cause the query to be repeated.

End Date:

Requests the end date of the report period.

Valid Response:

yymmdd

Where yy is the normal calendar not fiscal calendar year; mm is the month; and dd is the day. If data for only one day is required, make the end date the same as the start date.

Default: Carriage return will cause the query to be repeated.

Granularity Level:

Requests the summary (fiscal) level of the data to be reported.

Valid Response:

H - Hour
S - Sample Group Interval (same as hour)
D - Day
W - Week
M - Month
HOURS-WEEK [HRS-WEEK or H-W] - Composite Week Hours
HOURS-MONTH [HRS-MONTH or H-M] - Composite Month Hours

Composite refers to the type of data normally displayed in the Weekly/Monthly 'Typical Day' Reports. If TODAY.DB is being used, only the H or S responses are valid here.

Default: Carriage return will cause the query to be repeated.

Starting Hour:

This query requests the beginning hour of the report period. It will be displayed only if "S", "H", "HOURS-WEEK", etc. has been specified in response to the "Granularity Level:" query.

Valid Response:

nn Where nn = 01 through 24.

Default: Carriage return will cause the query to be repeated.

Ending Hour:

This query requests the ending hour of the report period. It will be displayed only if "S", "H", "HOURS-WEEK", etc. has been specified in response to the "Granularity Level:" query.

Valid Response:

nn Where nn = 01 through 24.

If data for only one hour is required, make the ending hour the same as the starting hour.

Default: Carriage return will cause the query to be repeated.

(P)rime (N)on Prime (B)oth or (W)eekend:

This query is used to further restrict the time period of the data to be displayed beyond that implied by the "Granularity Level:", "Starting Hour:", and "Ending Hour:" queries.

Valid Response:

P - Display only data marked as prime time.
N - Display only data marked as non-prime time.
B - Display both prime and non-prime data.
W - Display data only for composite weekend and holiday hours.

The "W" response will be accepted only if the composite weekend and holiday summary level has been selected under the "Granularity Level:" query.

"B" is allowed only if "H" or "S" has been specified in response to the "Granularity Level:" query.

The "B" response is not allowed if "Y" has been specified in response to the "Histogram Function:" query.

Default: Carriage return will cause the query to be repeated.

Item n:

This query is repeated up to 10 times (n = 1 to 10). Up to 10 items or subitems may be requested for display in one report.

Valid Response:

aaaa
aaaasssssss
?
aaaa?

Where aaaa is a 4 character item code and sssssss is a 7 character subitem code. See the Appendix called "System AMAR Item Definitions" for a list of item codes. The AMRGEN program can also be used to obtain the list of items and subitems for your own database. See the Section called "Examining/Changing Database Parameters (AMRGEN)".

? causes the entire list of item and subitem names to be displayed in alphabetical order.

aaaa? causes the subitem names to be listed for the item denoted by aaaa.

If ? or aaaa? is specified, the query is repeated.

Default: Carriage return immediately terminates the list of items and subitems even if none has been specified.

Suppress Blank Ranges (Y/N)?

This query requests whether or not you wish to print ranges of values with a sample count of 0. It is displayed only if "Y" has been specified in response to the "Histogram Function:" query.

Valid Response:

Y - Yes, suppress ranges with a sample count of 0.
N - No, print ranges with a sample count of 0.

Warning: It is recommended that the "Y" response normally be used, especially for items which could occasionally have one or two very large values, for example, AVRT (average scheduler response time.) Otherwise, a large amount of unnecessary data (all 0's) may be printed.

Default: Carriage return will cause the query to be repeated.

Headings on Report (Y/N)?

This query requests whether or not report headings should be produced. It is displayed only if "F" has been specified in response to the "Output at (T)erminal or in (F)ile:" query.

Valid Response:

Y - Yes, produce headings.
N - No, suppress production of headings.

If report headings are not produced, the user must develop another method of identifying to which items and subitems the reported values belong. No internal identification will be kept in the report.

Default: Carriage return will cause the query to be repeated.

More Requests (Y/N):

Requests whether or not the user wishes to extract more data.

Valid Response:

Y - Yes, repeat dialogue.
N - No, terminate program.

Default: Carriage return will cause the query to be repeated.

Blank Page

APPENDIX E

DATA EXTRACTION PROGRAM (AMAREX) DIALOGUE

AMAREX is normally run at the terminal. It can be used to access either the system AMAR database, AMAR.DB, or the output of the AMARSD program, TODAY.DB. See the Section called "Data Extraction Records" for a sample of the output and a description of the record formats.

In all queries except "DATABASE NAME:" and "OUTPUT:", carriage return may be entered as a response on subsequent passes through the dialogue. Carriage return means to re-use the last valid response given to that query.

Invalid responses to a query will cause the query to be repeated.

.RUN AMAREX

DATABASE NAME:

Requests the name of the database from which records are to be extracted.

Valid Response:

AMAR
AMAR.DB
TODAY
TODAY.DB

AMAR or AMAR.DB specifies the system AMAR database as the input file. TODAY or TODAY.DB specifies the mini-database created by the AMARSD program as the input file.

Default: None.

OUTPUT:

Requests the name of the output file which will contain the extracted records.

Valid Response:

filename
filename.ext

Care should be taken not to use any of the reserved names listed in the Appendix called "Summary of Programs and Filenames".

Default: None.

RECORD TYPE:

Requests the 2 character code which designates the type of records to be extracted.

Valid Response:

PD - Performance Detail Records
PS - Performance Summary Records
GR - Granularity Records
SC - System Uptime Log Records

Two or more record codes may also be strung together by commas.

DATE:

Requests the timeframe(s) for which you wish to extract records.

Valid Response:

yymmdd
yymmdd:hhss
yymmdd-yymmdd
yymmdd:hhss-yymmdd:hhss

Where:

yy is the actual calendar not fiscal year
mm is the month
dd is the day
hh is the hour
ss are the minutes

Two or more of the above date and time specifications may be strung together by commas.

All dates and times should be entered in chronological order.

Only data for a fiscal period which terminates within the specified timeframes will be extracted, regardless of the beginning date and time of the fiscal period. (See the "GRANULARITY LEVEL:" query).

Hourly data is the lowest level of data which may be extracted. To select a specific hour, specify that hour as the start time and/or end time of the interval. For example, to extract data for the hour ending at 2 AM on February 4, 1982, specify 820204:0200.

All time periods, except for hours, are assumed to end at midnight. You only need to specify the end date of the fiscal period for which data is to be extracted.

Whenever time is not specified, the start time is assumed to be 0001 of the first day and the end time is assumed to be 2400 of the last day.

GRANULARITY LEVEL:

This query requests the level of fiscal period to be extracted. It is displayed only if "PD", "PS" or "GR" has been specified in response to the "RECORD TYPE:" query.

Valid Response:

SGI [or S] - Sample Group Interval (hours)
DAY [or D]
WEEK [WK or W]
MONTH [MO or M]
HOURS-WEEK [HRS-WEEK or H-W] - Composite Week Hours
HOURS-MONTH [HRS-MONTH or H-M] - Composite Month Hours

Two or more of the above responses may also be strung together by commas.

Composite refers to those records which are displayed in the Weekly/Monthly 'Typical Day' Reports.

RESTRICTING ANY FISCAL PERIOD?

This query asks whether or not you wish to include or exclude any fiscal periods from the timeframe specified in the "DATE:" query. It is displayed only if "PD", "PS" or "GR" has been specified in response to the "RECORD TYPE:" query.

Valid Response:

Y - Triggers further queries used to specify the fiscal periods to be selected.

N - No restrictions on fiscal periods.

The fiscal calendar is defined within the system AMAR database as follows:

Each hour of the day is defined as a Sample Group Interval (SGI) numbered 1 through 24.

Each day of the week is assigned a number from 1 to 7, Sunday through Saturday.

Each week in a fiscal month is assigned a number from 1 to 4 for the first 2 months in a fiscal quarter and from 1 to 5 for the third month in the quarter. In a fiscal leap year, the last month will have from 1 to 6 fiscal weeks.

Each month in a fiscal quarter is assigned a number from 1 to 3.

Each fiscal quarter is assigned a number from 1 to 4.

Refer to the Appendix called "Fiscal Calendar" for an example of how to relate the fiscal calendar to a normal calendar and to AMAR filenames.

SGI PERIOD:

This query requests the range of hours you want included or excluded from the timeframes specified in the "DATE:" query. It is displayed only if "Y" has been specified in response to the "RESTRICTING ANY FISCAL PERIOD?" query.

Valid Response:

hhss-hhss	Where hh = hours; ss = minutes.
hhss-hhss,...,hhss-hhss	
VOID	Void any previously specified SGI restrictions.

Default: Carriage return <CR> on the first pass through the dialogue causes the next query to be displayed.

DAY PERIOD:

This query requests the range of fiscal days you want included or excluded from the timeframes specified in the "DATE:" query. It is displayed only if "Y" has been specified in response to the "RESTRICTING ANY FISCAL PERIOD?" query.

Valid Response:

n	Where n = 1,2,... or 7.
1,...,7	
1-7	
VOID	Void any previously specified day restrictions.

Default: Carriage return <CR> on the first pass through the dialogue causes the next query to be displayed.

WK PERIOD:

This query requests the range of fiscal weeks you want included or excluded from the timeframes specified in the "DATE:" query.

It is displayed only if "Y" has been specified in response to the "RESTRICTING ANY FISCAL PERIOD?" query.

Valid Response:

n	Where n = 1,2,... or 5.
1,...,5	
1-5	
VOID	Void any previously specified week restrictions.

Default: Carriage return <CR> on the first pass through the dialogue causes the next query to be displayed.

MO PERIOD:

This query requests the range of fiscal months you want included or excluded from the timeframes specified in the "DATE:" query. It is displayed only if "Y" has been specified in response to the "RESTRICTING ANY FISCAL PERIOD?" query.

Valid Response:

n	Where n = 1, 2 , or 3.
1,2,3	
1-3	
VOID	Void any previously specified month restrictions.

Default: Carriage return <CR> on the first pass through the dialogue causes the next query to be displayed.

QTR PERIOD:

This query requests the range of fiscal quarters you want included or excluded from the timeframes specified in the "DATE:" query. It is displayed only if "Y" has been specified in response to the "RESTRICTING ANY FISCAL PERIOD?" query.

Valid Response:

n	Where n = 1, 2, ... or 4.
1,...,4	
1-4	
VOID	Void any previously specified quarter restriction.

Default: Carriage return <CR> on the first pass through the dialogue causes the next query to be displayed.

YR PERIOD:

This query requests the date(s) of the fiscal year(s) you want included or excluded from the timeframes specified in the "DATE:" query. Note that fiscal year granularity records are not normally kept in the database. Thus, you may not get any output when using this query. This query is displayed only if "Y" has been specified in response to the "RESTRICTING ANY FISCAL PERIOD?" query.

Valid Response:

nn	Where nn = 78, 79, ... , 99.
78,...,99	
78-99	
VOID	Void any previously specified year restriction.

Default: Carriage return <CR> on the first pass through the dialogue causes the next query to be displayed.

ITEM:

This query requests the 4 character item code and the 7 character subitem code (see the Appendix called "System AMAR Item Definitions") of any items or subitems you wish to extract. Use the AMRGEN program to get a list of all the items and subitems contained in your database. This query is displayed only if "PD" or "PS" has been specified in response to the "RECORD TYPE:" query.

Valid Response:

aaaa
aaaasssssss
aaaa,....,aaaasssssss
ALL

Where aaaa is a 4 character item code and sssssss is a 7 character subitem code. If the item has subitems and you specify only the 4 characters of the item code, all subitems will be extracted. Items and subitems will be extracted in alphabetical order.

PRIMETIME:

This query requests the code which identifies the type of data (prime, non-prime, weekend, or composite ('Typical Day')) that you want extracted. It is displayed only if "PD" or "PS" has been specified in response to the "RECORD TYPE:" query.

Valid Response:

P - Prime Time Data
N - Non-prime Time Data
N-P - Both Non-prime and Prime Data
W - Weekend and Holiday Hours for Composite Data
ALL - Prime, Non-prime, and Weekend and Holiday Data

The "N-P" and "W" responses will be accepted only if "HOURS-WEEK" or "HOURS-MONTH" were specified in response to the "GRANULARITY LEVEL:" query.

The "N-P" response will cause data to be included for weekdays and excluded for weekends and holidays.

The "W" response will cause data to be excluded for weekdays and included for weekends and holidays.

The recommended way of extracting weekday and weekend and holiday composite data is to run through the AMAREX dialogue twice, once with a response of "N-P" to extract weekday composite data and once with a response of "W" to extract weekend and holiday composite data.

Following the "PRIMETIME:" query, the message [EXTRACTING] will be displayed. This message indicates that the extraction process has begun.

Once processing is complete, and the output file closed, the message [SPECIFY NEXT EXTRACTION CRITERIA] will be displayed. The dialogue will be repeated starting with the "OUTPUT:" query. Further extraction requests may be entered.

Except for input and output file specifications, all selection criteria will remain in effect. To retain the selection criteria for a specific query, hit carriage return in response to that query. To override any previously supplied selection criteria, enter new explicit values. To cancel selection criteria for the "SGI PERIOD:" through "MO PERIOD:" queries, respond "VOID".

To enter exclusions, preceed the value by the phrase 'NOT'. For example, the response "ALL,'NOT'LUFS,'NOT'LUWQDSKEO" to the "ITEM:" query causes data to be extracted for all items and subitems except LUFS (logical unit free space) and LUWQ (logical unit wait queue) for DSKE. As a guideline, specify the most inclusive response first, then exclusions. If ALL is used as a response, it must be the first in the string of responses to the query. Exclusion only works if a specific reply has aready been made to the query. For example, "'NOT'1" is a valid response to the "DAY PERIOD:" query only if "1,...,7" or "1-7" had already been specified in response to that query.

Blank Page

APPENDIX F

REPORT PROGRAM (WCRPT, WCRPTB, AND WCRPTC) DIALOGUE

This appendix explains, for each reporting program each possible prompt, its valid response, and any defaults.

Error messages which may occur while running WCRPTB, WCRPTC or WCRPT are listed in AMAR-10 Error Messages. The dialogue for WCRPTC is identical to that of WCRPTB. The WCRPTC program just contains larger internal arrays for more detail processing. However, some prompts and/or valid responses differ between WCRPTB and WCRPT. In these cases, WCRPTB will be discussed first and "WCRPTB only", or "WCRPT only" will be indicated in parentheses after the text of the prompt. The rest of this section is the description of each type of prompt in the WCRPTB, WCRPTC and WCRPT dialogues. Prompts are listed in approximately the same order that they appear in the dialogue.

.RUN WCRPTB (or .RUN WCRPTC, .RUN WCRPT)

REPORT DESCRIPTON =

Requests a free-form English description to be printed in the report header box.

Valid Response:

Arbitrary string of up to 90 characters, including blanks and punctuation.

Default: Carriage return says leave description blank.

INPUT FILE.EXT = (WCRPTB only).

Requests the name and extension of a workload database file which contains data for the period to be reported.

Valid Responses:

yyqmwd.DB0	for a weekday daily file
yyqmwd.DB1	for a weekend daily file

yyqmw.DB0 for a weekday weekly file

yyqmw.DB1 for a weekend weekly file

yyqm.DB0 for a weekday monthly file

yyqm.DB1 for a weekend monthly file

Where yy = fiscal year, q = fiscal quarter, m = fiscal month within the quarter, w = fiscal week within the month, d = day of the week (Sunday = 1).

Note: For purposes of automatic reporting, the filename (not the extension) may be implied by an appropriate number of question marks. Six question marks means a daily file, five questions marks means a weekly file, and four question marks means a monthly file. However, Sunday is normally the only day when weekly and monthly files are ready for automatic reporting. Therefore, ?????? is the only "wild name" useful for interactive dialogue with WCRPTB. Six question marks normally means yesterday, but could refer to a prior day if multiple days were input.

Note: Dialogue accepts filename and extension, not a complete file specification.

Default: None.

OUTPUT FILE.EXT = (WCRPTB only. See WCRPT below).

Requests the name and extension of the desired report file.

Valid Responses:

filename.ext

The filename (exclusive of extension) is arbitrary, but certain conventions are followed in the daily stream. (See the Section called "Procedure for Running WCRPTB.CTL".)

Any question mark in the name is replaced by the corresponding character of the input name.

Certain extensions such as .DB0, .DB1, .IN0, .IN1, .RA0, ,RA1, and .CON are not accepted by the program.

Extensions such as .RP1, .RP2, .RP3, and .RP4 are not recommended except in the daily stream.

Note: The dialogue accepts only filename and extension, not a complete file specification.

Default: None.

OUTPUT FILE.EXT = (WCRPT only. See WCRPTB above).

Requests the name and extension of the desired report file.

Valid Responses:

filename.ext

Before you run WCRPT, you should make sure you have a pair of daily files with the extensions of .INO and .IN1. These are expected as the input files for WCRPT. WCRPT will not prompt for an input filename; it will assume the existence of .INO and .IN1 files. .INO and .IN1 files are normally created by WCINC in the nightly batch stream. They may also be created for today's data by running WC before you run WCRPT. As a result, there may be .INO and .IN1 files for both yesterday and today. If you have run WC to preprocess today's data, the filename will be WC. Otherwise, you should do a directory in order to determine the precise 6 character filename of the .INO and .IN1 files which you want.

The filename must be identical to the filename of your input files which have extensions .INO and .IN1. The extension should be other than .INO, .IN1, .RAW, or .EXE.

Note: The dialogue accepts only filename and extension, not a complete file specification.

Default: None.

PPN GROUPING FILE.EXT =

Requests name and extension of desired PPN grouping file.

Valid Responses:

Carriage return or filename and extension of a correctly formatted PPN grouping file. (See the Appendix called "Grouping PPN's for Reporting Purposes".)

Default: Carriage return means do not group PPN's.

ENTER DESIRED START AS HH MM: (WCRPTB only. See WCRPT below).

Requests the time of the beginning of the first subreport interval.

Valid Responses:

Carriage return or 1 or 2 integers of the form:

hh mm

Where hh represents the hour, mm represents the minute.

If only one number is specified, it will be taken as the hour.

Since hourly data is the finest granularity in the database, only hours 0 through 23 are valid start times. The minutes, if specified, must be 0.

Default: Carriage return (or 0) says start at midnight.

ENTER DESIRED START AS HH MM SS DD: (WCRPT only. See WCRPT above).

Requests the time (and relative day) of the beginning of the first reporting interval.

Valid Responses:

Carriage return or 1 to 4 integers of the form:

hh mm ss dd

Where hh represents the hour, mm represents the minutes, ss represents the seconds, and dd represents the number of the day at which you wish to begin the report.

Days are numbered 0, 1, 2, etc. starting with the first day of the file as day 0.

If only one number is specified, it will be taken as the hour; two numbers separated by a blank will be taken as the hour and minutes; and so forth.

For example, if there are three days worth of data in the file for September 8th, 9th, and 10th: 9 30 0 1 means start at 9:30 on the second day of the file (September 9th).

Any date and time before the end time of the input file may be specified. For purposes of synchronization, it is possible to specify a start time before the start time of the input file.

Note:

If you want to start at the beginning of the first day in the input file, respond 0 0 1 which implies 1 second after midnight and is distinct from 0, 0 0, 0 0 0, and 0 0 0 0 which are all equivalent to carriage return and imply start at start time of the file.

Default: Carriage return means start at the start time of the input file.

ENTER DESIRED END AS HH MM: (WCRPTB only. See WCRPT below).

Requests the end time of the last reporting interval.

Valid Responses:

Carriage return or 1 or 2 integers of the form:

hh mm

Where hh represents the hour, mm represents the minutes.

If only one number is specified, it will be taken as the hour.

Since hourly data is the finest granularity in the database, only hours 1 through 24 are valid end times. Another constraint is that end time must be greater than start time. The minutes, if specified, must be 0.

Default: Carriage return (or 0) says end at midnight.

ENTER DESIRED END AS HH MM SS DD: (WCRPT only. See WCRPTB above).

Requests the time (and relative day) of the end of the last reporting interval.

Valid Responses:

Same format as the start time for WCRPT.

Any time after the start time of the file and after the specified start time.

Default: Carriage return means stop the last report interval at the end time of the input file.

ENTER DESIRED INTERVAL SIZE AS HH MM: (WCRPTB only. See WCRPT below).

Requests the size of the subreport interval.

Valid Responses:

Carriage return or 1 or 2 integers of the form:

hh mm

Where hh represents hours, and mm represents minutes.

If only one number is specified, it will be taken as hours.

Since hourly data is the finest granularity in the database, only an integral number of hours between 1 and 24 is acceptable. Minutes, if specified, must be 0.

Note: If the requested interval size <u>does not</u> divide evenly into the time between start time and end time, the last subreport will be short.

Default: Carriage return (or 0) requests that the entire interval from start time to end time should be reported in a single subreport.

ENTER DESIRED INTERVAL SIZE AS HH MM SS DD: (WCRPT only. See WCRPTB above).

Requests the size of the reporting interval.

Valid Responses:

A time and date specification like WCRPT's start time. dd is the number of whole days in the interval.

The interval specified may be any positive interval not significantly less than the average checkpoint interval.

NOTE: When there is a gap in the data which is longer than the report interval (because of a system crash, for example), reports covering single checkpoint intervals will be generated while "catching up". This applies only to WCRPT, not to WCRPTB.

Default: Carriage return means report everything between the specified start time and the specified end time as a single report interval.

ENTER MAXIMUM DETAIL LINES PER INTERVAL:

Requests the maximum number of detail lines you wish printed for any report interval.

Valid Responses:

Carriage return or any integer between 1 and 512 (inclusive).

Default: Carriage return (or 0) means print all detail lines.

Note: You should sort your data by at least one significant resource if you use this type of cutoff; otherwise, you may suppress significant detail lines.

ENTER CPU% CUTOFF:

Requests the minimum percentage of CPU time which qualifies a detail line to be printed.

Valid Responses:

Carriage return or a number between .01 and 100.00.

Default: Carriage return (or 0) means print all detail lines.

Note: This form of cutoff exists only for CPU%. It is independent of sort order. It may even be used simultaneously with a "MAXIMUM DETAIL LINES" cutoff, although the results should be interpreted carefully.

ANY SPECIAL MASKS OR SORT ORDERS? (Y or N):

Allows you to indicate whether or not you wish to use the special masking and/or sorting features described in the Appendix called "Special Masks and Sort Orders".

Valid Responses:

Carriage return or "Y" or "N".

Default: Carriage return (or anything that does not begin with "Y") means no special masks or sort orders.

ID ITEM 0-0:

Requests the first item to be held constant while building detail lines. These items form the "group" for which resource usage will be summarized. A very long list of grouping or sorting items will be truncated after 100 characters in the headings. The truncated portion could include "CUTOFF" criteria.

Valid Responses: Any of the grouping item codes listed in the Appendix called "Valid Grouping and/or Sort Items" (excluding items to be used only for sorting).

Default: None. ID ITEM 0-0: must be specified.

ID ITEM 0-1: (through ID ITEM 0-8:)

Requests the second (through ninth) item to be held constant while building detail lines.

Valid Responses: Carriage return or as above for ID ITEM 0-0:.

Default: Carriage return means no more items are to be held constant.

ID ITEM 0-9:

This query is used to terminate building of the list of grouping items.

Valid Responses: Carriage return only.

Default: No more items are to be held constant.

SORT ITEM 1-0:

Requests the major sort item for the first (usually only) subreport for a given interval.

Valid Response: Carriage return or any of the codes described in the Appendix called "Valid Grouping and/or Sort Items".

Default: Carriage return means produce only one subreport for a given interval, with items sorted as they were grouped.

SORT ITEM 1-1: (through SORT ITEM 1-8:)

Requests a sub-sort item for the first (usually only) subreport for a given interval.

Valid Response: Carriage return or any of the codes described in the Appendix called "Valid Grouping and/or Sort Items".

Default: Carriage return means no additional sub-sort items are to be specified.

SORT ITEM 1-9:

This query is used to terminate building the list of sort items for the first (usually only) subreport for a given interval.

Valid Response: Carriage return only.

Default: No more sort items are to be specified.

SORT ITEM 2-0: (through SORT ITEM 8-0:)

Requests the major sort item for the second (through eighth) subreport for a given interval).

Valid Response: Carriage return or any of the codes described in the Appendix called "Valid Grouping and/or Sort Items".

Default: Carriage return means that no additional subreports for a given interval are to be produced.

SORT ITEM 2-1: (through SORT ITEM 2-8:)

Analagous to SORT ITEM 1-1: (through SORT ITEM 1-8:).

SORT ITEM 2-9:

Analagous to SORT ITEM 1-9:.

SORT ITEM 3-x: (through SORT ITEM 8-x:) (where x = 1 to 8)

Analagous to SORT ITEM 1-1: (through SORT ITEM 1-8:).

SORT ITEM 3-9: (through SORT ITEM 8-9:)

Analagous to SORT ITEM 1-9:.

SORT ITEM 9-0:

This query is used to terminate the building of subreports for each interval.

Valid Responses: Carriage return only.

Default: Carriage return means no additional subreports for a given interval are to be produced.

SORT ITEM 9-1: (through SORT ITEM 9-9:)

This query should not be reached; if it is encountered, you have specified too many lists of sort items and you will have to terminate the run and start over.

Valid Response: Control-C.

Default: If you enter a carriage return you will get a fatal error message and the run will terminate.

ID MASK 0-y: (where y = 0 to 8)

Allows you to specify a special mask for grouping as described in the Appendix called "Special Masks and Sort Orders".

Valid responses: Carriage return or a string of 1 to 12 octal digits.

Default: Carriage return (or an all zero octal mask) will cause the default mask to be used.

ID ORDER 0-y: (where y = 0 to 8)

Allows you to specify a special sort order as described in the Appendix called "Special Masks and Sort Orders".

Valid Response: Carriage return (for the default sort order). Any response starting with "A" (for ascending order). Any response starting with "D" (for descending order).

Default: Carriage return means the default order should be used.

SORT MASK x-y: (where x = 1 to 8, y = 0 to 8)

Allows you to specify a special mask as described in the Appendix called "Special Masks and Sort Orders".

Valid responses: Carriage return or a string of 1 to 12 octal digits.

Default: Carriage return (or an all zero octal mask) will cause the default mask to be used.

SORT ORDER x-y: (where x = 1 to 8, y = 0 to 8)

Allows you to specify a special sort order as described in the Appendix called "Special Masks and Sort Orders".

Valid Responses: Carriage return (for the default sort order). Any response starting with "A" (for ascending order). Any response starting with "D" (for descending order).

Default: Carriage return means the default order should be used.

MORE REPORTS? (Y OR N):

This query permits you to specify additional reports in the same run.

Valid Responses: Carriage return or "Y" or "N".

Default: Carriage return causes the program to terminate.

APPENDIX G

VALID GROUPING AND/OR SORT ITEMS

The following items (1) may be used for both grouping and sorting when using the WCRPT, WCRPTB, and WCRPTC programs. Any of these items used for sorting should also be used for grouping. Item codes must be spelled exactly as shown here; no other abbreviations are allowed.

MNE-MONIC	DEFLT. ORDER	DEFLT. MASK	DESCRIPTION
JOB	A	000000000377	Job number
PPN	A	777777777777	Project/programmer number
PROJ	A	777777000000	Project half of PPN
PROG	A	000000777777	Programmer half of PPN
CHARG	A	777777777777	Charge number
PRGRM	A	777777777777	Program name
HPQ (2)	D	374000000000	High priority queue number
CLASS(2)	A	003740000000	Scheduler class number
SCD(2)	A	003740000000	Scheduler class number
CLS(2)	A	003740000000	Scheduler class number
CATEG	A	000037400000	LOGIN category
CT	A	000037400000	LOGIN category
BATCH	A	400000000000	Batch or timesharing
STATE	A	777777777777	Job state
TTY	A	777777777777	Terminal name
NODE	A	777777777777	Node name
LINE	A	000000377400	Line number on node
LOGIN	A	777777777777	Date/time of login
MISC(3)	A	777777777777	Miscellaneous identification
NOW	A	777777777777	Date/time stamp of sample

The following items should be used only for sorting:

JOBS	D	777777777777	Average simultaneous jobs
LIO	D	777777777777	LOGIN's and KJOB's
IN	D	777777000000	LOGIN's during interval
OUT	D	000000777777	KJOB's during interval
KCT(4)	D	777777777777	Kilo-core ticks
CPU%(5)	D	777777777777	Percent of processor used
UUOS(6)	D	777777777777	UUO's/second
READS(7)	D	777777777777	Disk blocks read/second

```
WRITE(7)   D       777777777777      Disk blocks written/second
```

Notes:

1. Items (and valid synonyms) are listed as they appear from left to right across the report.

2. Since high priority queue number overrides scheduler class number in the report, HPQ should always be used immediately before CLASS if CLASS is going to be used for grouping and sorting.

3. The code MISC refers to a word which contains six items (BATCH, HPQ, SCD, CATEG, LINE, and JOB). This code should only be used with a mask as described in the Appendix called "Special Masks and Sort Orders".

4. Kilo-core ticks is a valid sort item, although not reported directly. It may be derived by multiplying PAGES X .5 X CPU% X interval size expressed in ticks. Note that current program logic does not allow sorting by jobsize (PAGES), a derived item.

See the Appendix called "System AMAR Item Definitions" for references 5-7:

5. See "% CPU UTIL" for the variable representing system wide CPU utilization.

6. See "USER UUOS/SEC" for the variable representing the system wide UUO count.

7. See "USER DSK BLKS/SEC" for the variable representing system wide reads and writes. See "PK name BLKS/SEC" and "UN name BLKS/SEC" for the variables representing system wide reads and writes on individual disk packs and physical units.

APPENDIX H

GROUPING PPN'S FOR REPORTING PURPOSES

When using the WCRPT, WCRPTB, or WCPRTC programs, arbitrary groups of PPN's may be combined for reporting purposes through the use of a PPN-grouping file. Figure H-1 shows a dialogue using a PPN-grouping file and a portion of the report it generated.

Each record in a PPN-grouping file has a "before" PPN and an "after" PPN.

When WCRPT finds a PPN in the data which matches one of your "before" PPN's, it replaces that PPN with the corresponding "after" PPN.

If desired, the last "before" PPN in your list may be 0,777777, a special PPN which "matches" any PPN not found earlier in the list and allows you to create an "other" category.

If you have no "other" category, any PPN's not found in the "before" list will be left unchanged.

Both "before" and "after" PPN's must be octal, and not greater than 377777, 777777.

It is probably a good idea to make "group" PPN's distinctively different from actual PPN's.

The following sample PPN-grouping file will group [123,456], [234,567], and [345,670] as [0,1]; [456,701] and [567,012] as [0,2]; [1,2] as [1,2]; and all other PPN's as [0,3]. Note that "before" and "after" PPN's are entered without brackets, separated by a blank. "Wildcarding" with question marks and asterisks is not supported.

```
123,456 0,1
234,567 0,1
345,670 0,1
456,701 0,2
567,012 0,2
1,2 1,2
0,777777 0,3
```

```
.ru worptb

REPORT DESCRIPTION = report splitting [1,2] usage from all other usage (1)

INPUT  FILE.EXT = 823242.db0 (2)

OUTPUT FILE.EXT = workpp.rpt (3)

PPN GROUPING FILE.EXT = ppn.fil (4)

ENTER DESIRED START AS     HH MM: (5)
ENTER DESIRED END   AS     HH MM:
ENTER DESIRED INTERVAL SIZE AS  HH MM: 8 (6)

ENTER MAXIMUM DETAIL LINES PER INTERVAL:
ENTER CPU% CUTOFF: (7)

ANY SPECIAL MASKS OR SORT ORDERS? (Y OR N): n (8)

ID ITEM    0-0: ppn (9)

ID ITEM    0-1:

SORT ITEM  1-0:

MORE REPORTS? (Y OR N): n

END OF EXECUTION
CPU TIME: 1.35  ELAPSED TIME: 11.67
EXIT

.ty ppn.fil
1,2 1,2            (10)
0,777777 0,7
```

1. Free form report description. This report will summarize usage in two detail lines, one showing all [1,2] jobs grouped together and the other showing all other jobs grouped together.

2. Daily file for February 15,1982 (FY82, third quarter, second month, fourth week, second day.)

3. Report filename.

4. A special PPN grouping file will be used. See 10. below for the listing of this file.

5. The report will start at the beginning of the 823242.db0 file and stop at the end of it.

6. The report will contain subreport intervals of 8 hours each.

7. All detail lines (in this case a maximum of 2 per subreport interval) will be shown. No cutoffs will be used.

8. No other special reporting features will be used.

9. The data will be grouped and sorted by PPN.

10. The PPN grouping file. The group of [1,2] jobs will show up as [1,2] on the report. The group of all other jobs will show up as [,7].

Figure H-1
Dialogue Using a PPN Grouping File

```
**************************************************************************************************************
*                                                                                                            *
*                                        AMAR WORKLOAD REPORT                                                *
*                                                                                                            *
*   SITE:  <Put Any Title Here>                                                        SYSTEM:  PATH         *
*                                                                                                            *
*   REPORT DESCRIPTION:  report splitting [1,2] usage from all other usage (1)                               *
*                                                                                                            *
*   INPUT FILE:  823242.DBO (2) (FISCAL YEAR: 82   QUARTER: 3   MONTH: 2   WEEK: 4   DAY: 2   MONDAY)        *
*                                                                                                            *
**************************************************************************************************************

             (5)                                (5)                                     (6)
FROM:   0: 0: 2  ON  MONDAY    15-FEB-82    TO:   8: 0: 2  ON  MONDAY    15-FEB-82    INTERVAL:   7:59:59    MEASURED: 100%

GROUPED BY:    PPN
                      (9)
SORTED BY:     PPN

JOB  AVG  LOG LOG ------PPN----- CHARGE PRGRM  PAGES    CPU%    UUOS  READS WRITES SCD CT B STATE  TTY     NODE--LINE LOGIN AT
 #   JOBS  IN OUT    PROJ  PROG  NUMBER  NAME  (AVG)            /SEC   /SEC  /SEC CLS                                DAY TIME

    22.1  13   5 ******INTERVAL TOTALS******    52.6     0.8    11.8    0.6    0.4

     2.1  12   4 [            7] ?????? ??????  89.6     0.4     4.7    0.4    0.3          ??      ?TY??? ?????
    20.0   1   1 [      1     2] ?????? ??????  23.5     0.5     7.1    0.1    0.2          ?? ??   ?????? ?????
                       (10)

FROM:   8: 0: 2  ON  MONDAY    15-FEB-82    TO:  16: 0: 3  ON  MONDAY    15-FEB-82    INTERVAL:   8: 0: 0    MEASURED: 100%

GROUPED BY:    PPN

SORTED BY:     PPN

JOB  AVG  LOG LOG ------PPN----- CHARGE PRGRM  PAGES    CPU%    UUOS  READS WRITES SCD CT B STATE  TTY     NODE--LINE LOGIN AT
 #   JOBS  IN OUT    PROJ  PROG  NUMBER  NAME  (AVG)            /SEC   /SEC  /SEC CLS                                DAY TIME

    56.8 224 183 ******INTERVAL TOTALS******    57.0    37.9   265.7   41.3   18.5

    35.4 214 177 [            7] ?????? ??????  59.6    34.9   194.8   39.8   18.0          ?? ??   ?????? ?????
    21.5  10   6 [      1     2] ?????? ??????  25.7     2.9    70.9    1.5    0.6          ?? ??   ?????? ?????


FROM:  16: 0: 3  ON  MONDAY    15-FEB-82    TO:   0: 0: 2  ON  TUESDAY   16-FEB-82    INTERVAL:   7:59:59    MEASURED: 100%

GROUPED BY:    PPN

SORTED BY:     PPN

JOB  AVG  LOG LOG ------PPN----- CHARGE PRGRM  PAGES    CPU%    UUOS  READS WRITES SCD CT B STATE  TTY     NODE--LINE LOGIN AT
 #   JOBS  IN OUT    PROJ  PROG  NUMBER  NAME  (AVG)            /SEC   /SEC  /SEC CLS                                DAY TIME

    28.4  63 111 ******INTERVAL TOTALS******    34.8    13.1   103.8   32.4    9.9
                                      .
                                      .
                                      .
```

Figure H-1 (continued)

Blank Page

APPENDIX I

SPECIAL MASKS AND SORT ORDERS

In certain cases, it may be desirable to group items by a part of one of the standard items, for example, the first three digits of the charge number or the last two characters of the program name. This is possible when using the WCRPT, WCRPTB, or WCRPTC programs if you answer "Y" to the prompt

ANY SPECIAL MASKS OR SORT ORDERS? (Y OR N):

and answer the special prompts appropriately.

Selection of a part of a standard item is done by means of a mask, which is a computer word with "1" bits corresponding to the bits of the item you want to use and "0" bits corresponding to the bits you want to ignore. We represent a mask as 12 octal digits, each representing a group of 3 bits.

For items stored in SIXBIT (such as CHARGE, PRGRM, STATE, TTY, and NODE) each character you want to use is represented by a pair of 7's in the mask and each character you want to ignore is represented by a pair of 0's. Hence, the mask for the first three characters of the charge number is 777777000000. The mask for the last 2 characters of the program name is 000000007777, which may be abbreviated as 7777, since leading zeroes are implied.

Six of the items (JOB, BATCH, LINE, HPQ, SCD, and CATEG) are all packed together in a single word known as MISC. You can see how they fit together if you examine their default masks in the Appendix called "Valid Grouping and/or Sort Items". If you want to use a part of one of these, the mask you use should have a subset of the "1" bits implied in the default mask. In other words, the mask you enter is used for the whole word, not just the item you named.

Figure I-1 shows a dialogue using a special mask and the beginning of the report it generated.

The Appendix called "Valid Grouping and/or Sort Items" shows that each item has a default sort order associated with it (A = ascending, D = descending). If you wish to specify the opposite order for some item, answer "Y" to: "ANY SPECIAL MASKS OR SORT ORDERS? (Y OR N):".

```
.ru worptb

REPORT DESCRIPTION = report by last two digits of program name   1

INPUT  FILE.EXT = 823242.db0   2

OUTPUT FILE.EXT = worksm.rpt   3

PPN GROUPING FILE.EXT =

ENTER DESIRED START AS     HH MM: 8    4
ENTER DESIRED END   AS     HH MM: 16    5
ENTER DESIRED INTERVAL SIZE AS  HH MM:    6

ENTER MAXIMUM DETAIL LINES PER INTERVAL: 15   7
ENTER CPU% CUTOFF:   8

ANY SPECIAL MASKS OR SORT ORDERS? (Y OR N): y   9

ID ITEM     0-0: prgrm   10
ID MASK     0-0: 000000007777   11
ID ORDER    0-0:

ID ITEM     0-1:

SORT ITEM   1-0: cpu%   12
SORT MASK   1-0:
SORT ORDER  1-0:

SORT ITEM   1-1:

SORT ITEM   2-0:

MORE REPORTS? (Y OR N): n   13

END OF EXECUTION
CPU TIME: 1.09  ELAPSED TIME: 13.97
EXIT
```

1. Free form report description. This report will summarize usage by program, grouping the programs by the last two digits of the program name.

2. Daily file for February 15,1982 (FY82, third quarter, second month, fourth week, second day.)

3. Report filename.

4. Start the report at 8:00 AM. Minutes, seconds, and day default to 0 if only the hour is specified. If carriage return only had been entered, the report would have started at the beginning of the file .

5. The report will stop at 16:00 PM (4:00 PM). If carriage return only had been entered, the report would have stopped at the end of the file .

6. Since carriage return was entered, the report will cover the entire period between 8:00 AM and 4:00 PM.

7. At most 15 detail lines on programs grouped by their last two digits will be printed.

8. No CPU cutoff will be used. Up to 15 detail lines will be printed regardless of how much CPU the detail line contains. Note that since the report is sorted by CPU percent (12. below), the top 15 users of the CPU will be shown on the report.

9. The special mask feature will be used.

10. The data will be grouped by program (prgrm) name.

11. Program names will also be grouped by their last two digits. A pair of 0's means ignore the corresponding character, a pair of 7's means use the character for grouping.

12. The detail lines will be listed with the heaviest CPU users first.

13. If "y" had been specified, the entire program dialogue would be repeated and you would be able to specify more reports.

Figure I-1
Dialogue Using a Special Mask

```
****************************************************************************************************
*                                                                                                  *
*                                   AMAR WORKLOAD REPORT                                           *
*                                                                                                  *
*   SITE:  <Put Any Title Here>                                              SYSTEM:  PATH         *
*                                                                                                  *
*   REPORT DESCRIPTION:  report by last two digits of program name (1)                             *
*                                                                                                  *
*   INPUT FILE:  823242.DBO (2) (FISCAL YEAR: 82  QUARTER: 3  MONTH: 2  WEEK: 4  DAY: 2  MONDAY)   *
*                                                                                                  *
****************************************************************************************************

          (4)                                       (5)                               (6)
FROM:   8: O: 2  ON  MONDAY    15-FEB-82    TO:  16: O: 3  ON  MONDAY    15-FEB-82    INTERVAL:   8: O: O   MEASURED: 100%
GROUPED BY: (10) PRGRM C00000007777 (11)
SORTED BY:     CPU%  (12) CUTOFF:  15 LINES (7)

JOB  AVG  LOG LOG ------PPN-----  CHARGE PRGRM  PAGES   CPU%   UUOS  READS WRITES SCD CT B STATE   TTY    NODE--LINE LOGIN AT
 #   JOBS  IN OUT    PROJ  PROG   NUMBER  NAME  (AVG)          /SEC   /SEC   /SEC CLS                                 DAY TIME

     56.8 224 183 ******INTERVAL TOTALS******   57.0    37.9  265.7   41.3   18.5

     24.6 131 124 [????????????] ?????? ????    54.0    13.2  183.2   29.1   13.9            ?? ??  ?????? ?????
      0.3   1   0 [ ????????2???] ?????? COBOL   58.1   12.6    0.8    0.2    0.2   4  O T ??     TTY??  NODED         O
 32   0.0   1   1 [  15152112157] 216300 0010DE  64.0    2.9    0.1    0.1    0.0   5  O B R2     PTY12                O  O  8:23
      1.7  16  11 [ ???????????] ?????? DIRECT  42.5    2.2   11.2    2.7    0.6        T ?? ??  ?????? ?????
      0.4   0   2 [ ???????????] ?????? BACKUP  22.6    1.1    7.5    0.8    1.5          ?? ??  ?TY??  ?????         O
      1.4   6   3 [ 15375112477] 555571  1022FD 141.6    0.9    1.4    0.5    0.0   4  O T ?? ??  ?????? ?????         O
      4.7   2   0 [ ???????????] ?????? ???SPL  30.2    0.6   23.4    0.3    0.0        T ?? ??  ?TY??? ?????
  6   0.1   0   0 [  15152112157] 216300 0020CC 113.6    0.6    0.6    0.3    0.1   4  O T ??     TTY17  NODED     17  O  8:15
  1   1.0   0   0 [      1     2]        FILDAE  14.0    0.5    2.6    0.2    0.0  -1  O T ?? ??  DET313               O -4 23:18
      2.1   0   0 [ ?0?????????] ?????? ????ON  43.4    0.5    7.2    0.3    0.3        T ?? ??  ?????? ?????
      2.0   8   6 [ ????????1??] ?????? ????C   68.1    0.3    2.7    0.3    0.1   4    T ?? ??  ???1?? ?????
      0.3   0   0 [ 15152 4????] 216?00 ?0??HR  71.6    0.3    1.8    0.7    0.5   4  O T ?? ??  ?????  ?????         O
      0.1   0   2 [  ??0???????] 13310? ????T   29.8    0.2    0.4    0.0    0.0   4    T ??     TTY12? NODE?         O
      0.1   1   1 [  12400132150] 750244 TN0003  55.6    0.2    1.4    1.6    0.0   4  O T ??     TTY1?? NODEM         O
      0.4   6   6 [ ???????????] ?????? ????NT  40.5    0.2    1.7    1.2    0.7          ?? ??  ?TY??? ?????

     39.2 172 156 ***SUBTOTALS THRU CUTOFF***   56.7    36.4  245.8   38.4   17.9

     17.7  52  27 ***SUBTOTALS AFTER CUTOFF**   62.2     1.5   19.8    2.9    0.6
```

Figure I-1 (continued)

Blank Page

APPENDIX J

SYSTEM AMAR BATCH STREAM - AMAR.CTL

Reference Figure J-1 for a listing of an unedited AMAR.CTL stream.

ST000: Attempts to have the pack mounted which contains the AMAR programs and database. The request is kept pending until the operator responds. If the pack cannot be mounted, control is transferred to ST180.

ST008: Checks to see if the operator had, at some point, set an incorrect date and time on the machine and if AMAR data files were created with a time stamp in the future. If the current machine date is correct, you should delete such files.

ST009: Checks to see if the operator had set an incorrect date and time on the machine and the AMAR database had been updated with the incorrect date/time. If the current machine date is correct, you should delete the database and restore from a good disk or tape backup. Control is transferred to ST180.

ST010: Deletes obsolete files from the primary production pack.

ST020: On some systems, the AMAR daily raw files are kept on a separate pack from the database (production) pack. This step copies the AMAR raw files to the production pack. If there is no separate pack, this step is bypassed and control is passed to ST040.

ST030: Deletes obsolete AMAR daily raw files from the data collection pack. It also checks to see if the data collection program is running. If not the operator is requested to restart the data collection program.

ST040: Checks to see if the AMAR database is nonexistent or corrupted. If it is, control is passed to ST050. If not, control is passed to ST080. Corruption can occur if the system crashes while either the AMARIP or AMARUP program is running.

ST050: Checks to see if a backup AMAR database exists on disk. If not, control is transferred to BD070. Otherwise, it deletes the file which controls tape backup frequency. It then deletes the corrupted database and copies the backup to the production pack. If the backup attempt fails, control is transferred to ST060. Otherwise control is transferred to ST070.

ST060: Attempts to have the backup pack mounted on a different drive and the AMAR database restored to the production pack. If ST060 fails, control is passed to BD070.

ST070: Checks to see if the restored database is corrupted. If it is, control is passed to BD070. Otherwise, control is passed to ST100.

BD070: Asks the operator to restore the AMAR database from tape. This step is reached only if no good copies of the database exist on disk. Transfers control to ST190 which stops the stream. The operator should restore AMAR.DB from tape before restarting.

ST080: If the original database is good, this step creates a backup of it and protects it to prevent it from being accidentally deleted. If there is no space available for the backup, control is passed to ST180. If parity errors have occurred, control goes to ST090. Otherwise control is passed to ST100.

ST090: In the event of parity errors during database backup, the operator is requested to mount the pack on a different drive and try the backup again. If that fails, it is assumed the database is bad. The operator is requested to restore a good copy from tape before resubmitting the stream. Control is passed to ST190 which stops the stream. Otherwise control is passed to ST100.

ST100: Inputs raw data files into the system AMAR database.

ST110: Updates the system AMAR database, creating the proper summary records and deleting obsolete data.

ST120: Generates the automatic AMAR reports.

ST130: Prints the AMAR reports. If the site has decided not to use tape backup in this stream, control passes directly to ST170. Otherwise control passes to ST140.

ST140: Checks the tape backup frequency control file to see if tape backup should occur. If not, control is passed to ST170.

ST150: Creates a tape backup of the system AMAR database. If the mount fails control is passed to ST180. If there is an error during backup, control is passed to BD150 which requests a clean tape and drive. The operator is requested to cancel the mount after three tries.

ST160: Creates a new tape frequency control file.

<u>ST170:</u> Deletes obsolete raw files from the production pack.

<u>ST180:</u> Resubmits the stream for processing after 1:00 AM the next night.

<u>ST190:</u> Stops the stream. Creates a record of any "bad" logs - AMAR.LG - for later examination.

```
;SUBSTITUTIONS SHOULD BE MADE AS FOLLOWS FOR ALL WORDS ENCLOSED IN ANGLE BRACKETS "<>"
;<DB-STRUC> IS THE NAME OF THE PRIMARY (PRODUCTION) DISK PACK
;WHICH CONTAINS THE AMAR DATABASE AND ASSOCIATED FILES.
;<AMAR-PPN> IS THE PPN WHICH CONTAINS THE PRIMARY AMAR DATABASE AND ASSOCIATED
;FILES.
;<DC-STRUC> IS THE NAME OF THE PACK WHICH CONTAINS THE DATA COLLECTION
;PROGRAMS, RAW FILES, AND STREAMS - IT MAY BE THE SAME OR DIFFERENT FROM
;<DB-STRUC>.
;<SYS-ID> IS THE 4 CHARACTER SYSTEM CODE.
;<DC-RTEN> IS THE RETENTION PERIOD IN DAYS FOR DAILY RAW FILES
;ON THE <DC-STRUC>.
;<DB-RTEN> IS THE RETENTION PERIOD IN DAYS FOR DAILY RAW FILES
;ON THE <DB-STRUC>.
;<BKUP-STRUC> IS THE NAME OF THE PACK USED TO BACK UP THE AMAR DATABASE
;AND ASSOCIATED FILES.
;***************************************************************************
;***************************************************************************
;AMAR.CTL CAN RESTART ONLY AT RESUBMIT STEP ST180

;/CHARGE:000000

ST000:: !******** MOUNT THE PRODUCTION PACK ********!

.SET WATCH NONE
.SET DSKFUL PAUSE
.MO <DB-STRUC>:/SHOVE
.R SETSRC
*C <DB-STRUC>
.IF (ERROR) .GOTO ST180

ST008:: !******** CHECK FOR INVALID FUTURE RAW FILES ********!

.ERROR
.RU <DB-STRUC>:DIP
*<DC-STRUC>:<SYS-ID>??.*/SINCE:NOW/DIRECT
.IF (NOERROR) .NOERROR
.RU <DB-STRUC>:DIP
*<DB-STRUC>:<SYS-ID>??.*/SINCE:NOW/DIRECT
.IF (ERROR) .GOTO ST009

BD008: !!!!!!!!! FOUND FUTURE RAW FILE(S) !!!!!!!!
!DELETE ABOVE RAW FILE(S) THAT WERE CREATED FOR FUTURE DATE(S)!

ST009:: !******** CHECK FOR INVALID FUTURE DATABASE ********!

.ERROR
.RU <DB-STRUC>:DIP
*<DB-STRUC>:AMAR.DB[<AMAR-PPN>]/SINCE:NOW/DIRECT
.IF (ERROR) .GOTO ST010

BD009:: !!!!!!!!! FOUND FUTURE DATABASE !!!!!!!!!
!IF TODAY'S DATE IS INCORRECTLY SET TO A PAST DATE DO NOTHING!
!IF TODAY'S DATE IS CORRECT AND LAST STREAM WAS RUN ON FUTURE DATE!
!   THEN RESTORE AMAR DATABASE AND RAW FILES FROM DISK OR TAPE BACKUP!
.GOTO ST180

ST010:: !******** DELETE OBSOLETE FILES FROM PRODUCTION PACK ********!

.R QUOLST
.NOERROR
.RU <DB-STRUC>:DIP
```


Figure J-1

```
*AMAR??.RPT/OKNONE.AMARUP.IF?/OKNONE/DELETE

ST020:: !******** COPY RAW FILES TO PRODUCTION PACK ********!

.ERROR %
.OPERATOR $
.RU <DB-STRUC>:DIP
*<DB-STRUC>:/PRESERVE:ALL=<DC-STRUC>:<SYS-ID>??.*/BEFORE:TODAY:0:10/OKNONE'NOT'<DC-STRUC>:<SYS-ID>DC.EXE
.NOOPERATOR
.IF (ERROR) .GOTO ST040

ST030:: !******** DELETE RAW FILES FROM DATA COLLECTION PACK ********!

.ERROR
.RU <DB-STRUC>:DIP
*<DC-STRUC>:<SYS-ID>??.*/BEFORE:-<DC-RTEN>D/OKNONE'NOT'<DC-STRUC>:<SYS-ID>D?.*/DELETE
.IF (NOERROR) .GOTO ST040

BD030:: !!!!!!!!! OPSER DATA COLLECTION PROGRAM MAY HAVE STOPPED !!!!!!!!!
!CHECK IF OPSER <SYS-ID>DC JOB IS RUNNING!

ST040:: !******** CHECK IF AMAR DATABASE IS CORRUPTED ********!

.R QUOLST
.ERROR %
.RU <DB-STRUC>:DIP
*<DB-STRUC>:AMAR.DB[<AMAR-PPN>]/DIRECT
.IF (ERROR) .GOTO ST050
.RU AMRGEN
*E D
.IF (ERROR) .GOTO ST050
.DEAS
.GOTO ST080

ST050:: !******** SUBSTITUTE BACKUP FOR CORRUPTED DATABASE ********!

.DEAS
.ERROR
.RU <DB-STRUC>:DIP
*<BKUP-STRUC>:AMAR.DBK[<AMAR-PPN>]/DIRECT
*<DB-STRUC>:AMAR.TAP[<AMAR-PPN>]/OKNONE/DELETE
.IF (ERROR) .GOTO BD070
.NOERROR
.RU <DB-STRUC>:DIP
*<DB-STRUC>:AMAR.DB[<AMAR-PPN>]/OKNONE/DELETE
.ERROR %
.OPERATOR $
.RU <DB-STRUC>:DIP
*<DB-STRUC>:AMAR.DB[<AMAR-PPN>]=<BKUP-STRUC>:AMAR.DBK[<AMAR-PPN>]
.NOOPERATOR
.IF (ERROR) .GOTO ST060
.RU <DB-STRUC>:DIP
*<DB-STRUC>:AMAR.DB[<AMAR-PPN>]/DIRECT
.IF (ERROR) .GOTO BD050
.GOTO ST070

BD050:: !!!!!!!!! INSUFFICIENT DISK SPACE TO RESTORE AMAR.DB DATABASE !!!!!!!!!
!DELETE UNNECESSARY FILES ON <DB-STRUC>!
.GOTO ST180

ST060:: !******** IF PROBLEM RESTORE BACKUP FROM DIFFERENT DRIVE ********!
```

Figure J-1 (continued)

```
.NOERROR
.DIS <BKUP-STRUC>:/REMOVE
.PLEASE MOUNT <BKUP-STRUC> ON A DIFFERENT DRIVE SINCE READ ERRORS ON CURRENT DRIVE^[
.MO <BKUP-STRUC>:
.RU <DB-STRUC>:DIP
*<DB-STRUC>:AMAR.DB[<AMAR-PPN>]/OKNONE/DELETE
.ERROR %
.OPERATOR $
.RU <DB-STRUC>:DIP
*<DB-STRUC>:AMAR.DB[<AMAR-PPN>]=<BKUP-STRUC>:AMAR.DBK[<AMAR-PPN>]
.NOOPERATOR
.IF (ERROR) .GOTO BD070

ST070:: !******** CHECK IF BACKUP DATABASE IS CORRUPTED ********!

.ERROR %
.RU <DB-STRUC>:DIP
*<DB-STRUC>:AMAR.DB[<AMAR-PPN>]/DIRECT
.IF (ERROR) .GOTO BD070
.RU AMRGEN
*E D
.IF (ERROR) .GOTO BD070
.DEAS
.GOTO ST100

BD070:: !!!!!!!!! ERROR RESTORING BACKUP AMAR.DB DATABASE !!!!!!!!!
!RESTORE AMAR.DB DATABASE FROM BACKUP TAPE THEN SUBMIT STREAM!
.GOTO ST190

ST080:: !******** CREATE A DISK BACKUP OF AMAR DATABASE ********!

.NOERROR
.RU <DB-STRUC>:DIP
*<BKUP-STRUC>:AMAR.DBK[<AMAR-PPN>]/RENAME/OKNONE/PROTECTION:057
*<BKUP-STRUC>:AMAR.DBK[<AMAR-PPN>]/DELETE/OKNONE
.ERROR %
.OPERATOR $
.RU <DB-STRUC>:DIP
*<BKUP-STRUC>:AMAR.DBK[<AMAR-PPN>]/PROTECTION:777=<DB-STRUC>:AMAR.DB[<AMAR-PPN>]
.NOOPERATOR
.IF (ERROR) .GOTO ST090
.RU <DB-STRUC>:DIP
*<BKUP-STRUC>:AMAR.DBK[<AMAR-PPN>]/DIRECT
.IF (ERROR) .GOTO BD080
.GOTO ST100

BD080:: !!!!!!!!! INSUFFICIENT DISK SPACE TO CREATE BACKUP AMAR.DB DATABASE !!!!!!!!!
!DELETE UNNECESSARY FILES ON <BKUP-STRUC>!
.GOTO ST180

ST090:: !******** IF PROBLEM CREATE BACKUP ON DIFFERENT DRIVE ********!

.NOERROR
.DIS <DB-STRUC>:/REMOVE
.PLEASE MOUNT <DB-STRUC> ON A DIFFERENT DRIVE SINCE READ ERRORS ON CURRENT DRIVE^[
.MO <DB-STRUC>:
.RU <DB-STRUC>:DIP
*<BKUP-STRUC>:AMAR.DBK[<AMAR-PPN>]/RENAME/OKNONE/PROTECTION:057
*<BKUP-STRUC>:AMAR.DBK[<AMAR-PPN>]/DELETE/OKNONE
.IF (ERROR) .GOTO ST180
```

Figure J-1 (continued)

```
.ERROR %
.OPERATOR $
.RU <DB-STRUC>:DIP
*<BKUP-STRUC>:AMAR.DBK[<AMAR-PPN>]/PROTECTION:777=<DB-STRUC>:AMAR.DB[<AMAR-PPN>]
.NOOPERATOR
.IF (ERROR) .GOTO BD090
.GOTO ST100

BD090:: !!!!!!!!! ERROR CREATING BACKUP AMAR.DB DATABASE !!!!!!!!!
!RESTORE AMAR.DB DATABASE FROM BACKUP TAPE IF I/O ERROR THEN SUBMIT STREAM!
.GOTO ST190

ST100:: !******** STORE RAW FILE DATA IN THE AMAR DATABASE ********!

.SET WATCH ALL
.SET WATCH NO FILES ;FOR 7.01
.MO <DB-STRUC>:
.ERROR
.OPERATOR $
.RU AMARIP
*ANYDAY
.NOOPERATOR
.IF (ERROR) .GOTO BD100
.GOTO ST110

BD100:: !!!!!!!!! ERROR DURING AMARIP PROGRAM !!!!!!!!!
!CORRECT PROBLEM THEN SUBMIT STREAM!
.GOTO ST190

ST110:: !******** ROLLUP DATA IN THE AMAR DATABASE ********!

.OPERATOR $
.RU AMARUP
.NOOPERATOR
.IF (ERROR) .GOTO BD110
.GOTO ST120

BD110:: !!!!!!!!! ERROR DURING AMARUP PROGRAM !!!!!!!!!
!CORRECT PROBLEM THEN SUBMIT STREAM!
.GOTO ST190

ST120:: !******** GENERATE AUTOMATIC AMAR REPORTS ********!

.ERROR
.OPERATOR $
.RU AMREPT
;DAILY UTILIZATION REPORT
*DU
*AMAR
*AUTO
*AMARDR.RPT
;WEEKLY UTILIZATION REPORT
*WU
*AMAR
*AUTO
*AMARWU.RPT
;MONTHLY UTILIZATION REPORT
*MU
*AMAR
*AUTO
*AMARMU.RPT
```

Figure J-1 (continued)

```
;WEEKLY TREND REPORT
*WA
*AMAR
*AUTO
*AMARWA.RPT
;MONTHLY TREND REPORT
*MA
*AMAR
*AUTO
*AMARMA.RPT
;WEEKLY COMPOSITE UTILIZATION REPORT
*WC
*AMAR
*AUTO
*AMARWC.RPT
;MONTHLY COMPOSITE UTILIZATION REPORT
*MC
*AMAR
*AUTO
*AMARMC.RPT
;DAILY DISK REPORT
*DD
*AMAR
*AUTO
*AMARDD.RPT
;WEEKLY DISK REPORT
*WD
*AMAR
*AUTO
*AMARWD.RPT
;MONTHLY DISK REPORT
*MD
*AMAR
*AUTO
*AMARMD.RPT
;DAILY TAPE REPORT
*DT
*AMAR
*AUTO
*AMARDT.RPT
;WEEKLY TAPE REPORT
*WT
*AMAR
*AUTO
*AMARWT.RPT
;MONTHLY TAPE REPORT
*MT
*AMAR
*AUTO
*AMARMT.RPT
*EXIT
.NOOPERATOR
.IF (ERROR) .GOTO BD120
.GOTO ST130

BD120:: !!!!!!!!! ERROR DURING AMREPT PROGRAM !!!!!!!!!
!ENSURE ADEQUATE DISK SPACE FOR REPORT FILES!

ST130:: !******** PRINT/XEROX AUTOMATIC AMAR REPORTS ********!

.SET WATCH NONE
```

Figure J-1 (continued)

```
.DEAS
.NOERROR
.PRINT AMAR??.RPT/DIS:RENAME
;.XEROX AMAR??.RPT/DIS:RENAME

.GOTO ST170
ST140:: !******** CHECK IF DAY TO CREATE BACKUP TAPE OF AMAR DATABASE ********!

.ERROR
.RU <DB-STRUC>:DIP
*<DB-STRUC>:AMAR.TAP[<AMAR-PPN>]/SINCE:NOW/OKNONE/DELETE
*<DB-STRUC>:AMAR.TAP[<AMAR-PPN>]/RENAME/PROTECTION:057
.IF (ERROR) .GOTO ST150
.RU <DB-STRUC>:DIP
*<DB-STRUC>:AMAR.TAP[<AMAR-PPN>]=<DB-STRUC>:AMAR.TAP[<AMAR-PPN>]/BEFORE:-<TAPE-PRD>D
.IF (ERROR) .GOTO ST170

ST150:: !******** CREATE A BACKUP TAPE OF AMAR DATABASE ********!

.ERROR [
.MO MTA:MTAA/WENABL/REE:SCRTCH/VID:"<TAPE-ID>"
.IF (ERROR) .GOTO BD151
.SET DENSITY MTAA 6250
.IF (ERROR) .GOTO BD151
.SET WATCH ALL
.SET WATCH NO FILES
.ERROR %
.OPERATOR $
.R SYS:BACKUP
*TAPE MTAA
*SSNAME AMAR
*SAVE  <DB-STRUC>:[<AMAR-PPN>]AMAR.DB,<SYS-ID>??.*
*PRINT TTY:
*EXIT
.NOOPERATOR
.IF (ERROR) .GOTO BD150
.SET WATCH NONE
.NOERROR
.UNLOAD MTAA
.DIS MTAA
.GOTO ST160

BD150:: !!!!!!!!! ERROR CREATING BACKUP TAPE !!!!!!!!!
!PROVIDE CLEAN ERROR FREE DRIVE AND NEW SCRATCH TAPES!
.SET WATCH NONE
.NOERROR
.UNLOAD MTAA
.DIS MTAA
.PLEASE SCRATCH TAPE(S) ALREADY CREATED AND LABELLED^[
.PLEASE CLEAN TAPE DRIVE AND MOUNT NEW SCRATCH SINCE BACKUP ERROR^[
.PLEASE IF THIS REMOUNT REQUEST REPEATS 3 OR MORE TIMES CANCEL ENSUING MOUNT^[
.PLEASE THEN THE STREAM WILL SKIP THE BACKUP TAPE STEP ASSUMING I/O ERROR^[
.BACKTO ST150

BD151:: !!!!!!!!! ERROR READING BACKUP INPUT FILES !!!!!!!!!
!IF BACKUP INPUT FILE HAS I/O ERROR RESTORE FILE FROM BACKUP DISK COPY!
!SKIPPING CREATION OF BACKUP TAPE!
.GOTO ST170

ST160:: !******** RECORD DATE OF BACKUP TAPE CREATION ********!
```

Figure J-1 (continued)

```
.RU <DB-STRUC>:DIP
*<DB-STRUC>:AMAR.TAP[<AMAR-PPN>]/CDATE:TODAY=<DB-STRUC>:<SYS-ID>DR.RFD[<AMAR-PPN>]

ST170:: !******** DELETE RAW FILES FROM PRODUCTION PACK ********!

.RU <DB-STRUC>:DIP
*<DB-STRUC>:AMAR.TAP[<AMAR-PPN>]/OKNONE/DELETE
*<DB-STRUC>:<SYS-ID>??.*[<AMAR-PPN>]/BEFORE:-<DB-RTEN>D/OKNONE'NOT'<DB-STRUC>:<SYS-ID>??.*[<AMAR-PPN>]/ASINCE:TODAY'NOT'<DB-STRUC>:<
SYS-ID>D?.*[<AMAR-PPN>]/DELETE


ST180:: !******** RESUBMIT THE AMAR BATCH STREAM ********!

.CHKPNT ST180
.DEAS
.NOERROR
.SET WATCH NONE
.DEL AMAR.LG
.SUB <DC-STRUC>:AMAR.CTL,<DC-STRUC>:AMAR.LOG/DISPOSE:DELETE/UNIQUE:1/RESTART:1/TIME:1:0:0/AFTER:1:0:0
.GOTO FIN


%TERR::
.GOTO BD180
%CERR::
.GOTO BD180
%ERR::
.GOTO BD180

BD180:: !!!!!!!!! TIME LIMIT EXHAUSTED OR MONITOR OR PROGRAM ERROR !!!!!!!!!
!!!!!!!!! STREAM IS RESUBMITTED IF THE DATABASE IS OK !!!!!!!!!
!!!!!!!!! CORRECT PROBLEM IF IT WILL RECUR !!!!!!!!!


.ERROR
.RU AMRGEN
*E D
.IF (NOERROR) .BACKTO ST180

ST190:: !******** STOP THE AMAR BATCH STREAM ********!

.DEAS
.SET WATCH NONE
.NOERROR
.COP <DB-STRUC>:AMAR.LG=<DC-STRUC>:AMAR.LOG
.DEL <DB-STRUC>:AMAR.TAP[<AMAR-PPN>]

FIN::
%FIN::
```

Figure J-1 (continued)

APPENDIX K

WORKLOAD AMAR BATCH STREAM - WCRPTB.CTL

Reference Figure K-1 for a listing of an unedited WCRPTB.CTL stream.

DOINC: This step runs WCINC which preprocesses raw "checkpoint" data to form incremental data. WCINC "steals" WC.RAW from the data collection program by renaming it yyqmwd.RA0, where yyqmwd is today's fiscal date. (The data collection program will automatically start a new WC.RAW the next time it takes samples). If possible, WCINC "primes the pump" from yyqmwd.RA1, where yyqmwd is yesterday's fiscal date. Then it reads today's .RA0 file and writes a pair of files, yyqmwd.IN0 and yyqmwd.IN1, where yyqmwd is yesterday's fiscal date. The file with extension .IN0 is a summary file with one record per five-minute sampling interval. The file with extension .IN1 is a detail file, with one record per active job per sampling interval. When all of yesterday's data has been processed, WCINC writes a new pump primer file which contains status information and the remainder of today's .RA0 file. WCINC deletes yesterday's .RA1 file and today's .RA0 file.

Note: This discussion has assumed that WCINC is run every day, as it should be. If WCINC is not run for a few days, appropriate files will still be created when it finally is run.

DOUPD: This step runs WCUPD which performs database management including input, update, and deletion. First WCUPD reads the appropriate pair of preprocessed files (.IN0 and .IN1) and creates a database daily file. The name of the created file is yesterday's fiscal date. The extension is .DB0 if yesterday was a normal workday. The extension is .DB1 if yesterday was a Saturday, Sunday, or holiday. Next the new database daily file is rolled up into the appropriate weekly database file. At the end of the fiscal week, the weekly files are rolled up into the appropriate monthly files. After a database daily file is created, its name is entered in a list of files ready for automatic reporting. After a weekly or monthly file has been completed (the last day or week has been included), its name is entered in the list of files ready for automatic reporting. When input and rollup are complete through yesterday, WCINC deletes the oldest preprocessed and database files if more than the user-specified number exist.

DORPT: This step produces the automatic reports. It is divided into six substeps, corresponding to the six types of file which may be ready for automatic reporting:

STEPNAME	REPORT PERIOD	TYPE OF DAY REPORTED ON	INPUT FILE SPEC.	REPORT FILENAME
DODY0	Day	Weekday	??????.DB0	WCDY0.ext
DOWK0	Week	Weekday	?????.DB0	WCWK0.ext
DODY1	Day	Weekend	??????.DB1	WCDY1.ext
DOWK1	Week	Weekend	?????.DB1	WCWK1.ext
DOMN0	Month	Weekday	????.DB0	WCMN0.ext
DOMN1	Month	Weekend	????.DB1	WCMN1.ext

Substeps are skipped if the input file for the type of day and report period has not yet been created. For example, even though the stream is run daily, monthly reports will be produced only once at the end of each fiscal month - not every day. Also question marks must be used in the input file specification. This notifies WCRPTB that automatic reporting is taking place.

DOPRT: This step prints the reports. All workload reports created on a given day are normally concatenated into a single file for efficient printing. Any monthly or weekly files are put at the beginning of this large file so that you won't miss them. The concatenated file is printed /DISP:RENAME to avoid accidental overwriting. The individual report files are left on disk until overwritten by other files of the same name and extension. This leaves the most recent copy of each report on disk.

DOSUB: This step is almost always executed to submit tomorrow's run, even if earlier steps failed. It also copies WCRPTB.LOG to WCRPTB.LG, because WCRPTB.LOG will be deleted after printing, and it may occasionally be handy for troubleshooting.

```
.DAYTIME
.SET WATCH ALL
.SET WATCH NO FILES
.IF (ERROR)
.R SETSRC
*C <DB-STRUC>,*
.IF (ERROR) .GOTO DOSUB
.DELETE WC.IN?,WC.RAX,WC.RP?
.IF (ERROR)
.R DIP
*<DB-STRUC>:WC.RAX=WC.RAW
*<DB-STRUC>:WCDBS.COX=WCDBS.CON
.IF (ERROR)
DOINC:: .CHKPNT DOINC
.R SETSRC
*C <DB-STRUC>,*
.IF (ERROR) .GOTO DOSUB
.RU WCINC
.IF (NOERROR) .GOTO DOUPD
.PLEASE STREAM WCRPTB.CTL FAILED IN STEP DOINC^[
.PLEASE NOTIFY AMAR ADMINISTRATION^[
.GOTO DOSUB
DOUPD:: .CHKPNT DOUPD
.R SETSRC
*C <DB-STRUC>,*
.IF (ERROR) .GOTO DOSUB
.RU  WCUPD
.IF (NOERROR) .GOTO DORPT
.PLEASE STREAM WCRPTB.CTL FAILED IN STEP DOUPD^[
.PLEASE NOTIFY AMAR ADMINISTRATOR^[
.GOTO DOSUB
DORPT:: .CHKPNT DORPT
.R SETSRC
*C <DB-STRUC>,*
.IF (ERROR) .GOTO DOSUB
;HERE TO DO ANY REPORTS ON WEEKDAY DAY FILES
DODYO::
.RUN WCRPTB
*HOURLY REPORT BY PROGRAM AND JOB
*??????.DBO
*WCDYO.PR1
*
*0
*24
*1
*
*.9
*N
*JOB
*PRGRM
*LOGIN
*
*CPU%
*
*
*N
;ONLY LIKELY ERROR IS NO READY WEEKDAY DAY FILE...
;...WHICH MEANS NO WEEKDAY WEEK FILE WILL BE READY
.IF (ERROR) .GOTO DODY1
.RUN WCRPTB
*SHIFT REPORT BY PROGRAM
```

Figure K-1

```
*??????.DBO
*WCDYO.PR2
*
*0
*24
*8
*
*.5
*N
*PRGRM
*
*CPU%
*
*
*N
.IF (ERROR)
.RUN WCRPTB
*SHIFT REPORT BY BATCH VS. TIMESHARING
*??????.DBO
*WCDYO.PR3
*
*0
*24
*8
*
*
*N
*BATCH
*
*CPU%
*
*
*N
.IF (ERROR)
.RUN WCRPTB
*SHIFT REPORT BY PPN
*??????.DBO
*WCDYO.PR4
*
*0
*24
*8
*
*
*N
*PPN
*
*CPU%
*
*
*N
.IF (ERROR)
;INSERT EXTRA REPORTS ON WEEKDAY DAY FILES AFTER THIS LINE
;INSERT EXTRA REPORTS ON WEEKDAY DAY FILES BEFORE THIS LINE
;HERE TO DO ANY REPORTS ON WEEKDAY WEEK FILES
DOWKO::
.RUN WCRPTB
*WEEKLY REPORT BY PROGRAM (TYPICAL 8-HOUR SHIFTS)
*?????.DBO
*WCWKO.PR2
*
```

Figure K-1 (continued)

```
*O
*24
*8
*
*.1
*N
*PRGRM
*
*CPU%
*
*
*N
;ONLY LIKELY ERROR IS NO READY WEEKDAY WEEK FILE...
;...WHICH MEANS WE CAN SKIP TO CHECK FOR WEEKEND DAY FILES
.IF (ERROR) .GOTO DODY1
.RUN WCRPTB
*WEEKLY REPORT BY PPN (TYPICAL 8-HOUR SHIFTS)
*?????.DBO
*WCWKO.PR4
*
*O
*24
*8
*
*
*N
*PPN
*
*CPU%
*
*
*N
.IF (ERROR)
;INSERT EXTRA REPORTS ON WEEKDAY WEEK FILES AFTER THIS LINE
;INSERT EXTRA REPORTS ON WEEKDAY WEEK FILES BEFORE THIS LINE
;HERE TO DO ANY REPORTS ON WEEKEND DAY FILES
DODY1::
.RUN WCRPTB
*HOURLY REPORT BY PROGRAM AND JOB
*??????.DB1
*WCDY1.PR1
*
*O
*24
*1
*
*.9
*N
*JOB
*PRGRM
*LOGIN
*
*CPU%
*
*
*N
;ONLY LIKELY ERROR IS NO READY WEEKEND DAY FILE, WHICH MEANS...
;...NO WEEKEND WEEK, WEEKEND MONTH OR WEEKDAY MONTH FILES WILL BE READY
.IF (ERROR) .GOTO DOPRT
.RUN WCRPTB
*SHIFT REPORT BY PROGRAM
```

Figure K-1 (continued)

```
*??????.DB1
*WCDY1.PR2
*
*0
*24
*8
*
*.5
*N
*PRGRM
*
*CPU%
*
*
*N
.IF (ERROR)
.RUN WCRPTB
*SHIFT REPORT BY BATCH VS. TIMESHARING
*??????.DB1
*WCDY1.PR3
*
*0
*24
*8
*
*
*N
*BATCH
*
*CPU%
*
*
*N
.IF (ERROR)
.RUN WCRPTB
*SHIFT REPORT BY PPN
*??????.DB1
*WCDY1.PR4
*
*0
*24
*8
*
*
*N
*PPN
*
*CPU%
*
*
*N
.IF (ERROR)
;INSERT EXTRA REPORTS ON WEEKEND DAY FILES AFTER THIS LINE
;INSERT EXTRA REPORTS ON WEEKEND DAY FILES BEFORE THIS LINE
;HERE TO DO ANY REPORTS ON WEEKEND WEEK FILES
DOWK1::
.RUN WCRPTB
*WEEKLY REPORT BY PROGRAM (TYPICAL 8-HOUR SHIFTS)
*?????.DB1
*WCWK1.PR2
*
```

Figure K-1 (continued)

```
*0
*24
*8
*
*.1
*N
*PRGRM
*
*CPU%
*
*
*N
;ONLY LIKELY ERROR IS NO READY WEEKEND WEEK FILE, WHICH MEANS...
;...NO WEEKEND MONTH OR WEEKDAY MONTH FILES WILL BE READY
.IF (ERROR) .GOTO DOPRT
.RUN WCRPTB
*WEEKLY REPORT BY PPN (TYPICAL 8-HOUR SHIFTS)
*?????.DB1
*WCWK1.PR4
*
*0
*24
*8
*
*
*N
*PPN
*
*CPU%
*
*
*N
.IF (ERROR)
;INSERT EXTRA REPORTS ON WEEKEND WEEK FILES AFTER THIS LINE
;INSERT EXTRA REPORTS ON WEEKEND WEEK FILES BEFORE THIS LINE
;HERE TO DO ANY REPORTS ON WEEKDAY MONTH FILES
DOMNO::
.RUN WCRPTB
*MONTHLY REPORT BY PROGRAM (TYPICAL 8-HOUR SHIFTS)
*????.DBO
*WCMNO.PR2
*
*0
*24
*8
*
*.1
*N
*PRGRM
*
*CPU%
*
*
*N
;ONLY LIKELY ERROR IS NO READY WEEKDAY MONTH FILE...
;...WHICH MEANS NO WEEKEND MONTH FILES WILL BE READY
.IF (ERROR) .GOTO DOPRT
.RUN WCRPTB
*MONTHLY REPORT BY PPN (TYPICAL 8-HOUR SHIFTS)
*????.DBO
*WCMNO.PR4
```

Figure K-1 (continued)

```
*
*O
*24
*8
*
*
*N
*PPN
*
*CPU%
*
*
*N
.IF (ERROR)
;INSERT EXTRA REPORTS ON WEEKDAY MONTH FILES AFTER THIS LINE
;INSERT EXTRA REPORTS ON WEEKDAY MONTH FILES BEFORE THIS LINE
;HERE TO DO ANY REPORTS ON WEEKEND MONTH FILES
DOMN1::
.RUN WCRPTB
*MONTHLY REPORT BY PROGRAM (TYPICAL 8-HOUR SHIFTS)
*????.DB1
*WCMN1.PR2
*
*O
*24
*8
*
*.1
*N
*PRGRM
*
*CPU%
*
*
*N
.IF (ERROR)
.RUN WCRPTB
*MONTHLY REPORT BY PPN (TYPICAL 8-HOUR SHIFTS)
*????.DB1
*WCMN1.PR4
*
*O
*24
*8
*
*
*N
*PPN
*
*CPU%
*
*
*N
.IF (ERROR)
;INSERT EXTRA REPORTS ON WEEKEND MONTH FILES AFTER THIS LINE
;INSERT EXTRA REPORTS ON WEEKEND MONTH FILES BEFORE THIS LINE
DOPRT:: .CHKPNT DOPRT
.R DIP
*<DB-STRUC>:WORKLD.RPT=WCMN?.PR?/OKNONE,WCWK?.PR?/OKNONE,WCDY?.PR?
*<DB-STRUC>:WC???.RP?=WC???.PR?
*<DB-STRUC>:WC???.PR?/DELETE
```

Figure K-1 (continued)

```
.IF (ERROR)
.PRINT WORKLD.RPT/DISP:RENAME
.IF (ERROR)
DOSUB:: .CHKPNT DOSUB
%TERR::
%CERR::
%ERR::
%FIN::
.SUBMIT WCRPTB.CTL/TIME:0:15:0/RESTART:1/UNIQUE:0/AFTER:0:30:0,WCRPTB.LOG/DELETE
.IF (ERROR)
.COPY <DB-STRUC>:WCRPTB.LG=WCRPTB.LOG
```

Figure K-1 (continued)

Blank Page

APPENDIX L

INSTALLATION AND RESOURCE REQUIREMENTS

INSTALLATION:

Prior to installation, the data center must resolve the following issues:

A. A four character code must be selected to represent the system running AMAR. This code is embedded in the System AMAR database and jobstream. It uniquely identifies the system to which the reports belong.

B. An account must be set up on the system which will be running AMAR. The project-programmer numbers can be anything the site wishes. Privileges on this account must be as follows:

SPY/PEK on all of core
SPY/PEK on monitor

With these privileges the word in ACCT.SYS becomes 000003000000. Also a 200 page jobsize limit is needed.

C. Next, temporary and permanent disk storage must be set up. The amount of disk space necessary is determined by the sizes of the System AMAR and Workload AMAR databases. The size of each database is a function of the amount of retained data at each summary level (i.e., monthly, weekly, daily, hourly) inside the database. Attached is a table that can be used to estimate the amount of storage necessary.

D. AMAR has two data collection programs--xxxxDC (where xxxx is the four character code for the system) and WHOWC. These programs are run as subjobs and collect data continuously. Therefore, once you have been notified that installation is complete, entries for these programs should be inserted in the OPR.ATO file as follows:

```
:SLOG 1,2
:DEF AMAR=
AMAR-RUN structure:xxxxDC[ppn]

:SLOG ppn
:DEF WC=
WC-RUN structure:WHOWC[ppn]
```

The site may decide on which pack this data is to be written. The AMAR subjob will allow hard-coding of the pack name. The WC subjob will write to the first pack in the search list. It should be kept in mind that these jobs run continuously and therefore the pack chosen should be permanently mounted.

E. For reporting purposes, each site must determine what portion of each day will be considered prime time. The prime time of each site is recorded into the database at the time it is created. The prime time interval must fall on whole hour boundaries. The default is 0800-1700. Up to 4 daily prime time periods may be selected.

F. The report generation function of System AMAR is very flexible, five different types of reports are offered at various summary levels. The different types are: Utilization, Trend Analysis, Typical Day, Disk, and Tape. The summary levels are: daily, weekly, and monthly. Each site can easily select and deselect the reports automatically generated by the batch stream, as well as generate any reports interactively in an ad-hoc fashion. The default reports for a new site are: daily utilization; weekly utilization, disk, and tape; monthly utilization, trend analysis, disk, and tape.

G. There are two batch streams that run nightly. These are AMAR.CTL and WCRPTB.CTL. The streams process the data collected during the day and generate reports. Although the streams are self-submitting, the logs should be checked daily. <u>It is important that both streams run every night.</u> If the streams are not run for an extended period of time, disk overflow and lengthy updating procedures will occur! Therefore, to restart the streams if they disappear from the queue, use the following commands:

```
.SUB AMAR.CTL,AMAR.LOG/DISPOSE:DELETE/UNIQUE:1
     /RESTART:1/TIME:1:0:0/AFTER:1:0:0

.SUB WCRPTB.CTL,WCRPTB.LOG/DISPOSE:DELETE/UNIQUE:0
     /RESTART:1/TIME:0:15:0/AFTER:0:31:0
```

Note: If a stream failed because of errors, simply resubmitting the stream may not work. The error should be looked up in <u>AMAR-10 Error Messages</u> in order to determine the appropriate course of action.

H. AMAR allows the site to set the holidays within the databases. Only holidays inserted into the database before they occur will be treated as holidays. Holidays must be respecified each fiscal year.

RESOURCE REQUIREMENTS:

AMAR is very efficient. Approximately .5% CPU time will be used on a KL10 for data collection, database maintenance, and reporting for both System and Workload AMAR combined. This may vary depending on the number of reports you select, the number of items you choose to measure, and the level of activity on your system.

TOTAL DISK STORAGE ESTIMATES IN BLOCKS

	System AMAR	Workload AMAR
Program Software	2500	500
Daily Raw File	100	2000
Previous Raw Files (7 days retention)	600	2000
TOTAL (without databases)	3200	4500

Database*

Size	Retention	Blocks	Retention**	Blocks
mini	M-12 W-13 D-35 H-7	3000	M-2 W-2 D-7	3500
midi	M-12 W-13 D-35 H-7 COW-1 COM-1	4000	M-3 W-5 D-14	6500
maxi	M-12 W-13 D-35 H-7 COW-5 COM-3	6000	M-12 W-13 D-35	20000

* A backup copy of the System AMAR database should also be kept on disk. Typically it is kept on a different pack from the primary database, although that is not a requirement.

Note that more than one permanent structure may be used for holding the software, databases, files, etc.

** The abbreviations for the retentions are as follows:

M - monthly
W - weekly
D - daily
H - hourly
COW - composite weekly
COM - composite monthly

Example:

For midi System AMAR and Workload databases the total space required is:

System AMAR programs and files	3,200	3200
Workload Programs and files	4,500	4500
System AMAR database	4,000	6000
Workload database	6,500	20000
Backup System AMAR database	4,000	6000
		34700
TOTAL	22,200	

It should be understood that these totals are only estimates. The actual disk space used is a function of the amount of data collected and retained. Typically, systems that are very busy will have slightly larger data files and databases than systems that aren't -- despite having the same retention periods.

APPENDIX M

SUMMARY OF PROGRAMS AND FILENAMES

The following programs and files will typically be found in the AMAR area. Most programs and files are used in daily processing. However, a few are used only in special cases such as installation and error recovery. These program names and filenames are considered reserved for AMAR use only. The user should take care to select other filenames when specifying report files, etc.

System AMAR:

AMAR.BWR	- AMAR Beware File
AMAR.DB	- System AMAR database
AMAR.DBK	- Backup copy of System AMAR database
AMAR.1ST	- Initialization file used only during AMAR installation
AMAR.CTL	- System AMAR jobstream
AMAR.TAP	- Control file used by AMAR.CTL for scheduling of tape backup
AMAREX.EXE	- Data Extraction Program
AMAREX.HLP	- Data Extraction Program help file
AMARIP.EXE	- Input Program
AMARON.EXE	- Online Inquiry Program
AMARSD.EXE	- Raw File Preprocessor Program
AMARUP.EXE	- Rollup Program
AMARcc.RPT*	- Default reports produced by AMAR.CTL
AMONLD.EXE	- Reserved for use in troubleshooting by the performance specialist
AMREPT.EXE	- Reporting Program
AMREPT.OVL	- Overlay for the Reporting Program - only if using the version of AMREPT.EXE that includes its own object time system
AMRGEN.EXE	- Database Generation Program
RFD.HLP	- Report File Description help file
TODAY.DB	- Mini-database output from AMARSD.EXE
xxxxDC.EXE	- Data Collection Program
xxxxDR.RFD	- Report File Description
xxxxdd.mmm*	- Raw data files created by xxxxDC.EXE

*See the notes starting at the end of this section.

Workload AMAR:

```
WC.EXE        - Special preprocessor Program for the
                current raw data file
WC.INO        - Summary preprocessed file output by WC.EXE
WC.IN1        - Detail preprocessed file output by WC.EXE
WC.RAW        - Raw data file created by WHOWC.EXE
WC.RAX        - Copy of yesterday's WC.RAW file
WCDBS.CON     - Control file for Workload database
WCDBS.COX     - Yesterday's WCDBS.CON (used for recovery)
WCDBS.LOK     - Access control file for database
WCDYn.RPz*    - Default daily reports produced by
                WCRPTB.CTL
WCFIX.EXE     - Program to set holidays.
WCINC.EXE     - Normal preprocessor Program for raw files
WCMNn.RPz*    - Default monthly reports produced by
                WCRPTB.CTL
WCRPT.EXE     - Reporting Program for preprocessed files
WCRPTB.EXE    - Reporting Program for database files
WCRPTB.CTL    - Workload jobstream
WCRPTC.EXE    - Reporting Program for large requests from
                database files
WCUPD.EXE     - Database Management Program
WCWKn.RPz*    - Default weekly reports produced by
                WCRPTB.CTL
WHOWC.EXE     - Data Collection Program
yyqm.DB0*     - Monthly workload database files for
                weekdays
yyqm.DB1*     - Monthly workload database files for
                weekends and holidays
yyqmw.DB0*    - Weekly workload database files for
                weekdays
yyqmw.DB1*    - Weekly workload database files for
                weekends and holidays
yyqmwd.DB0* - Daily workload database files for
                weekdays
yyqmwd.DB1* - Daily workload database files for
                weekends and holidays
yyqmwd.IN0* - Summary preprocessor file output by WCINC.EXE
yyqmwd.IN1* - Detail file output by WCINC.EXE
yyqmwd.RA0* - Raw data file renamed by WCINC.EXE
yyqmwd.RA1* - Portion of raw file with today's data
                (since midnight) left over after WCINC.EXE
                processes yesterday's data
```

*

d = day numbered 1 - 7 depending on its position within the fiscal week

m = month numbered 1 - 3 depending on its position within the fiscal quarter

n = 0, 1

q = quarter numbered 1 - 4 depending on its position with the fiscal year

w = week numbered 1 - 5 (and, on rare occasion, 6) depending on its position within the fiscal month

z = 0, 1, etc.

cc = report code

dd = calendar day

yy = fiscal year

mmm = alpha abreviation for month

xxxx = system code

Blank Page

APPENDIX N

THE FISCAL CALENDAR

The fiscal calendar is used primarily to control rollup of data into weekly and monthly summary records, deletion of old data, and automatic reporting. It is based on a year divided into quarters of 3 months each. The first and second months within each quarter contain 4 weeks each. The third month normally contains 5 weeks. Each fiscal week contains 7 days starting on a Sunday and ending on a Saturday. There are exactly 52 weeks in the normal fiscal year with no leftover days. Approximately once every 6 years, an extra week is added to the fiscal calendar to compensate for the days which have been "lost" by this process. This extra 6th week occurs in the third month of the fourth quarter.

Figure N-1 is an example of the FY82 Digital fiscal calendar. The Digital calendar normally starts at the end of June or beginning of July. This particular calendar begins on June 28, 1981. It contains the extra week mentioned above. This extra week occurs at the end of the fiscal year (June 1982).

Fiscal dates are used extensively by AMAR. They are always specified in the following format:

```
yyqmwd
 |||||- Fiscal day (1-7)
 ||||
 ||||-- Fiscal week (1-5)
 |||
 |||--- Fiscal month (1-3)
 ||
 ||---- Fiscal quarter (1-4)
 |
 |----- Fiscal year
```

For example, by looking at Figure N-1, it can be determined that:

September 1, 1981 = 821323

September occurs in quarter 1 of FY82; it is the 3rd month;

September 1 occurs in week 2; it is the 3rd day.

It is recommended that you keep a similar copy of your fiscal calendar on hand for quick help in such date translations.

1982 FISCAL CALENDAR 1982

PLEASE NOTE: FY82 IS A 53 WEEK FISCAL YEAR. THE EXTRA WEEK WILL BE TAKEN IN JUNE, MAKING Q4 A 14 WEEK QUARTER.

FIRST QUARTER

MONTH	WEEK NO.	S 1	M 2	T 3	W 4	T 5	F 6	S 7
1 JULY 4 WEEKS	1	28	29	30	1	2	3	4
	2	5	6	7	8	9	10	11
	3	12	13	14	15	16	17	18
	4	19	20	21	22	23	24	25
2 AUG 4 WEEKS	5	26	27	28	29	30	31	1
	6	2	3	4	5	6	7	8
	7	9	10	11	12	13	14	15
	8	16	17	18	19	20	21	22
3 SEPT 5 WEEKS 1	9	23	24	25	26	27	28	29
2	10	30	31	$\boxed{1}$	2	3	4	5
3	11	6	7	8	9	10	11	12
4	12	13	14	15	16	17	18	19
5	13	20	21	22	23	24	25	26

SECOND QUARTER

MONTH	WEEK NO.	S	M	T	W	T	F	S
OCT 4 WEEKS	14	27	28	29	30	1	2	3
	15	4	5	6	7	8	9	10
	16	11	12	13	14	15	16	17
	17	18	19	20	21	22	23	24
NOV 4 WEEKS	18	25	26	27	28	29	30	31
	19	1	2	3	4	5	6	7
	20	8	9	10	11	12	13	14
	21	15	16	17	18	19	20	21
DEC 5 WEEKS	22	22	23	24	25	26	27	28
	23	29	30	1	2	3	4	5
	24	6	7	8	9	10	11	12
	25	13	14	15	16	17	18	19
	26	20	21	22	23	24	25	26

THIRD QUARTER

MONTH	WEEK NO.	S	M	T	W	T	F	S
JAN 4 WEEKS	27	27	28	29	30	31	1	2
	28	3	4	5	6	7	8	9
	29	10	11	12	13	14	15	16
	30	17	18	19	20	21	22	23
FEB 4 WEEKS	31	24	25	26	27	28	29	30
	32	31	1	2	3	4	5	6
	33	7	8	9	10	11	12	13
	34	14	15	16	17	18	19	20
MARCH 5 WEEKS	35	21	22	23	24	25	26	27
	36	28	1	2	3	4	5	6
	37	7	8	9	10	11	12	13
	38	14	15	16	17	18	19	20
	39	21	22	23	24	25	26	27

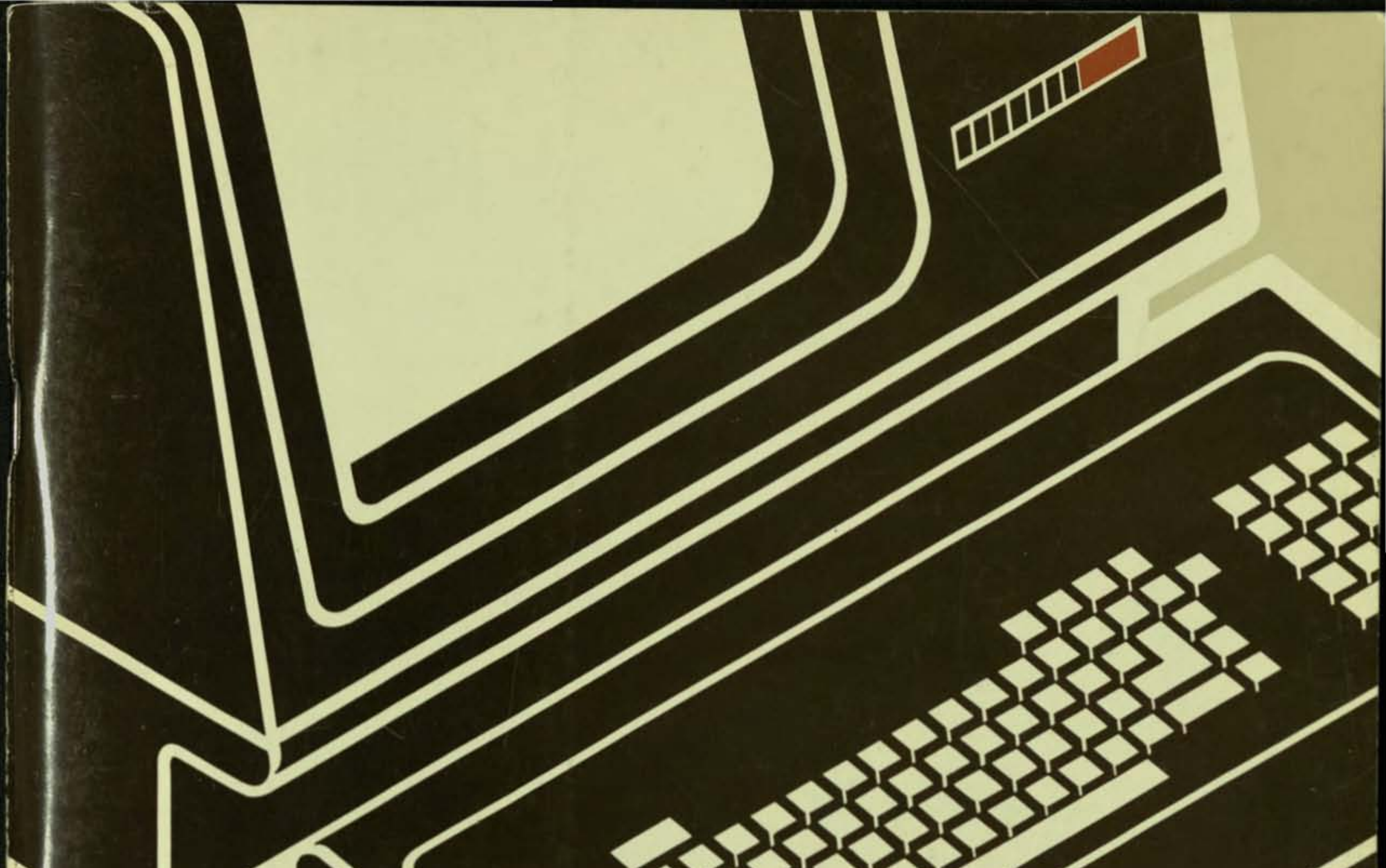
FOURTH QUARTER

MONTH	WEEK NO.	S	M	T	W	T	F	S
APRIL 4 WEEKS	40	28	29	30	31	1	2	3
	41	4	5	6	7	8	9	10
	42	11	12	13	14	15	16	17
	43	18	19	20	21	22	23	24
MAY 4 WEEKS	44	25	26	27	28	29	30	1
	45	2	3	4	5	6	7	8
	46	9	10	11	12	13	14	15
	47	16	17	18	19	20	21	22
JUNE 6 WEEKS	48	23	24	25	26	27	28	29
	49	30	31	1	2	3	4	5
	50	6	7	8	9	10	11	12
	51	13	14	15	16	17	18	19
	52	20	21	22	23	24	25	26
	53	27	28	29	30	1	2	3

Please Note:
All listed holidays are standard throughout the U.S. Assignable holidays and local location holidays are not shown.

digital

Figure N-1



VT105

GRAPHIC TERMINAL USER'S GUIDE

digital

EK-VT105-UG-001

VT105 GRAPHIC TERMINAL USER'S GUIDE

digital equipment corporation • marlboro, massachusetts

Preliminary, December 1978
First Edition, August 1979

Printed in U.S.A.

This document was set on DIGITAL'S DECset-8000 computerized typesetting system.

CONTENTS

FIGURES

TABLES

CHAPTER 1

OPERATOR INFORMATION

1.1 INTRODUCTION

The VT105 Graphic Terminal is a video terminal that features alphanumeric and graphical display. When interfaced to a host computer, the VT105 (shown in Figure 1-1) can display as many as two graphs, two shaded graphs, or two strip charts, along with alphanumeric labels. Control of the terminal is via a detachable keyboard that operates much like a typewriter. This chapter defines the operation of the terminal. Chapter 1 is divided into five parts:

1. Controls and indicators
2. Set-up mode
3. Definition of each Set-up feature
4. Self-testing the VT105
5. What to do in the event of a problem.

Part 1 shows all the controls and indicators on the terminal and summarizes the function of each to provide a quick reference for these functions.

Part 2 defines the Set-up mode and briefly summarizes the terminal's features.

Part 3 describes each feature in detail. Refer to this section for further information on a feature mentioned in Part 2.

Part 4 provides information on self-testing the VT105. It outlines the steps required to start the built-in self-tests and how to interpret the results of the tests.

Part 5 provides a table of easily recognized failures with simple corrective actions. *Check this list before calling for service.*

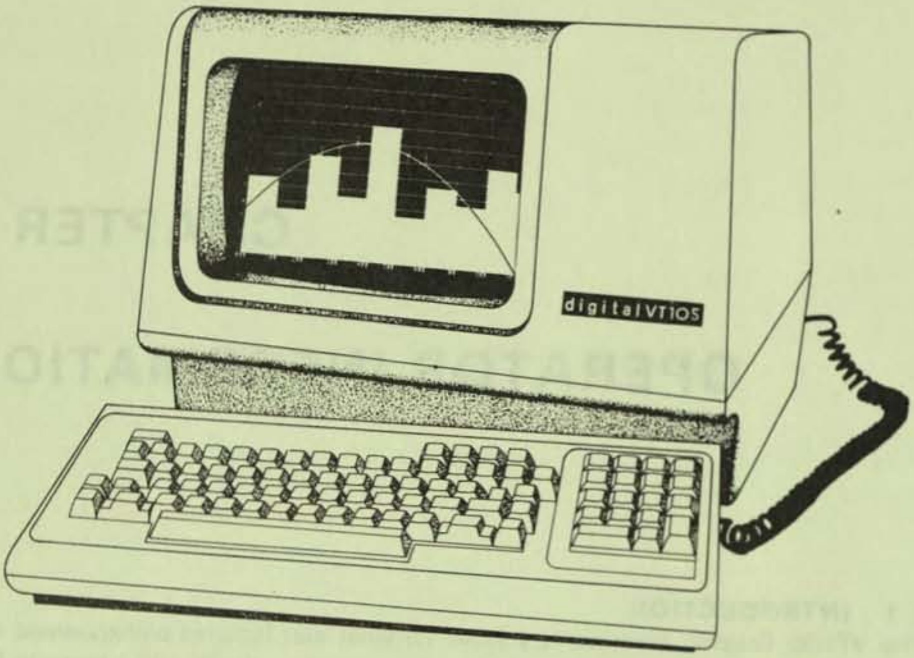


Figure 1-1 VT105 Graphic Terminal

1.2 CONTROLS AND INDICATORS

The controls and indicators of the VT105 are categorized as follows.

Monitor Controls
Keyboard Controls
Keyboard Indicators
Audible Indicators

1.2.1 Monitor Controls

The VT105 is equipped with two monitor controls (shown in Figure 1-2) – a POWER ON/OFF switch and a POWER SELECTOR switch. The POWER ON/OFF switch applies ac power to the terminal as follows.

Switch Position	AC Power
Up	On
Down	Off

The POWER SELECTOR switch is used to configure the terminal to the available ac input voltage as follows.

Switch Position	Voltage Range
115	90 – 128 Vac rms
230	180 – 256 Vac rms

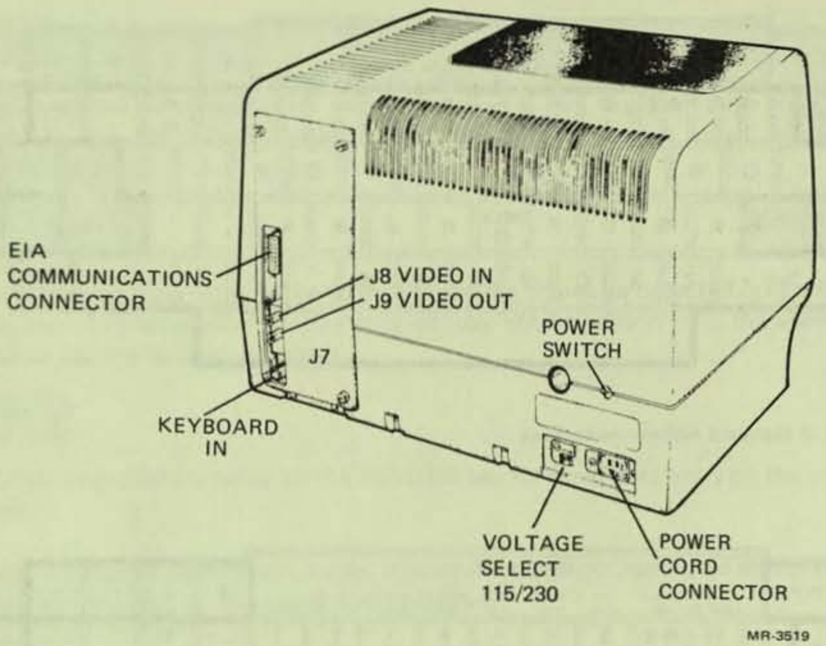


Figure 1-2 VT105 Graphic Terminal (Rear View)

1.2.2 Keyboard Controls

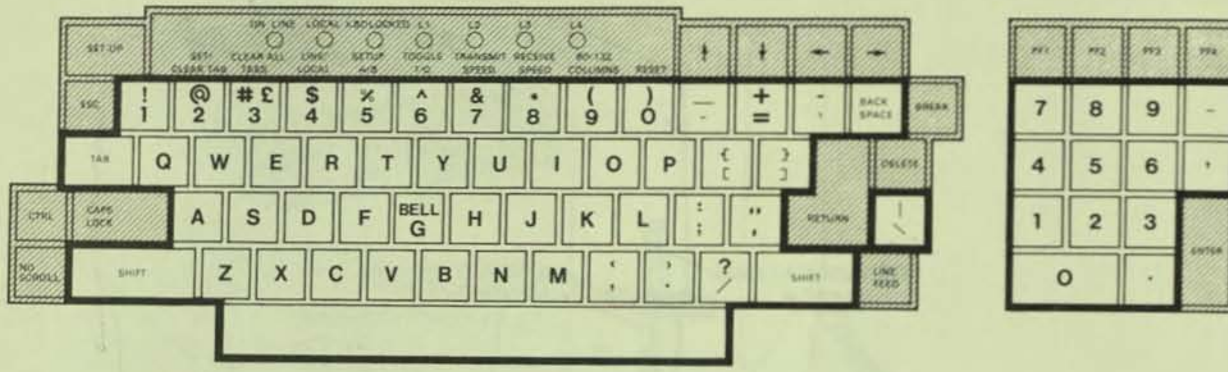
The VT105 has a detachable keyboard equipped with two keypads. Its main keypad array is arranged, and functions similar to a standard office typewriter. An auxiliary numeric keypad provides rapid data entry similar to an adding machine or calculator. Within this document, the keyboard keys are defined by functionality as follows.

Standard alphanumeric keys
Special function keys
Set-up mode keys

Standard Alphanumeric Keys – Figure 1-3 identifies the keys on the keyboard that function as standard typewriter or calculator keys when the terminal is off-line in LOCAL mode. When the terminal is on-line, operation of these keys is dependent on the system software of the host computer.

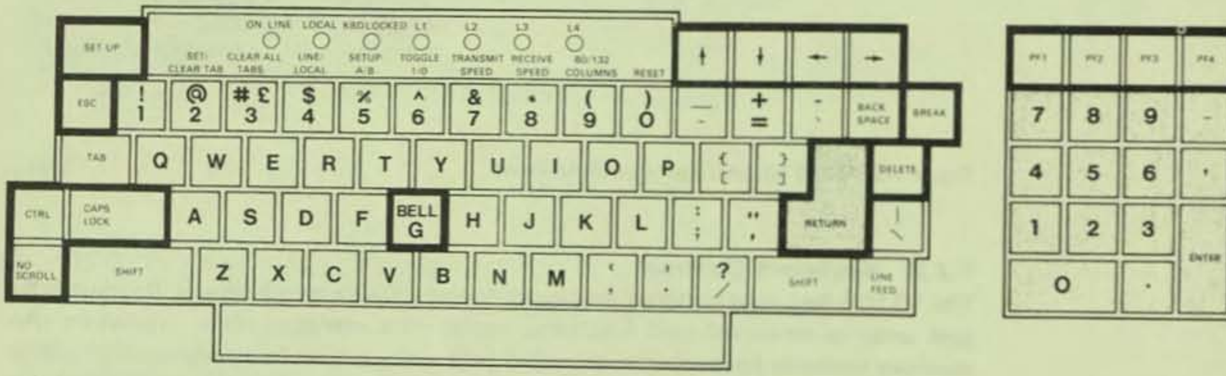
Special Function Keys – Figure 1-4 identifies the special function keys on the keyboard. Each of these keys provide a unique operation to the terminal. The function of these keys is dependent on the system software and may or may not have meaning to your system.

When in Keypad Application mode, the shaded keys, shown in Figure 1-4, provide additional special functions that may be used by an application program.



MR-3520

Figure 1-3 Standard Alphanumeric Keys



MR-3521

Figure 1-4 Special Function Keys

The general operation of the special function keys is provided in the following paragraphs.

BELL G

When pressed in combination with the CTRL key, this key causes a bell code (007_8) to be sent to the host computer.

BREAK

This key causes the console to generate a timed break signal.

CAPS LOCK

This key enables the transmission of uppercase alphabetic characters only. All numeric and special symbol keys remain in lowercase.

CTRL

When pressed in combination with another key, this key alters the code transmitted by the other keys on the main keyboard. These new codes may have a special meaning to the system.

DELETE

This key causes the terminal to transmit a delete character code (177_8) to the host computer. The deleted character may or may not be erased from the screen depending on the system software.

ENTER

This key operates the same as the RETURN key for rapid data entry on the numeric keypad.

When in Keypad Application mode, a control function is generated that may have a special meaning to the application program.

ESC

This key transmits an escape code (033_8) that normally has a special meaning to your system. In many applications, it treats the next keys pressed as a command.

LINEFEED

This key transmits a linefeed code (012_8).

NO SCROLL

When first pressed, this key stops transmission of data from the computer to the terminal. When pressed again, transmission resumes. Recognition of this key is dependent on the system software. This key causes the console to generate XOFF (DC3, 023_8) and XON (DC1, 021_8).

RETURN

This key transmits either a carriage return code (CR, 015_8) or a carriage return (CR) and linefeed code (LF, 012_8). This is a SET-UP selectable feature. (See Paragraph 1.3.2.)

PF1 **PF2** **PF3** **PF4**

These keys are used to generate special function codes that may be used by the system software.

↑ **↓** **→** **←**

Each of these keys causes the VT105 to transmit a code that may have a special meaning to your system. In Set-up mode the ↑ and ↓ keys increase or decrease the brightness of the display. The ← and → keys move the cursor left and right.

Set-Up Mode Keys – Figure 1-5 identifies the keyboard keys that establish the configuration of the terminal in Set-up mode. The dotted keys are used to position the cursor while in this mode. The paragraph that follows briefly describes the function of each key. Refer to Paragraph 1.3 for more details of the Set-up procedure.

SET-UP

This key is used to enter and exit the Set-up mode. SET-UP A is displayed on the screen when this key is pressed.

↑ ↓

Set-up A
These keys increase or decrease the brightness of the video characters. ↑ = increase; ↓ = decrease.

@
2

SET/CLEAR – Set-up A
This key sets or clears individual horizontal tabs. Position the cursor over the tab stop position and press this key.

£
3

CLEAR ALL TABS – Set-up A
This key clears all horizontal tabs.

$
4

LINE/LOCAL – Set-up A or Set-up B
This key switches the terminal to communicate with your system (ON LINE) or stops the terminal from communicating with your system (LOCAL). Two indicators above this key switch from ON LINE to LOCAL or from LOCAL to ON LINE.

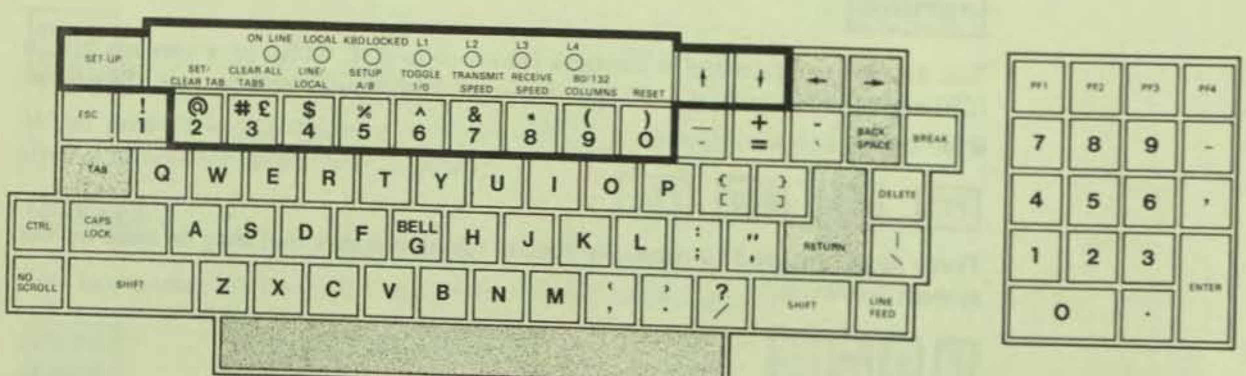


Figure 1-5 Set-Up Mode Keys

%
5

SET-UP A/B – Set-up A or Set-up B
This key switches the terminal from Set-up A to Set-up B or from Set-up B to Set-up A. The display indicates which mode the terminal is in.

^
6

TOGGLE – Set-up B
This key turns the selected operational feature on or off. (Refer to Paragraph 1.3.4 for more details.)

&
7

TRANSMIT SPEED – Set-up B
This key steps the terminal through the transmit baud rate settings in ascending order. The display indicates the current transmit baud rate (T SPEED) setting.

*
8

RECEIVE SPEED – Set-up B
This key steps the terminal through the receive baud rate settings in ascending order. The display indicates the current receive baud rate (R SPEED) setting.

(
9

80/132 COLUMNS – Set-up A
This key switches the display line size from 80 to 132 characters per line or from 132 to 80 characters per line. The number of characters displayed in the rule at the bottom of the screen in Set-up A mode is 80 or 132.

)
0

RESET – Set-up A or Set-up B
This key starts the reset sequence. This has the same result as turning the terminal power off and then on. Pressing this key exits set-up mode, performs a self-test, and places the cursor in the upper-left corner of the screen when the test is completed successfully. (If this does not happen, refer to Paragraph 1.6 to try to isolate the problem.)

1.2.3 **Keyboard Indicators** – Figure 1-6 shows the location of the keyboard indicators. The indicators are defined as follows.

ON LINE

This indicator lights to show that the terminal is on-line and ready to transmit or receive messages.

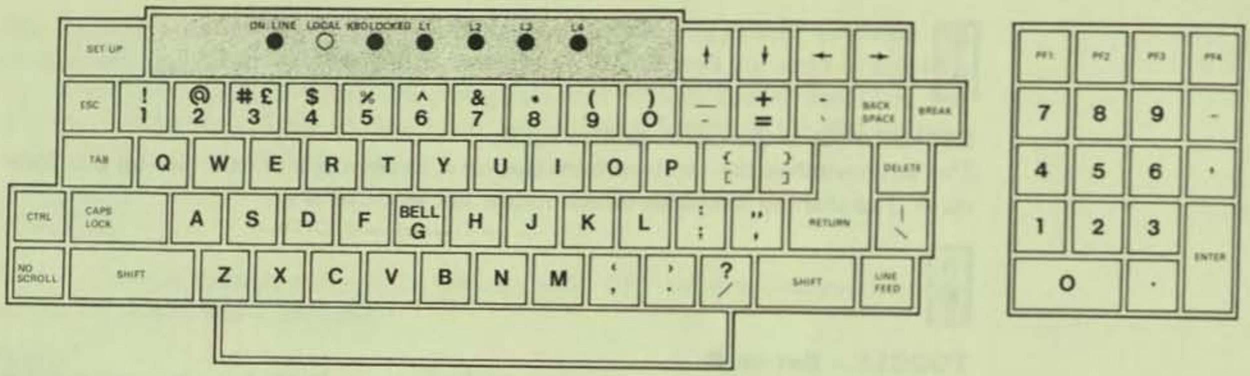


Figure 1-6 Keyboard Indicators

MR-3523

LOCAL

This indicator lights to show that the terminal is off-line and cannot communicate with the host device. In local mode, the keyboard remains active and all characters typed are placed on the screen.

> **NOTE**
> **Power on is indicated by lighting either the ON LINE or LOCAL indicator.**

KBD LOCKED

This indicator lights when the keyboard has been turned off. The terminal is still able to receive data from the host computer. This condition can be cleared by entering and exiting Set-up mode.

L1-L4

These indicators are turned on and off by the system software. L1-L4 are also used to show self-test errors.

1.2.4 Audible Indicators (Tones) – There are three audible alarms associated with the VT105: a short tone (click), a long tone (bell), and a series of long tones.

Short Tone (Click) – The short tone is sounded by the terminal whenever a key is pressed, with the following exceptions.

1. The SHIFT or CTRL keys do not generate any keyclick.
2. The KBD LOCKED indicator is turned on; in which case, the characters typed are lost.
3. The keyclick feature is turned off in Set-up B mode.

Long Tone (Bell) – The long tone is sounded by the terminal to indicate one of the following conditions.

1. A bell code is received by the terminal.

2. The margin bell feature is enabled, and the cursor is eight characters away from the right margin.

Series of Long Tones – The terminal sounds the long tone several times in rapid succession to indicate that the terminal's memory is having difficulty in reading or writing the Set-up features. (When this occurs, check the Set-up features and perform the Recall or Save operation again.)

1.3 SET-UP MODE

The VT105 has many selectable built-in features. These features provide compatibility with a host computer, adapt the terminal to local power, or alter the terminal's operation for user comfort and efficiency.

The VT105 does not use switches or jumpers to turn the features on or off. It uses a memory to electronically select and store the features. This is performed in the Set-up mode that is entered by pressing the SET-UP key. There are two Set-up displays:

1. Set-up A – displays the location of the tab stops along a visual ruler numbering each character position on the line.

2. Set-up B – summarizes the status of the other terminal features.

NOTE
Data on the video screen before entering Set-up mode is restored to the screen after exiting Set-up mode.

1.3.1 Determining What a Set-Up Feature Does

The Set-up features allow the terminal to be tailored to its operating environment. Table 1-1 lists each feature in one of the following general categories.

- Installation
- Computer compatibility
- Operator comfort

During initial installation, or when options are added or removed, or when the physical location of the terminal is changed, verify the settings of the features in the installation category.

Features that affect computer compatibility must be set correctly so that the terminal can communicate with the host computer. An error in these settings may cause incorrect data to be sent to or received from the computer; or an error may prevent the terminal from communicating with the computer. The settings for these features must be obtained from the host computer programmer, operator, or system manager since there are many combinations of settings designed to work with particular computers and special software. These feature settings would normally change only when you need to communicate with a different computer or a unique software package.

Table 1-1 Categories of Set-Up Features

Set-Up Feature	Installation	Computer Compatibility	Operator Comfort
ANSI/VT52 Mode		X	
ANSWERBACK Message		X	
AUTO REPEAT			X
AUTO XON XOFF		X	
BITS PER CHARACTER		X	
CHARACTERS PER LINE		X	
CURSOR			X
INTERLACE	X		
KEYCLICK			X
LINE/LOCAL		X	
MARGIN BELL			X
NEW LINE		X	
PARITY		X	
PARITY SENSE		X	
POWER	X		
RECEIVE SPEED		X	
SCREEN BACKGROUND			X
SCREEN BRIGHTNESS			X
SCROLL		X	X
TABS		X	
TRANSMIT SPEED		X	
WRAPAROUND		X	
# £ 3 (shifted)		X	

1.3.2 How to Change a Set-Up Feature

Changing any or all of the Set-up features is a simple operation and is generally performed by following the same basic steps.

1. Enter Set-up mode by pressing the SET-UP key.

2. Select the appropriate Set-up mode by pressing the 5 key on the main keyboard each time you want to switch from Set-up A to Set-up B or from Set-up B to Set-up A.

3. Position the cursor above the feature switch or tab stop to be changed. To position the cursor, the "SPACE" bar, "←", "→", "TAB" and "RETURN" keys may be used. Some features do not use this step since a specific key is dedicated to changing the feature.

4. Change the feature setting by pressing either the 6 key on the main keyboard or the appropriate dedicated key. Each time the key is pressed the feature will change, generally to the opposite state.

Table 1-2 briefly summarizes the Set-up features, the Set-up mode you must be in to change a given feature, and the key used to change the feature setting.

Table 1-2 Set-Up Feature Change Summary

Set-Up Feature	Changed In Set-Up A Mode	Changed In Set-Up B Mode	Key Used to Change Feature
ANSWERBACK message		X	†
ANSI/VT52 mode		X	^ 6
AUTO REPEAT		X	^ 6
AUTO XON/XOFF		X	^ 6
BITS PER CHARACTER		X	^ 6
BRIGHTNESS	X	X	↑ or ↓
CHARACTERS PER LINE	X		(9
CURSOR		X	^ 6
INTERLACE		X	^ 6
KEYCLICK		X	^ 6
LINE/LOCAL	X	X	$ 4
MARGIN BELL		X	^ 6
NEW LINE		X	^ 6
PARITY		X	^ 6
PARITY SENSE		X	^ 6
POWER		X	^ 6
RECEIVE SPEED		X	• 8
SCREEN		X	^ 6
SCROLL		X	^ 6

†A special sequence is required for this feature. See Paragraph 1.3.5.

Table 1-2 Set-Up Feature Change Summary (Cont)

Set-Up Feature	Changed In Set-Up A Mode	Set-Up B Mode	Key Used to Change Feature
TABS	X		@ 2 and # 3
TRANSMIT SPEED		X	$ 7
WRAPAROUND		X	^ 6
# £ 3 (shifted)		X	^ 6

1.3.3 Set-Up A

To enter Set-up A, press the SET-UP key. The display is similar to Figure 1-7. The bottom line of the display is a "ruler" that numbers each character position on a line. The location of each tab stop is shown by a "T" placed above the ruler. If the tab stops present are those desired, no action is necessary.

To exit Set-up A, press the SET-UP key.

Setting a Tab Stop – To set a tab stop, position the cursor above the desired location on the ruler. Use the SPACE BAR, →, and ← keys to move the cursor.

Press the 2 key on the main keyboard. A "T" is displayed in the desired tab stop location.

Erasing a Tab Stop – Position the cursor on the "T" displayed above the ruler. Press the 2 key. The "T" disappears.

Clear All Tab Stops – Press the 3 key. All "Ts" above the ruler disappear.

> **NOTE**
> **Tab stops set or cleared are stored temporarily by exiting the Set-up mode. To set tab stops on a fixed basis, a Save operation must be performed. Refer to Paragraph 1.3.6.**

ON LINE/LOCAL – While in Set-up A, pressing the 4 key toggles the terminal between ON LINE and LOCAL operation. Two indicators directly above this key reflect the current operation.

80/132 Characters per Line – While in Set-up A, the number of characters per line can be changed. Pressing the 9 key toggles the terminal between 80 and 132 characters per line.

The ruler on the screen reflects the current state of this feature. The line is not physically longer in 132 character mode; however, the characters are more compressed. For graph drawing in the VT105, 80 characters per line is required.

> **NOTE**
> **The screen contents are lost when toggling between 80 and 132 characters per line.**

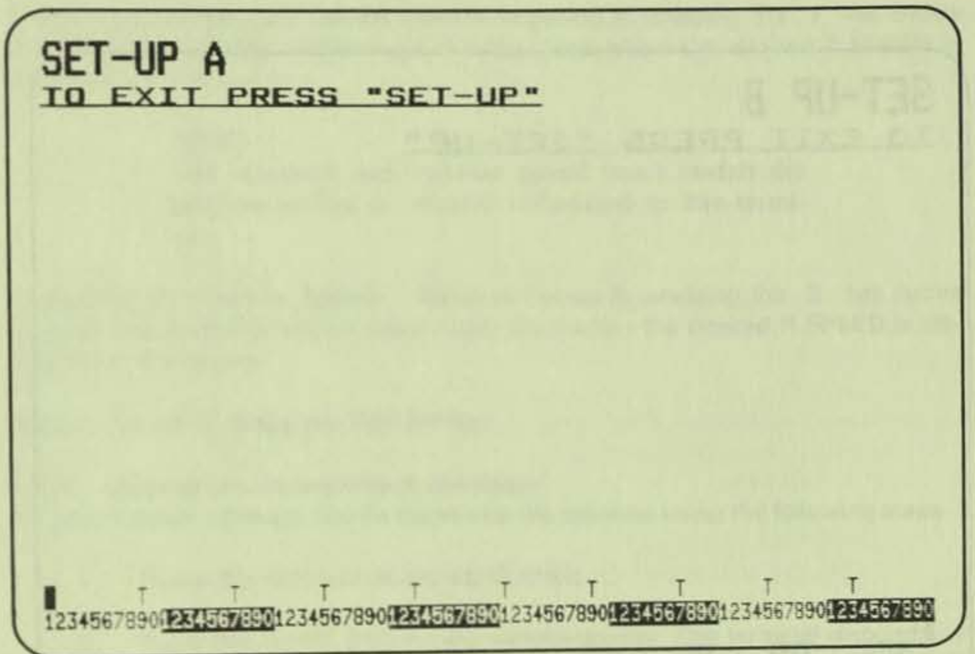


Figure 1-7 Set-Up A Mode Presentation

1.3.4 Set-Up B (Operational Features)

Set-up B mode is entered from Set-up A mode. When in Set-up A, press the 5 key on the main keyboard. The display is similar to Figure 1-8. Figure 1-9 summarizes the Set-up B presentation. This summary shows the operational features enabled. For additional information on a feature, refer to Paragraph 1.4.

Changing an Operational Feature – Position the cursor above the location of the feature to be changed using the SPACE BAR, ←, or →.

Press the 6 key on the main keyboard. This changes the displayed "0" to a "1" (or the "1" to a "0").

For graph drawing in the VT105, set the following features as indicated.

80 characters per line
Interlace disabled

The following additional features are recommended.

Auto XON/XOFF enabled
Smooth scroll
ANSI mode

For previous DECgraphic software (i.e., PLOT 55), the VT52 mode should be used.

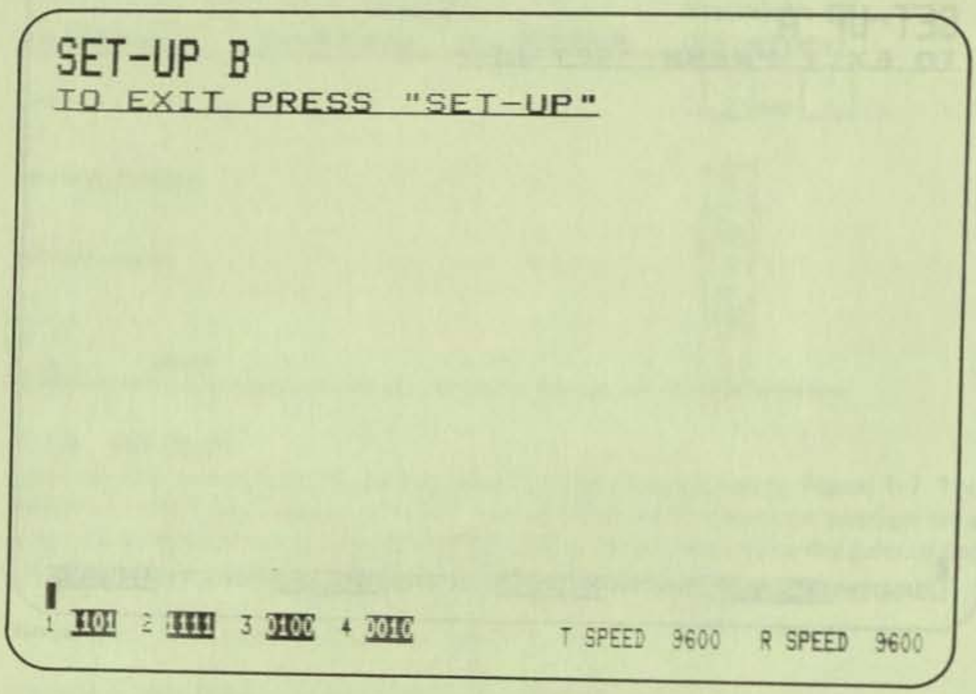


Figure 1-8 Set-Up B Mode Presentation

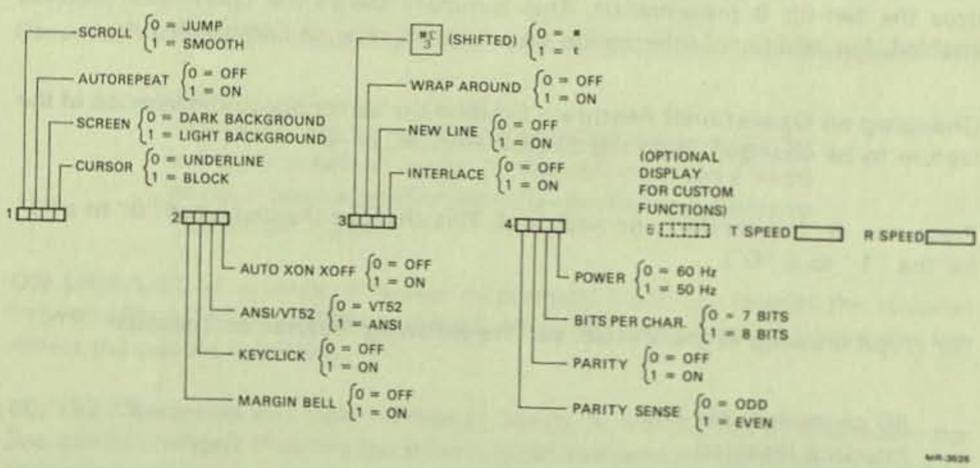


Figure 1-9 Set-Up B Mode Summary

Changing the Transmit Speed – While in Set-up B, pressing the 7 key cycles through the available transmit baud rates. Stop when the desired T SPEED is displayed on the screen.

> **NOTE**
> **The transmit and receive speed must match the host computer or modem connected to this terminal.**

Changing the Receive Speed – While in Set-up B, pressing the 8 key cycles through the available receive baud rates. Stop when the desired R SPEED is displayed on the screen.

To exit Set-up B, press the SET-UP key.

1.3.5 Setting the Answerback Message

An answerback message can be typed into the terminal using the following steps.

1. Place the terminal in Set-up B mode.

2. Press the SHIFT and A keys simultaneously. The terminal responds by placing "A =" on the screen. (The SHIFT key is required. The CAPS LOCK key does not work here.)

3. Type the message delimiter character. This may be any character not used in the actual answerback message. The message delimiter character is not a part of the answerback message.

4. Type the answerback message. The message may be up to 20 characters, including space and control characters. Control characters are displayed as a ♦ character to indicate their presence in the message.

   If a mistake is made when typing the answerback message, type the message delimiter character again and go back to step 2. This is the *only* way to correct errors in the answerback message.

5. Type the message delimiter character. Once the message delimiter character is typed, the answerback message disappears from the screen.

The answerback message is temporarily stored in the terminal and can be saved with the Save operation.

1.3.6 Saving the Set-Up Features

Set-up features may be changed and stored on either a temporary or a fixed basis. To temporarily store a feature, exit Set-up mode after changing the feature; the terminal now reacts according to the new setting. If a Recall operation is performed, or the terminal is reset, or the terminal power is turned off, all temporary feature settings are replaced by the features that have been stored on a fixed basis.

To store Set-up feature settings on a fixed basis, perform a Save operation as follows.

1. Place the terminal in either Set-up mode.

2. Press the SHIFT and S keys simultaneously. The screen clears and the message WAIT is displayed in the upper-left corner. After a brief wait, the terminal returns to Set-up A mode.

Set-up features stored temporarily are now stored on a fixed basis.

> **NOTE**
> **The Save operation must be performed at the terminal keyboard. The computer cannot perform this operation, although it can temporarily modify these settings.**

1.3.7 Recalling Set-Up Features

The temporarily stored Set-up feature settings may differ from the settings that are stored on a fixed basis. To return to the fixed settings, perform a Recall operation as follows.

1. Place the terminal in either Set-up mode.

> **NOTE**
> **When a Recall operation is performed, the contents of the screen are destroyed.**

2. Press the SHIFT and R keys simultaneously. The screen clears and the terminal returns to Set-up A mode.

1.3.8 Resetting the Terminal

The VT105 may be reset from the keyboard without turning power off. When the terminal is reset, the terminal memory is cleared and the self-test program is run as if the terminal power switch had been turned OFF and then back ON. Reset the terminal using the following steps.

1. Place the terminal in either Set-up mode.

2. Press the 0 (zero) key on the main keyboard. The terminal resets, the power on self-test is run, and the terminal responds to the fixed Set-up features. The cursor is returned to the upper-left corner of the screen.

> **CAUTION**
> **When a Reset operation is performed, the contents of the screen and graph memories are destroyed. Other options present may also be affected.**

1.4 DEFINITION OF EACH SET-UP FEATURE

The following paragraphs describe each Set-up feature in detail (in alphabetical order) and states how each feature affects the terminal.

> **NOTE**
> **Unless otherwise stated, entering Set-up mode and changing features does not result in the loss of data displayed on the screen.**

ANSI/VT52 Mode – (Set-up B)
The VT105 terminal follows one of two programming standards: American National Standards Institute (ANSI) or DIGITAL VT52. In ANSI mode, the VT105 generates and responds to control functions per ANSI standards X3.41-1974 and X3.64-1977. In VT52 mode, the terminal responds to control functions that are compatible with previous DIGITAL software using the VT52 or VT55 DECscope. Both modes are outlined in the programmer's chapter of this manual.

ANSWERBACK Message – (Set-up B)
Answerback is a question and answer sequence that allows the host computer to ask the terminal to identify itself. The terminal responds by sending a message to the host. The answerback sequence takes place automatically without affecting the screen or requiring operator action. The answerback message may also be transmitted by typing CTRL BREAK; this does not occur in LOCAL operation. (See Paragraph 1.3.5 for setting this feature.)

AUTO REPEAT – (Set-up B)
The auto repeat feature allows a key to be automatically repeated at the rate of about 30 characters per second when the key is held down for more than one-half second. The auto repeat feature affects all keyboard keys except the following.

SET-UP	TAB
ESC	RETURN
NO SCROLL	ENTER
	CTRL and any key

AUTO XON/XOFF – (Set-up B)
The VT105 supports the synchronizing codes XON (DC1) and XOFF (DC3). The XOFF control sequence is used to stop the transmission of data from the computer to the terminal; the XON sequence is used to resume transmission. With the feature enabled, the terminal generates the XOFF code when one of the following events occurs.

1. The internal buffer is nearly full.
2. The NO SCROLL key is pressed.
3. The terminal is placed in Set-up mode.
4. CTRL S is pressed.

The terminal resumes transmission when:

1. The internal buffer empties.
2. The NO SCROLL key is pressed again.
3. The terminal is taken out of Set-up mode.
4. CTRL Q is pressed.

If the host computer software does not support the XON/XOFF codes, data sent during buffer full conditions, or when the terminal is in Set-up mode, may be lost.

> **NOTE**
> **The VT105 always stops transmission when an XOFF (DC3) code is received and resumes transmission when an XON (DC1) code is received regardless of the setting of the AUTO XON/XOFF feature.**

BITS PER CHARACTER – (Set-up B)
This feature allows the terminal to transmit and receive either 7- or 8-bit characters. When set for 8-bit operation, bit 8 is set to a space (or 0) for characters transmitted and is ignored for all characters received.

CHARACTERS PER LINE – (Set-up A)
The VT105 is capable of displaying either 80 or 132 characters per line. In the 80 characters per line mode, the screen is 80 characters wide by 24 lines high.

In the 132 characters per line mode, the screen is 132 characters wide by 14 lines high (24 lines if the VT105 is equipped with the Advanced Video Option). In the 132 characters per line mode, the displayed lines are physically the same width as in the 80 characters per line mode, but the characters are more compact.

NOTES

1. **When changing from 80 to 132 characters per line mode or vice-versa, the current contents of the screen are lost.**

2. **The 80 characters per line mode must be used for graph drawing in the VT105.**

3. **The use of double-width characters reduces the number of characters per line by half.**

CURSOR – (Set-up B)
The VT105 offers a choice of two cursor displays to indicate the "active position" or where the next character will be placed on the screen. The cursor may be displayed as either a blinking underline (—) or a blinking block (■). The cursor selection may perform an additional function; see the SGR control function definition in Chapter 2.

INTERLACE – (Set-up B)
This feature is used for high resolution options. The interlace feature should be turned off if such an option is not installed to reduce screen flicker. For graph drawing in the VT105, this feature should be off (noninterlaced).

KEYCLICK TONE – (Set-up B)
The keyclick is a tone that is generated every time a key is pressed to provide audible feedback from the keyboard. The keyclick may be turned on or off. The keyclick volume is *not* adjustable.

LINE/LOCAL – (Set-up A or B)
The LINE/LOCAL feature places the terminal in either an ON-LINE or a LOCAL (off-line) condition. When the terminal is on-line, the keyboard ON-LINE indicator is ON. All characters typed on the keyboard are sent directly to the computer, and messages from the computer are displayed on the screen.

In the LOCAL condition, the keyboard LOCAL indicator is ON. The terminal is electrically disconnected from the computer; messages are not sent to or received from the computer; and characters typed on the keyboard are echoed on the screen directly.

MARGIN BELL – (Set-up B)
The margin bell sounds when the cursor is eight characters from the end of the current line to alert the operator while typing. This feature may be turned off. The bell tone volume is *not* adjustable.

NEW LINE – (Set-up B)
The new line feature enables the RETURN key on the terminal to function like the RETURN key on an electric typewriter. When the new line feature is enabled, pressing the RETURN key generates a carriage return (CR) and a line feed (LF). When a line feed code is received, the code is interpreted as a carriage return and line feed.

When the new line feature is disabled, the RETURN key generates only the CR code; an LF code causes the terminal to perform a line feed only.

NOTE
The application software may not recognize this new line feature.

PARITY – (Set-up B)
Parity, when enabled, checks for correct data transmission. If a transmission error occurs, the terminal indicates its presence by placing a checkerboard character (▒) on the screen in place of the character with the error. The parity sense feature determines if the parity is even or odd. When parity is disabled, no parity bit is transmitted or received.

PARITY SENSE – (Set-up B)
The parity sense defines which of two methods of parity checking, odd or even, is being used by the terminal. If parity is enabled, the terminal's parity sense must be matched to the parity of the computer. If the parity sense does not match, most characters sent to the computer are rejected. If a parity incompatibility occurs, the checkerboard character (▒) is shown on the screen in place of the received character.

NOTE
If parity is disabled, parity sense is disregarded.

POWER – (Set-up B)
During the initial installation, the terminal display must be set to the power line frequency 50 or 60 Hz. In the United States, this is set to 60 Hz.

RECALL – (Set-up A or B)
The fixed terminal memory may hold Set-up features that differ from those temporarily set in the terminal. To return to these fixed settings, use the Recall feature. Refer to Paragraph 1.3.7 to use this feature.

RECEIVE SPEED – (Set-up B)
The receive speed must be set to match the computer transmit speed. The VT105 is capable of receiving at any one of the following preselected speeds: 50, 75, 110, 134.5, 150, 200, 300, 600, 1200, 1800, 2000, 2400, 3600, 4800, 9600, 19,200 baud.

The receive speed is independent of the transmit speed; the terminal may receive data at one speed and transmit data at a different speed.

RESET – (Set-up A or B)
The reset feature initializes the system, clears the graph memories, and invokes the power-up sequence. The power-up sequence performs the self-tests and returns the terminal to its initialize state. Refer to Paragraph 1.3.8 to use this feature.

SAVE – (Set-up A or B)
The save feature is used to store Set-up features on a fixed basis. Refer to Paragraph 1.3.6 to use this feature.

SCREEN BACKGROUND – (Set-up B)
The screen background feature allows the operator to determine the background of the screen. In the normal screen mode, the display contains light characters on a dark background; in the reverse screen mode, the display contains dark characters on a light background.

SCREEN BRIGHTNESS – (Set-up A or B)
The VT105 electronically controls the brightness of characters displayed on the screen when using a dark background; or it controls the background screen brightness when the light background is selected. This feature allows the operator to select the desired level of brightness for maximum comfort under varied lighting conditions. This setting may be saved like any other feature in the terminal.

SCROLL – (Set-up B)
Scrolling is the upward or downward movement of existing lines on the screen to make room for new lines. It can be performed in two ways: jump scroll or smooth scroll. In jump scroll mode, new lines appear on the screen as fast as the computer sends them to the terminal. At the higher baud rates, the data is very difficult to read due to the rapid movement of the lines.

In smooth scroll mode, a limit is placed on the speed at which new lines of data may be sent to the terminal. The movement of lines occurs at a smooth steady rate allowing the data to be read as it appears on the screen.

> **NOTE**
> **Smooth scroll mode allows a maximum of six lines of data per second to be added to the screen. The AUTO XON/XOFF feature must be enabled and supported by the host computer to ensure that data is not lost when smooth scroll mode is enabled.**

TABS – (Set-up A)
The VT105 can tab to preselected points on a line. These tab stops may be individually changed, or totally cleared and then set. Refer to Paragraph 1.3.3 to set or clear tab stops.

TRANSMIT SPEED – (Set-up B)
Transmit speed must be set to match the computer receive speed. The VT105 is capable of transmitting at any one of the following preselected transmit speeds: 50, 75, 110, 134.5, 150, 200, 300, 600, 1200, 1800, 2000, 2400, 3600, 4800, 9600, and 19,200 baud.

Transmit speed is independent of receive speed; the terminal may transmit data at one speed and receive data at a different speed.

WRAPAROUND - (Set-up B)
When this feature is enabled, characters entered beyond 80 (or 132) per line (depending upon the line size selected) are placed on the next line. If the wraparound feature is not enabled, these characters are overwritten in the last character position of the current line.

> **NOTE**
> **The use of double-width characters reduces the number of characters per line by half.**

 # £ 3 (shifted) – (Set-up B)
The VT105 contains character sets for the United States and the United Kingdom. The difference between the two character sets is one character, the # or £ symbol. When the standard U.S. character set is selected, the uppercase 3 key on the main keyboard displays the # character. The £ character is displayed when the U.K. character set is selected.

1.5 SELF-TESTING THE VT105

A self-test program is built into the VT105 to automatically, or on command, test the condition of the terminal. The self-test program checks the following items.

Advanced video memory (if option is installed)
Nonvolatile memory (NVR)
Internal memory
Keyboard

This test is performed automatically whenever the terminal is turned on. It also can be invoked by the operator.

1. Enter Set-up mode; press the SET-UP key.
2. Press the 0 (zero) key on the main keyboard to perform a reset operation.

If the test is successful, the program flashes several test patterns, prints WAIT in the upper-left corner, and after a few seconds returns the cursor to its home (upper-left) position. LEDs L1–L4 are off.

If the test fails, check the self-test error codes.

Self-Test Error Codes
There are two broad categories of errors: fatal and nonfatal.

Fatal errors cause the terminal to immediately stop all operations. No intelligible information is displayed on the screen; however, the screen may contain a random pattern of characters. In addition, an error code may be displayed on the programmable keyboard LEDs, L1–L4. No terminal function, including the lighting of LEDs, is guaranteed if a fatal error occurs.

Nonfatal errors do not halt the terminal processor. Instead, the terminal is forced to LOCAL mode and an error code character is displayed in the upper-left corner of the screen.

There are five types of nonfatal errors.

1. Advanced Video Option RAM data (AVO)
2. Nonvolatile RAM data checksum error (NVR)
3. Keyboard missing or malfunction (KBD)
4. Data loopback error (Data)
5. EIA modem control error (EIA)

NOTE
The loopback and EIA tests are not performed on power-up; they must be invoked separately with the proper control function. See the programmer's chapter for further information on these tests.

Table 1-3 shows the possible nonfatal error characters that may appear on the screen and the failure represented by each character. Refer to Paragraph 1.6 to possibly isolate the cause.

Interactive Graphic Test
An interactive graphic test is built into the waveform generator module to test the graph drawing features of the VT105. Test patterns are generated by "Key-In" instructions. Refer to Appendix A to perform this test.

Table 1-3 Nonfatal Displayed Error Codes

Displayed Character	Fault Detected AVO	NVR	KBD	Data	EIA
1	X				
2		X			
3	X	X			
4			X		
5	X		X		
6		X	X		
7	X	X	X		
8				X	
9	X			X	
:		X		X	
;	X	X		X	
<			X	X	
=	X		X	X	
>		X	X	X	
?	X	X	X	X	
@					X
A	X				X
B		X			X
C	X	X			X
D			X		X
E	X		X		X
F		X	X		X
G	X	X	X		X
H				X	X
I	X			X	X
J		X		X	X
K	X	X		X	X
L			X	X	X
M	X		X	X	X
N		X	X	X	X
O	X	X	X	X	X

1.6 WHAT TO DO IN THE EVENT OF A PROBLEM

If it appears that there is a problem in the terminal, initiate the power-up self-test program. This test will help to determine if the problem is within the terminal or in some other part of the computer system.

If a problem occurs while in graphic mode, refer to Appendix A to perform the Interactive Graphics Test. This test checks the waveform generator module within the terminal.

Table 1-4 describes the items an operator can check prior to making a service call.

Table 1-4 Problem Checklist

Symptom	Possible Cause and Corrective Action
Terminal does not turn ON when the power switch is on.	The power cord is not plugged into wall outlet. Plug in cord.
	Power is not coming from the wall outlet. Check outlet with a known working electrical device (such as a lamp). If no power, call your electrician.
	The ac power cord is not plugged into the terminal. Plug in cord.
	The ac line fuse is blown. Turn the terminal OFF and replace the fuse. (See Figure 1-2 for location.)
No keyboard response	Keyboard cable is not plugged into the monitor. Plug in keyboard cable.
	KBD LOCKED indicator ON; the computer has turned the keyboard off. The KBD LOCKED condition may be cleared by entering and exiting the Set-up mode. If this condition persists, check with the host computer programmers for a possible error.
	Perform the self-test operation and note any error indications. Refer to Paragraph 1.5.
Garbled or error characters (▒)	Incorrect Set-up feature selection. Check the Set-up features. Suggested features that may be in error are: ANSI/VT52 mode AUTO XON/XOFF Bits per character Parity Parity sense Receive speed Transmit speed
	Perform the self-test operation and note any error indications.
Last character is garbled	Wraparound feature may be disabled. Enable this feature.
Double line feeds occur	New line feature may be enabled with the computer already performing this function. Disable the new line feature (Set-up B).

Table 1-4 Problem Checklist (Cont)

Symptom	Possible Cause and Corrective Action
Losing data in graph drawing mode	Check the following features. AUTO XON/XOFF should be enabled INTERLACE should be disabled 80 characters per line should be enabled
	ANSI/VT52 mode may be erroneously selected for software being used.
Several successive long tones	The terminal is having difficulty reading or writing the Set-up features in the nonvolatile memory.
	Check the feature settings and perform the Save operation. Perform the Self-Test operation.

1.7 RELATED HARDWARE MANUALS

The following hardware manuals are related to the VT105 and may be purchased from Digital Equipment Corporation.

Title	Part Number	Availability
VT105 Graphics Terminal Technical Manual	EK-VT105-TM	Microfiche Library
VT100 Technical Manual	EK-VT100-TM	Microfiche
VT105 Illustrated Parts Breakdown VT100 Illustrated Parts Breakdown	EK-VT105-IP EK-VT100-IP	Hardcopy or microfiche
VT100 Print Set (base terminal) VT105 Print Set (supplement)	MP-00663 MP-00642	Hardcopy Hardcopy

All purchase orders for hardware manuals should be forwarded to:

Digital Equipment Corporation
Cotton Road
Accessory and Supplies Group (P086)
Nashua, N H 03060

Purchase orders must show shipping and billing addresses and state whether a partial shipment will be accepted.

All correspondence and invoicing inquiries should be directed to the above address.

For information concerning microfiche libraries, contact:

Digital Equipment Corporation
Micropublishing Group BU/D2
Bedford, MA. 01730

CHAPTER 2

PROGRAMMER INFORMATION

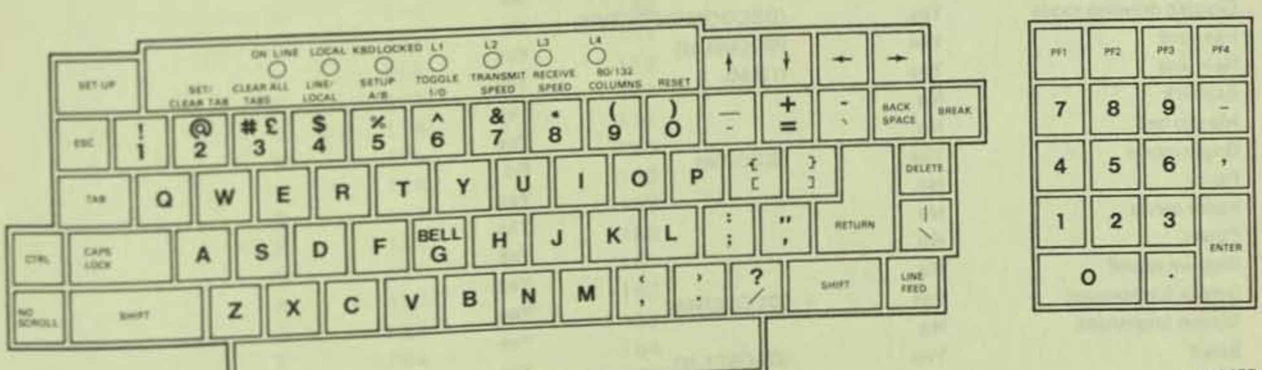
2.1 INTRODUCTION

The VT105 normally performs a two-part function. It is an input device to a computer; information entered through the keyboard is sent to the computer. It is simultaneously an output device for the computer; that is, data coming from the computer is displayed on the video screen.

This chapter discusses communication between the VT105 and the host. Included are the codes generated by the keyboard; the transmission protocol followed by the terminal; and the actions and reactions of the terminal to control codes and control functions in both ANSI and VT52 modes of operation.

2.2 THE KEYBOARD

The VT105 uses a keyboard with a key arrangement similar to an ordinary office typewriter, as shown in Figure 2-1. In addition to the standard typewriter keys, the keyboard has additional keys to generate control functions and cursor control commands. The keyboard also has indicators to show the current terminal status.



MR-2577

Figure 2-1 VT105 Keyboard

2.2.1 LED Indicators

The keyboard has seven light-emitting diodes (LEDs) of which two are committed to the complementary ON-LINE/LOCAL function. The power-on condition is shown by either of the two LEDs being on.

A third LED indicates a "keyboard locked" condition. When this LED is on, the keyboard is automatically "turned off" by the terminal due to a full buffer, or by the host through the transmission of an XOFF to the terminal.

The four remaining LEDs are programmable and can be assigned any meaning for specific applications. The code sequences to turn these LEDs on or off are discussed in Paragraph 2.4.1, Programmable LEDs.

2.2.2 SET-UP Key

The SET-UP key is at the upper-left corner of the main key array. Operations performed in Set-up mode can be stored in a nonvolatile memory (NVR) so that turning the terminal power off does not, by itself, alter the terminal configuration.

The procedures to change the Set-up features are provided in Chapter 1 of this manual. Those Set-up features which may be modified by the host are listed in Table 2-1 and described in detail in this chapter. (See Paragraph 2.4.)

Table 2-1 Set-Up Features and Terminal Modes

Set-Up Feature or Terminal Mode	Changeable from Host Computer	Control Function Mnemonic	Saved in Memory and Changeable in Set-Up
Alternate keypad mode	Yes	(DECKPAM/DECKPNM)	No
ANSI/VT52	Yes	(DECANM)	Yes
Auto repeat	Yes	(DECARM)	Yes
AUTO XON XOFF	No		Yes
Bits per character	No		Yes
Characters per line	Yes	(DECCOLM)	Yes
Cursor	No		Yes
Cursor key mode	Yes	(DECCKM)	No
Graphic drawing mode	Yes	(DECGON/DECGOFF)	No
Interlace	Yes	(DECINLM)	Yes
New line	Yes	(LNM)	Yes
Keyclick	No		Yes
Margin bell	No		Yes
Origin mode	Yes	(DECOM)	No
Parity	No		Yes
Parity sense	No		Yes
Power	No		Yes
Receive speed	No		Yes
Screen background	Yes	(DECSCNM)	Yes
Screen brightness	No		Yes
Scroll	Yes	(DECSCLM)	Yes
Tabs	Yes	(HTS/TBC)	Yes
Transmit speed	No		Yes
Wraparound	Yes	(DECAWM)	Yes
[# £ 3] (shifted)	Yes	(SCS)	Yes

2.2.3 Keyboard Operation

The keyboard transmits ASCII codes to the host. Some keys transmit one or more codes to the host immediately when typed. Other keys, such as CTRL and SHIFT, do not transmit codes when typed but modify the codes transmitted by other keys. If two code-transmitting keys are pressed together, two codes are transmitted in the order the keys are typed. The terminal does not wait for the keys to be lifted. If three keys are pressed simultaneously, the codes for the first two keys are transmitted immediately; the code for the third is transmitted when one of the first two keys is lifted.

2.2.4 Alphabetic Keys

The VT105 prints either upper or lowercase alphabetic characters. The codes required to print these are listed in Table 2-2. The VT105 transmits the lowercase code until either SHIFT key or the CAPS LOCK key is pressed. The CAPS LOCK key locks only the 26 alphabetic keys in the uppercase mode.

2.2.5 Nonalphabetic Keys

Each of the nonalphabetic keys generates two different codes. One code is generated without pressing the SHIFT key; the other is generated if either or both SHIFT keys are pressed. Table 2-3 shows the codes generated by these keys. The CAPS LOCK key does not affect these keys.

Table 2-2 Alphabetic Key Codes

Key	Uppercase Code (octal)	Lowercase Code (octal)
A	101	141
B	102	142
C	103	143
D	104	144
E	105	145
F	106	146
G	107	147
H	110	150
I	111	151
J	112	152
K	113	153
L	114	154
M	115	155
N	116	156
O	117	157
P	120	160
Q	121	161
R	122	162
S	123	163
T	124	164
U	125	165
V	126	166
W	127	167
X	130	170
Y	131	171
Z	132	172

Table 2-3 Nonalphabetic Key Codes

Lowercase Character	Neither SHIFT Key Down (Octal)	Uppercase Character	Either or Both SHIFT Keys Down (Octal)
1	061	!	041
2	062	@	100
3	063	# or £	043
4	064	$	044
5	065	%	045
6	066	^	136
7	067	&	046
8	070	*	052
9	071	(	050
0	060	)	051
-	055	_	137
=	075	+	053
[	133	{	173
;	073	:	072
'	047	"	042
,	054	<	074
.	056	>	076
/	057	?	077
\	134	|	174
`	140	~	176
]	135	}	175

2.2.6 Function Keys

There are several keys on the keyboard that transmit control codes. Control codes do not produce displayable characters but are codes for functions. If these codes are received by the terminal, it performs the function shown in Table 2-4.

2.2.7 NO SCROLL Key

The NO SCROLL key generates a single XOFF code to inhibit scrolling and freezes the screen. When pressed again, the same key generates XON. If the software recognizes XOFF, the host stops transmitting until the NO SCROLL key is pressed again. Disabling the XOFF/XON feature in Set-up B, disables the NO SCROLL key.

2.2.8 BREAK Key

Typing the BREAK key causes the transmission line to be forced to its zero or space state for 0.2333 seconds ±10 percent. If either SHIFT key is down, the time is increased to 3.5 seconds ±10 percent and Data Terminal Ready is disabled. After the 3.5 second interval, Data Terminal Ready is again asserted.

The SHIFT BREAK provides a long-break-disconnect function. Modems with EIA RS-232-C levels can be configured to use this long-break to cause both the local and remote data sets to disconnect. Modems that are connected via the 20 mA current loop may be configured to disconnect the remote data set only.

Table 2-4 Function Key Codes

Key	Octal Code	Terminal Action
RETURN*	015	Carriage return
LINE FEED	012	Line feed
BACK SPACE	010	Backspace
TAB	011	Tab to next tab stop
SPACE BAR	040	Deposit a space on the screen, erasing that position
ESC	033	Escape – interpret the following character string from the host as a command, rather than displaying it.
Delete	177	Ignored

*The RETURN key can be redefined to issue 015_8, 012_8 (carriage return and line feed). The new line feature in Set-up mode provides this capability.

The CTRL and BREAK keys typed together cause the transmission of the answer-back message.

> **NOTE**
> **The BREAK key does not function when the terminal is in LOCAL mode.**

2.2.9 CTRL (Control) Key

The CTRL key is used with other keys on the keyboard to generate control codes. If the CTRL key is held down, the code transmitted by the other keys is shown in Table 2-5.

2.2.10 Cursor Control Keys

The keyboard contains four keys labeled with arrows in each of four directions. These keys transmit control functions. If the host echoes these control functions back to the terminal, the cursor moves one character up, down, right, or left. Table 2-6 shows the escape sequence generated by each key.

> **NOTE**
> **In Set-up mode, the left and right arrow keys move the cursor; the up and down keys increase or decrease the screen brightness.**

2.2.11 Auto Repeating

All keys will auto repeat except: SET-UP, ESC, NO SCROLL, TAB, RETURN, and any key pressed with CTRL. Auto repeating may be disabled (SET-UP function). Auto repeating works as follows: when a key is typed, its code(s) is sent once, immediately. If the key is held down for more than 1/2 second, the code(s) are sent repeatedly at a rate of approximately 30 Hz until the key is released. (This rate is less if a low transmit baud rate is used.)

Table 2-5 Control Codes Generated

Key Pressed with CTRL Key Down (Shifted or Unshifted)	Octal Code Transmitted	Function Mnemonic
Space Bar	000	NUL
A	001	SOH
B	002	STX
C	003	ETX
D	004	EOT
E	005	ENQ
F	006	ACK
G	007	BELL
H	010	BS
I	011	HT
J	012	LF
K	013	VT
L	014	FF
M	015	CR
N	016	SO
O	017	SI
P	020	DLE
Q	021	DC1 or XON
R	022	DC2
S	023	DC3 or XOFF
T	024	DC4
U	025	NAK
V	026	SYN
W	027	ETB
X	030	CAN
Y	031	EM
Z	032	SUB
[	033	ESC
\	034	FS
]	035	GS
~	036	RS
?	037	US

Table 2-6 Cursor Control Key Codes

Cursor Key (arrow)	VT52 Mode	ANSI Mode and Cursor Key Mode Reset	ANSI Mode and Cursor Key Mode Set
Up	ESC A	ESC [A	ESC O A
Down	ESC B	ESC [B	ESC O B
Right	ESC C	ESC [C	ESC O C
Left	ESC D	ESC [D	ESC O D

2.2.12 Special Graphic Characters If the special graphic set is selected, the ASCII codes 137_8 through 176_8 are replaced with characters shown in Table 2-7. (See the SCS control function to enable this mode.)

Table 2-7 Special Graphic Characters

Octal Code	Graphic With US or UK Set	Graphic with "Special Graphics" Set	
137	–		Blank
140	`	♦	Diamond
141	a	▒	Checkerboard (error indicator)
142	b	HT	Horizontal tab
143	c	FF	Form feed
144	d	CR	Carriage return
145	e	LF	Line feed
146	f	°	Degree symbol
147	g	±	Plus/minus
150	h	NL	New line
151	i	VT	Vertical tab
152	j	┘	Lower-right corner
153	k	┐	Upper-right corner
154	l	┌	Upper-left corner
155	m	└	Lower-left corner
156	n	+	Crossing lines
157	o	–	Horizontal line - Scan 1
160	p	–	Horizontal line - Scan 3
161	q	–	Horizontal line - Scan 5
162	r	–	Horizontal line - Scan 7
163	s	–	Horizontal line - Scan 9
164	t	├	Left "T"
165	u	┤	Right "T"
166	v	┴	Bottom "T"
167	w	┬	Top "T"
170	x	|	Vertical bar
171	y	≤	Less than or equal to
172	z	≥	Greater than or equal to
173	{	π	Pi
174	|	≠	Not equal to
175	}	£	UK pound sign
176	~	•	Centered dot (bullet)

NOTES

1. Codes 152_8–156_8, 161_8, and 164_8–170_8 are used to draw rectangular grids; each piece of this line drawing set is contiguous with others so that the lines formed are unbroken.

2. Codes 157_8–163_8 give better vertical resolution than dashes and underlines when drawing lines; using these segments 120 × 132 resolution may be obtained in 132 column mode with the Advanced Video Option installed.

2.2.13 Auxiliary Keypad

The keys on the auxiliary keypad normally transmit the codes for the numerals, decimal point, minus sign, and comma. The ENTER key transmits the same code as the RETURN key. The host cannot tell if these keys are typed on the auxiliary keypad or on the main keyboard. Therefore, software that requires numeric data entry may use either keypad.

If software must be able to distinguish between pressing a key on the auxiliary keypad and pressing the corresponding key on the main keyboard, the host can give the terminal a command to place it in keypad application mode. In this mode, all keys on the auxiliary keypad give control functions that may be used by the software as user-defined functions.

The codes sent by the auxiliary keypad for the four combinations of the VT52/ANSI mode and keypad numeric/application mode are shown in Tables 2-8 and 2-9. These keys are not affected by pressing the SHIFT, CAPS LOCK, or CONTROL keys.

> **NOTE**
> **In ANSI mode, if the codes are echoed back to the terminal, or if the terminal is in LOCAL mode, the last character of the sequence is displayed on the screen; e.g., PF1 will display a "P."**

Table 2-8 VT52 Mode Auxiliary Keypad Codes

Key	Keypad Numeric Mode	Keypad Application Mode
0	0	ESC ? p
1	1	ESC ? q
2	2	ESC ? r
3	3	ESC ? s
4	4	ESC ? t
5	5	ESC ? u
6	6	ESC ? v
7	7	ESC ? w
8	8	ESC ? x
9	9	ESC ? y
–	–	ESC ? m
,	,	ESC ? l
•	•	ESC ? n
ENTER	Same as RETURN key	ESC ? M
PF1	ESC P	ESC P
PF2	ESC Q	ESC Q
PF3	ESC R	ESC R
PF4	ESC S	ESC S

Table 2-9 ANSI Mode Auxiliary Keypad Codes

Key	Keypad Numeric Mode	Keypad Application Mode
0	0	ESC O p
1	1	ESC O q
2	2	ESC O r
3	3	ESC O s
4	4	ESC O t
5	5	ESC O u
6	6	ESC O v
7	7	ESC O w
8	8	ESC O x
9	9	ESC O y
–	–	ESC O m
,	,	ESC O l
.	.	ESC O n
ENTER	Same as RETURN key	ESC O M
PF1	ESC O P	ESC O P
PF2	ESC O Q	ESC O Q
PF3	ESC O R	ESC O R
PF4	ESC O S	ESC O S

2.3 TERMINAL CONTROL COMMANDS

The VT105 has many control commands that cause the terminal to take action other than displaying a character on the screen. The host can command the terminal to move the cursor, change modes, ring the bell, etc. The following paragraphs discuss the terminal control commands.

2.3.1 Control Characters

Control characters have values of 000_8 – 037_8, and 177_8. The control characters recognized by the VT105 are shown in Table 2-10. All other control codes cause no action to be taken.

Control characters (codes 0 to 37_8 inclusive) may be imbedded within a control function sequence. Imbedded control characters are executed as soon as they are encountered by the terminal. Then, the control function continues to be processed. The exceptions are as follows.

1. If the character ESC (033_8) occurs, the current control function is aborted and a new one commences.
2. If the character CAN (030_8) or the character SUB (032_8) occurs, the current control function is aborted.

The ability to imbed control characters within sequences allows the synchronization characters XON and XOFF to be interpreted without affecting the control function.

Table 2-10 Control Characters

Control Character	Octal Code	Action Taken
NUL	000	Ignored on input (not stored in input buffer).
ENQ	005	Transmit answerback message.
BEL	007	Sound bell tone.
BS	010	Move the cursor to the left one character position. If cursor is at the left margin, no action occurs.
HT	011	Move the cursor to the next tab stop, or move cursor to the right margin if no further tab stops are present on the line.
LF	012	This code causes a line feed or a new line operation. (See new line mode.)
VT	013	Interpreted as LF.
FF	014	Interpreted as LF.
CR	015	Move cursor to left margin on the current line.
SO	016	Invoke G1 character set, as designated by SCS control function.
SI	017	Select GO character set, as designated by SCS control function.
XON	021	Resume transmission to terminal.
XOFF	023	Stop transmitting to terminal all codes except XOFF and XON.
CAN	030	If sent during an escape or control sequence, the sequence is immediately terminated and not executed. It also causes the error character to be displayed.
SUB	032	Interpreted as CAN.
ESC	033	Introduces an escape sequence.
DEL	177	Ignored on input (not stored in input buffer).

2.4 CONTROL FUNCTIONS

The VT105 is an upward and downward software compatible terminal; that is, previous DIGITAL video terminals have private standards for control functions. The American National Standards Institute (ANSI) has since standardized control functions within terminals in documents X3.41-1974 and X3.64-1977.

The VT105 is compatible with both the previous DIGITAL standard and the ANSI standards. Customers may use existing DIGITAL software designed around the VT52 or new VT100 or VT105 software. The VT105 has a VT52-compatible mode, in which, the terminal responds to control functions like a VT52 or VT55.

Throughout this section of the manual references are made to "VT52 mode" or "ANSI mode." These two terms are used to indicate the software compatibility. All new software should be designed around the ANSI mode. Future DIGITAL video terminals may not be committed to VT52 compatibility.

2.4.1 ANSI Mode Control Functions

Definitions – The following listing briefly defines the basic elements of the ANSI control functions. A more complete listing appears in Appendix B.

1. Control Sequence Introducer (CSI) – A prefix to a control sequence. In the VT105, **ESC [** is the CSI.

2. Parameter –

   a. A string of 0 or more decimal characters that represent a single value. Leading 0s are ignored. The characters have a range of 0 (60_8) to 9 (77_8).

   b. The value so represented.

3. Numeric Parameter – A parameter that represents a number, designated by **Pn**.

4. Selective Parameter – A parameter that selects a subfunction from a specified list, designated by **Ps**.

5. Parameter String – A string of parameters separated by a semicolon (73_8).

6. Default Value – A value that is assumed when no explicit value, or a value of 0, is specified.

7. Final Character – A character that terminates a control sequence or escape sequence.

Examples:

1. General

selective
parameter

$$\underbrace{\textbf{ESC [}}_{\text{CSI}}\underbrace{\textbf{Ps; Ps}}_{\text{Parameter String}}\ \underbrace{\textbf{h}}_{\text{Final Character}}$$

2. Escape sequence for double-width line (DEC DWL).

Escape Sequence	**Octal Representation of Same Sequence**
ESC # 6	033 043 066

3. Control sequence to set 132 column mode.

Control Sequence	**Octal Representation of Same Sequence**
$\underbrace{\text{ESC[}}_{\text{CSI}}\ \underbrace{\text{?3}}_{\text{Parameter}}\ \underbrace{\text{h}}_{\text{Final Character}}$	$\underbrace{\text{033 133}}_{\text{CSI}}\ \underbrace{\text{077 063}}_{\text{Parameter}}\ \underbrace{\text{150}}_{\text{Final Character}}$

Refer to Appendix B for a more complete listing of definitions and notation used in this manual.

All of the following control functions are transmitted from the host computer to the terminal unless otherwise noted. All of the control functions are a subset of those specified in ANSI X 3.64 1977 and ANSI X 3.41 1974.

> **NOTE**
> **Control functions designated by an asterisk have DIGITAL private parameters.**

CPR Cursor Position Report – Terminal to Host

ESC [Pn; Pn R

default value: **1**

The CPR control sequence reports the active position by means of the parameters. This sequence has two parameter values. The first specifies the line, and the second specifies the column. The default condition with no parameters present, or parameters of 0, is equivalent to a cursor at home position.

The numbering of lines depends on the state of the origin mode (DECOM).

This control sequence is solicited by a device status report (DSR) sent from the host.

CUB Cursor Backward – Host to Terminal and Terminal to Host

ESC [Pn D *default value:* **1**

The CUB control sequence moves the active position to the left. The distance moved is determined by the parameter. If the parameter value is zero or one, the active position is moved one position to the left. If the parameter value is n, the active position is moved n positions to the left. If an attempt is made to move the cursor to the left of the left margin, the cursor stops at the left margin. *Editor function*

CUD Cursor Down – Host to Terminal and Terminal to Host

ESC [Pn B *default value:* **1**

The CUD control sequence moves the active position downward without altering the column position. The number of lines moved is determined by the parameter. If the parameter value is zero or one, the active position is moved one line downward. If the parameter value is n, the active position is moved n lines downward. If an attempt is made to move the cursor below the bottom margin, the cursor stops at the bottom margin. *Editor function*

CUF Cursor Forward – Host to Terminal and Terminal to Host

ESC [Pn C *default value:* **1**

The CUF control sequence moves the active position to the right. The distance moved is determined by the parameter. A parameter value of zero or one moves the active position one position to the right. A parameter value of n moves the active position n positions to the right. If an attempt is made to move the cursor to the right of the right margin, the cursor stops at the right margin. *Editor function*

CUP Cursor Position

ESC [Pn; Pn H *default value:* **1**

The CUP control sequence moves the active position to the position specified by the parameters. This sequence has two parameter values. The first specifies the line position, and the second specifies the column position. A parameter value of zero or one for the first or second parameter moves the active position to the first line or column in the display, respectively. The default condition, with no parameters present, is equivalent to a cursor to home action. This control sequence behaves identically with the HVP command. *Editor function*

The numbering of lines depends on the state of the origin mode (DECOM).

CUU Cursor Up – Host to Terminal and Terminal to Host

ESC [Pn A *default value:* **1**

This control sequence moves the active position upward without altering the column position. The number of lines moved is determined by the parameter. A parameter value of zero or one moves the active position one line upward. A parameter value of n moves the active position n lines upward. If an attempt is made to move the cursor above the top margin, the cursor stops at the top margin. *Editor function*

DA Device Attributes

ESC [Pn c *default value:* **0**

1. The host requests the terminal to send a device attributes (DA) control sequence to identify itself. The DA request is a control sequence with either no parameter or a parameter of 0.

2. The response to the request generated by the terminal is a DA control sequence with numeric parameters as follows:

Option Present	**Sequence Sent**
No options	ESC [?1;0c
Processor option (STP)	ESC [?1;1c
Advanced video option (AVO)	ESC [?1;2c
AVO and STP	ESC [?1;3c
Graphic waveform generator option (GPO)	ESC [?1;4c
GPO and STP	ESC [?1;5c
GPO and AVO	ESC [?1;6c
GPO, STP, and AVO	ESC [?1;7c

DECALN Screen Alignment Display*

ESC # 8

This command fills the entire screen area with uppercase Es for screen focus and alignment. This command is used by DIGITAL manufacturing and Field Service personnel.

DECANM ANSI/VT52 Mode*

This parameter is applicable to set mode (SM) and reset mode (RM) control sequences. The reset state causes only VT52-compatible control functions to be interpreted and executed. The set state causes only ANSI-compatible control functions to be interpreted and executed.

DECARM Auto Repeat Mode*

This is a private parameter applicable to set mode (SM) and reset mode (RM) control sequences. The reset state causes no keyboard keys to auto repeat. The set state causes certain keyboard keys to auto repeat.

All keys will auto repeat except: SET-UP, ESC, NO SCROLL, TAB, RETURN, and any key pressed with CTRL key down. Auto repeating works as follows: when a key is typed, its code is sent once, immediately. If the key is held down for more than 1/2 second, its code is sent repeatedly at a rate of approximately 30 Hz until the key is released. This rate is lower at low transmit baud rates.

DECAWM Autowrap Mode*

This parameter is applicable to set mode (SM) and reset mode (RM) control sequences. The reset state causes any displayable characters received when the cursor is at the right margin to replace any previous characters there. The set state causes these characters to advance to the start of the next line, performing scroll up if required and permitted.

DECCKM Cursor Keys Mode*

This parameter is applicable to set mode (SM) and reset mode(RM) control sequences. This mode is only effective when the keypad application mode (DECKPAM) and the ANSI/VT52 mode (DECANM) is set. If the cursor key mode is reset, the four cursor function keys send ANSI cursor control commands. If cursor key mode is set, the four cursor function keys send application functions.

DECCOLM Column Mode*

This parameter is applicable to set mode (SM) and reset mode (RM) control sequences. The reset state causes a maximum of 80 columns on the screen. The set state causes a maximum of 132 columns on the screen.

> **NOTE**
> **The use of double-width characters reduces the number of characters per line by half.**

DECDHL Double Height Line*

Top Half: ESC # 3
Bottom Half: ESC # 4

These escape sequences cause the line containing the active position to become the top or bottom half of a double-height, double-width line. The sequences must be used in pairs on adjacent lines and the same character output must be sent to both lines to form full double-height characters. If the line was single-width single-height, all characters to the right of the center of the screen are lost. The cursor remains over the same character position unless it would be to the right of the right margin, in which case, it is moved to the right margin.

DECDWL Double-Width Line*

ESC # 6

This escape sequence causes the line that contains the active position to become double-width single-height. If the line was single-width single-height, all characters to the right of the center of the screen are lost. The cursor remains over the same character position unless it would be to the right of the right margin, in which case, it is moved to the right margin.

DECGOFF Graphics Waveform Generator Off*

ESC 2

Turn off the graphics waveform generator.

DECGON Graphics Waveform Generator On*

ESC 1

Turn on the graphics waveform generator. All subsequent characters are interpreted as commands or data to the graphics waveform generator option. The terminal remains in this mode until the graphics waveform generator off command (ESC 2) is received. This command is ignored if the option is not installed.

DECHCP Hard Copy*

ESC # 7

This escape sequence signals the hardcopy unit to obtain control of the terminal while it copies the video screen. Data to update the screen ceases until the hardcopy output is obtained, after which, the normal operation resumes. This command is ignored if no hardcopy option is installed.

DECID Identify Terminal*

ESC Z

This function causes the same response as the ANSI device attributes (DA) command. The DA function is preferred over the DECID function for new software.

DECINLM Interlace Mode*

This parameter is applicable to set mode (SM) and reset mode (RM) control sequences. The reset state (noninterlace) causes the video processor to display 240 scan lines per frame. The set state (interlace) causes the video processor to display 480 scan lines per frame. There is no difference in character resolution.

> **NOTE**
> **Noninterlace (reset) mode must be used in the VT105 for graph drawing.**

DECKPAM Keypad Application Mode*

ESC =

Enable the auxiliary keypad keys and cursor control keys to transmit escape sequences as defined in Tables 2-8 and 2-9.

DECKPNM **Keypad Numeric Mode***

ESC >

Return to the numeric keypad mode. The auxiliary keypad keys send ASCII codes corresponding to the characters engraved on the keys.

DECLL **Load LEDs***

ESC [Ps q *default value:* **0**

Light the four programmable LEDs on the keyboard according to the parameter(s).

Parameter	Meaning
0	Clear LEDs L1 through L4
1	Light LED L1
2	Light LED L2
3	Light LED L3
4	Light LED L4

LED numbers are indicated on the keyboard.

DECOM **Origin Mode***

This parameter is applicable to set mode (SM) and reset mode (RM) control sequences. The reset state causes the origin to be at the upper-left character position on the screen. Line numbers are independent of current margin settings. The cursor may be positioned outside the margins with a cursor position (CUP) command or a horizontal and vertical position (HVP) command.

The set state causes the origin to be at the upper-left character position within the margins. Line numbers are relative to the current margin settings. The cursor is not allowed to be positioned outside the margins.

The cursor is moved to the new home position when this mode is set or reset.

Lines and columns are numbered consecutively with the origin being line 1, column 1.

DECRC **Restore Cursor***

ESC 8

This escape sequence causes the previously saved cursor position, graphic rendition, and character set to be restored.

DECREPTPARM **Report Terminal Parameters***

ESC [<sol>; <par>; <nbits>; <xspeed>; <rspeed>; <clkmul>; <flags>x

The parameters for this control sequence are explained in the next paragraph.

DECREQTPARM **Request Terminal Parameters***

ESC [<sol> x

The DECREPTPARM control sequence is sent by the terminal controller to notify the host of the status of selected terminal parameters. The status sequence may be sent when requested by the host or at the terminal's discretion. DECREPTPARM is sent upon receipt of a DECREQTPARM. On power-up or reset, the terminal is inhibited from sending unsolicited reports.

The meanings of the sequence parameters are as follows.

Parameter	Value	Meaning
<sol>	0 or none	This message is a request (DECREQTPARM) and the terminal is allowed to send unsolicited reports. Unsolicited reports are sent when the terminal exits the Set-up mode.
	1	This message is a request; the terminal may only report in response to a request.
	2	This message is a report (DECREPTPARM).
	3	This message is a report and the terminal is only reporting on request.
<par>	1	No parity set
	4	Parity is set and odd
	5	Parity is set and even
<nbits>	1	8 bits per character
	2	7 bits per character
<xspeed>		Bits per second
<rspeed>	0	50
	8	75
	16	110
	24	134.5
	32	150
	40	200
	48	300
	56	600
	64	1200
	72	1800
	80	2000
	88	2400
	96	3600
	104	4800
	112	9600
	120	19200

Parameter	Value	Meaning
<clkmul>	1	The bit rate multiplier is 16.
<flags>	0–15	This value communicates the four switch values in block 5 of Set-up B, which are only visible to the user when a processor option (STP) is installed. These bits may be assigned for an STP device. The four bits are a decimal-encoded binary number.

DECSC Save Cursor*

ESC 7

This escape sequence causes the cursor position, graphic rendition, and character set to be saved. (See DECRC.)

DECSCLM Scrolling Mode*

This parameter is applicable to set mode (SM) and reset mode (RM) control sequences. The reset state causes scrolls to "jump" instantaneously. The set state causes scrolls to be "smooth" at a maximum rate of six lines per second.

DECSCNM Screen Mode*

This parameter is applicable to set mode (SM) and reset mode (RM) control sequences. The reset state causes the screen to be black with white characters. The set state causes the screen to be white with black characters.

DECSTBM Set Top and Bottom Margins*

default value: **(see below)**

ESC [Pn; Pn r

This control sequence sets the top and bottom margins to define the scrolling region. The first parameter is the line number of the first line in the scrolling region; the second parameter is the line number of the bottom line in the scrolling region. Default is the entire screen (no margins). The minimum size of the scrolling region allowed is two lines; that is, the top margin must be less than the bottom margin. The cursor is placed in the home position. (See origin mode DECOM.)

DECSWL Single-Width Line*

ESC # 5

This escape sequence causes the line that contains the active position to become single-width single-height. The cursor remains in the same character position. This is the default condition for all new lines on the screen.

DECTST Invoke Confidence Test*

ESC [2 ; Ps y

Ps is the parameter indicating the test to be done. Ps is computed by taking the value indicated for each desired test and adding them together. If Ps is 0 no test is performed but the terminal is reset.

Test	Value
Power up self-test (ROM checksum, RAM, NVR keyboard, and AVO if installed)	1
Data loop back test	2 (loop-back connector required)
EIA modem control test	4 (loop-back connector required)
Repeat selected test(s) indefinitely until failure or power off.	8

DSR Device Status Report

ESC [Ps n *default value:* **0**

This control sequence requests and reports the general status of the terminal according to the following parameter(s).

Parameter	Parameter Meaning
0	Response from terminal – Ready; no malfunctions detected (default).
3	Response from terminal – Malfunction; retry.
5	Command from host – Report status (using a DSR control sequence).
6	Command from host – Report active position (using a CPR control sequence).

A parameter value of 0 or 3 is always sent in response to a DSR request with a parameter value of 5.

ED Erase In Display

ESC [Ps J *default value:* **0**

Erase some or all of the characters in the display according to the parameter. Any complete line erased by this sequence returns that line to single-width mode.
Editor function

Parameter	Parameter Meaning
0	Erase from the active position to the end of the screen, inclusive (default).
1	Erase from start of the screen to the active position, inclusive.
2	Erase all of the display – all lines are erased, changed to single-width, and the cursor does not move

EL Erase In Line

ESC [Ps K *default value:* **0**

Erase some or all characters in the active line according to the parameter. *Editor function*

Parameter	Parameter Meaning
0	Erase from the active position to the end of the line, inclusive (default).
1	Erase from the start of the screen to the active position, inclusive.
2	Erase all of the line, inclusive.

HTS Horizontal Tabulation Set

ESC H

Set one horizontal tab stop at the active position. *Format effector*

HVP Horizontal and Vertical Position

ESC [Pn ; Pn f *default value:* **1**

Move the active position as specified by the parameters. This control sequence has two parameter values: the first specifies the line position and the second specifies the column. A parameter value of either zero or one causes the active position to move to the first line or column in the display, respectively. The default condition with no parameters present moves the active position to the home position. This command reacts identically with its editor function counterpart, CUP. The numbering of lines and columns depends on the reset or set state of the origin mode (DECOM). *Format effector*

IND Index

ESC D

This escape sequence causes the active position to move downward one line without changing the column position. If the active position is at the bottom margin, a scroll up is performed. *Format effector*

LNM Line Feed/New Line Mode

This is a parameter applicable to set mode (SM) and reset mode (RM) control sequences. The reset state causes the interpretation of the line feed (LF), defined in ANSI Standard X3.64-1977, to imply only vertical movement of the active position and causes the return key (CR) to send the single code CR. The set state causes the LF to imply movement to the first position of the following line and causes the return key to send the two codes (CR and LF). This is the new line (NL) Set-up feature.

This mode does not affect the index (IND), or next line (NEL), format effectors.

NEL Next Line

ESC E

This escape sequence causes the active position to move to the first position on the next line downward. If the active position is at the bottom margin, a scroll up is performed. *Format effector*

RI Reverse Index

ESC M

Move the active position to the same horizontal position on the preceding line. If the active position is at the top margin, scroll down is performed. *Format effector*

RIS Reset To Initial State

ESC c

Reset the VT105 to its initial state as powered on. This also causes the execution of the power-up self-test and signal INIT H to be asserted briefly to clear graph memories.

RM Reset Mode

ESC [Ps ;Ps ; . . .; Ps l

default value: **none**

Reset one or more modes as specified by each selective parameter in the parameter string. Each mode to be reset is specified by a separate parameter. [See set mode (SM) control sequence.]

SCS Select Character Set

The appropriate G0 and G1 character sets are designated from one of the five possible character sets. The G0 and G1 sets are invoked by the codes SI and SO (shift in and shift out) respectively.

G0 Sets	G1 Sets	
Sequence	**Sequence**	**Meaning**
ESC(A	ESC) A	United Kingdom set
ESC(B	ESC) B	ASCII set
ESC(0	ESC) 0	Special graphics character set
ESC(1	ESC) 1	Alternate character ROM standard character set
ESC(2	ESC) 2	Alternate character ROM special graphics set

The United Kingdom and ASCII sets conform to the "ISO international register of character sets to be used with escape sequences." The other sets are DIGITAL private character sets. Special graphics means that the graphic characters for the codes 137_8 to 176_8 are replaced with other characters. The specified character set is used until another SCS is received.

SGR Select Graphic Rendition (Character Attributes)

ESC [Ps; . . . ; Ps m *default value:* **0**

Invoke the character attributes specified by the parameter(s). All subsequent characters transmitted to the video screen react according to the parameter(s) selected until the next occurrence of SGR. *Format effector*

Parameter	Parameter Meaning
0	Attributes off
1	Bold or increased intensity
4	Underscore
5	Blink
7	Negative (reverse) image

All other parameter values are ignored.

Without the Advanced Video Option, only one type of character attribute is possible as determined by the cursor selection. Specify either the blinking underline or the blinking block cursor to activate the character attribute. (See cursor selection in Chapter 1.)

SM Set Mode

ESC [Ps; . . . ; Ps h *default value:* **none**

This control sequence causes one or more modes to be set as specified by each parameter in the parameter string. Each mode to be set is specified by a separate parameter. A mode is considered set until it is reset by a reset mode (RM) control sequence. (See Paragraph 2.4.2 to select the parameters for the modes.)

TBC Tabulation Clear

ESC [Ps g *default value:* **0**

Parameter	Parameter Meaning
0	Clear the horizontal tab stop at the active position (the default case).
3	Clear all horizontal tab stops.

Any other parameter values are ignored. *Format effector*

2.4.2 Modes

The following is a list of modes that may be changed with set mode (SM) and reset mode (RM) control sequences.

2.4.2.1 ANSI Specified Modes

Parameter	Mode Mnemonic	Mode Function
0		Error (ignored)
20	LNM	Line feed/new line mode

2.4.2.2 DIGITAL Private Modes – If the first character in the parameter string is ? (77_8), the parameters are interpreted as DIGITAL private parameters according to the following.

Parameter	Mode Mnemonic	Mode Function	Reset	Set
0	–	Error (ignored)		
1	DECCKM	Cursor key	ANSI functions	Application functions
2	DECANM	ANSI/VT52	VT52 mode	ANSI mode
3	DECCOLM	Column	80	132
4	DECSCLM	Scrolling	Jump	Smooth
5	DECSCNM	Screen Background	Black	White
6	DECOM	Origin	Screen reference	Margin reference
7	DECAWM	Auto wrap	Disabled	Enabled
8	DECARM	Auto repeating	Disabled	Enabled
9	DECINLM	Interlace	Noninterlace	Interlace

Any other parameter values are ignored.

2.4.2.3 Other ANSI Mode States – The following modes, specified in the ANSI X3.641977 standard, may be considered to be permanently set, permanently reset, or not applicable, as noted. Refer to that standard for further information concerning these modes.

Mode Mnemonic	Mode Function	State
CRM	Control representation	Reset
EBM	Editing boundary	Reset
ERM	Erasure	Set
FEAM	Format effector action	Reset
FETM	Format effector transfer	Reset
GATM	Guarded area transfer	NA
HEM	Horizontal editing	NA
IRM	Insertion-replacement	Reset
KAM	Keyboard action	Reset
MATM	Multiple area transfer	NA
PUM	Positioning unit	Reset
SATM	Selected area transfer	NA
SRTM	Status reporting transfer	Reset
TSM	Tabulation stop	Reset
TTM	Transfer termination	NA
VEM	Vertical editing	NA

2.4.3 ANSI Control Function Summary

The following is a summary of the VT105 ANSI control functions. The ANSI mode must be enabled to use the control functions outlined in the following paragraphs. Recognition of the escape sequences is dependent on system software and in some cases the presence of terminal options. The following definitions apply.

1. Pn refers to a numeric parameter in the range of 0 (060_8) to 9 (071_8).
2. Ps refers to a selective parameter from a specified list. Ps has a range of 0 (060_8) to 9 (071_8).
3. Multiple parameters are separated by a semicolon (073_8).
4. If a parameter is omitted or specified to be 0, the default parameter value is used. (For the cursor movement commands, the default parameter value is 1.)

Character Attributes

ESC [Ps;Ps;Ps;...; Ps m

Ps refers to a selective parameter. Multiple parameters are separated by a semicolon (073_8). The parameters are executed in order and have the following meanings.

Parameter (Ps)	Attribute
0 or None	All Attributes Off
1	Bold on
4	Underscore on
5	Blink on
7	Reverse video on

Any other parameter values are ignored.

Character Sets (G0 and G1 Designators)

The G0 and G1 character sets are designated as follows.

Character set	G0 designator	G1 designator
United Kingdom (UK)	ESC (A	ESC) A
U.S. (ASCII)	ESC (B	ESC) B
Special graphics characters and line drawing set	ESC (0	ESC) 0
Alternate character ROM	ESC (1	ESC) 1
Alternate character ROM special graphic characters	ESC (2	ESC) 2

Cursor Movement Commands

Cursor up	ESC [Pn A
Cursor down	ESC [Pn B
Cursor forward (right)	ESC [Pn C
Cursor backward (left)	ESC [Pn D
Direct cursor addressing	ESC [Pl; Pc H *or* ESC [Pl; Pc f
Index	ESC D
Reverse index	ESC M
Save cursor and attributes	ESC 7
Restore cursor and attributes	ESC 8

Pl = line number; Pc = column number; default character = 1.

Erase

From cursor to end of line	ESC [K *or* ESC [0 K
From beginning of line to cursor	ESC [1 K
Entire line containing cursor	ESC [2 K
From cursor to end of screen	ESC [J *or* ESC [0 J
From beginning of screen to cursor	ESC [1 J
Entire screen	ESC [2 J

Line Size (Double-Height and Double-Width) Commands

Change this line to double-height top half	ESC # 3
Change this line to double-height bottom half	ESC # 4
Change this line to single-width single-height	ESC # 5
Change this line to double-width single-height	ESC # 6

Modes

	To Set		To Reset	
Mode Name	**Mode**	**Sequence**	**Mode**	**Sequence***
Line feed/new line	New line	ESC [20h	Line feed	ESC [20 l
Cursor key mode	Application	ESC [?1h	Cursor	ESC [?1 l
ANSI/VT52 mode	ANSI	*N/A*	VT52	ESC [?2 l
Column mode	132 Col	ESC [?3h	80 Col	ESC [?3 l
Scrolling mode	Smooth	ESC [?4h	Jump	ESC [?4 l
Screen mode	Reverse	ESC [?5h	Normal	ESC [?5 l
Origin mode	Relative	ESC [?6h	Absolute	ESC [?6 l
Wraparound	On	ESC [?7h	Off	ESC [?7 l
Auto repeat	On	ESC [?8h	Off	ESC [?8 l
Interlace	On	ESC [?9h	Off	ESC [?9 l
Waveform Generator	On	ESC 1	Off	ESC 2
Keypad mode	Application	ESC =	Numeric	ESC >

Programmable LEDs

ESC [Ps;Ps;...Ps q

Ps refers to a selective parameter. Multiple parameters are separated by semicolons (073_8) and executed in order, as follows.

Parameter (Ps)	**LED Selected**
0 or None	All LEDs Off
1	LED L1 On
2	LED L2 On
3	LED L3 On
4	LED L4 On

Any other parameter values are ignored.

Reports

Cursor Position Report

Invoked by: ESC [6 n

Response is: ESC [Pl ; Pc R
Pl = line number and Pc = column number.

Status Report

Invoked by: ESC [5 n

Response is: ESC [0 n (*terminal ok*)
ESC [3 n (*terminal not ok*)

*Last character in sequence is lower case L.

What Are You? (Identify Report)

Invoked by: ESC [c
or
ESC [0 c

Response is: ESC [? 1 ; Ps c

Ps is the "option present" parameter with the following meaning:

Ps	Meaning
0	Base VT100, no options
1	Processor option (STP)
2	Advanced video option (AVO)
3	AVO and STP
4	Graphic Waveform Generator Option (GPO)
5	GPO and STP
6	GPO and AVO
7	GPO, STP, and AVO

The identify report is alternately invoked by ESC Z sequence; however, this is not recommended for new software. The response is the same.

Reset

Reset causes the power-up reset routine to be executed.

ESC c

Scrolling Region

ESC [Pt ; Pb r

Pt is the number of the top line of the rolling region; Pb is the number of the bottom line of the scrolling region and must be greater than Pt.

Tab Stops

Set tab at current column	ESC H
Clear tab at current column	ESC [g *or* ESC [0 g
Clear all tabs	ESC [3 g

Tests

Fill Screen with "Es"	ESC # 8
Invoke Test(s)	ESC [2 ; Ps y

Ps is the parameter indicating the test to be done and is a decimal number computed by taking the "value" indicated for each desired test and adding them together.

Test	Value
Power-up self-test (ROM checksum, RAM, NVR, keyboard, and AVO if installed)	1
Data loop-back test	2 (Loop-back connector required)
EIA modem control test	4 (Loop-back connector required)
Repeat selected test(s) indefinitely until failure or power off.	8

2.4.4 VT52 Mode Control Functions

The VT52 mode must be enabled to use the following control functions. Table 2-11 provides a brief summary of the valid VT52 mode control functions, and the following paragraphs explain the functions in detail.

Table 2-11 VT52 Mode Control Functions

Control Function/Action	Escape Sequence
Cursor Up	ESC A
Cursor Down	ESC B
Cursor Right	ESC C
Cursor Left	ESC D
Select Special Graphics character set	ESC F
Select ASCII character set	ESC G
Cursor to home	ESC H
Reverse line feed	ESC I
Erase to end of screen	ESC J
Erase to end of line	ESC K
Direct cursor address	ESC Ylc (see note 1)
Identify	ESC Z (see note 2)
Enter alternate keypad mode	ESC =
Exit alternate keypad mode	ESC >
Waveform generator ON	ESC 1 (see note 3)
Waveform generator OFF	ESC 2 (see note 3)
Enter ANSI mode	ESC <

NOTES:

1. **Line and column numbers for the direct cursor address are single character codes with octal values of the desired number plus 37_8. Line and column numbers start at 1.**
2. **Response to ESC Z is ESC / Z.**
3. **Ignored if waveform generator option is not installed.**

Waveform Generator ON

ESC 1

Turn on the waveform generator. All subsequent characters are interpreted as commands to the graphics waveform generator option until ESC 2 is received. This sequence is ignored if this option is not installed.

Waveform Generator OFF

ESC 2

Turn off the waveform generator.

Cursor Up

ESC A

Move the active position upward one position without altering the horizontal position. If an attempt is made to move the cursor above the top margin, the cursor stops at the top margin.

Cursor Down

ESC B

Move the active position down one position without altering the horizontal position. If an attempt is made to move the cursor below the bottom margin, the cursor stops at the bottom margin.

Cursor Right

ESC C

Move the active position to the right. If an attempt is made to move the cursor to the right of the right margin, the cursor stops at the right margin.

Cursor Left

ESC D

Move the active position one position to the left. If an attempt is made to move the cursor to the left of the left margin, the cursor stops at the left margin.

Enter Special Graphic Character Mode

ESC F

This command causes the special graphics character set to be used.

> **NOTE**
> **The special graphics characters in the VT105 are different from those in the VT52.**

Exit Special Graphic Character Mode

ESC G

This function returns the terminal to the standard ASCII character set.

Cursor to Home

ESC H

Move the cursor to the home position.

Reverse Line Feed

ESC I

Move the active position up one position without altering the column position. If the active position is at the top margin, a scroll down is performed.

Erase to End of Screen

ESC J

Erase all characters from the active position to the end of the screen. The active position is not changed.

Erase to End of Line

ESC K

Erase all characters from the active position to the end of the current line. The active position is not changed.

Direct Cursor Addressing

ESC Y line column

Move the cursor to the specified line and column. The line and column numbers are sent as ASCII codes whose values are the number plus 037_8; for example, 040_8 refers to the first line or column, 050_8 refers to the eighth line or column, etc.

Identify

ESC Z

This function causes the terminal to send an identifier escape sequence to the host as follows.

ESC / Z

NOTE
Information regarding options must be obtained in ANSI mode, using the device attributes (DA) control sequence.

Enter Alternate Keypad Mode

ESC =

The auxiliary keypad keys send unique identifiable escape sequences for use by applications programs.

Exit Alternate Keypad Mode

ESC >

The auxiliary keypad keys send the ASCII codes for the functions or characters engraved on the key.

Enter ANSI Mode

ESC <

Entering this mode allows the terminal to recognize ANSI-compatible control functions. (Refer to Paragraph 2.4.1.) The VT52 control functions in this section will not be recognized.

2.5 COMMUNICATION PROTOCOL

2.5.1 Full Duplex – XON/XOFF Response

This terminal can operate at transmission speeds up to 19,200 baud. However, the terminal may not be able to keep up with incoming data. The terminal stores incoming characters in a 64-character buffer and processes them on a first-in/first-out basis. When the contents of the buffer reaches 32 characters, the terminal transmits 023_8 (XOFF or DC3). On this signal, the host should suspend its transmission to the terminal. If the host stops transmitting, the terminal soon depletes the buffer. When 16 characters remain in the buffer, the terminal transmits 021_8 (XON or DC1) to signal the host to resume transmission.

The terminal always recognizes received XOFF and XON. Receipt of XOFF inhibits transmission of any codes except XOFF and XON. From three to seven keystrokes on the keyboard will be stored in a keyboard buffer. (Some keys transmit two or three codes, e.g., cursor controls). If the keyboard buffer overflows, keyclicks stop and the KBD LOCKED LED turns on. Transmission resumes upon receipt of XON.

Also, entering Set-up mode causes the terminal to temporarily stop taking characters from the buffer. An XOFF is sent if the buffer becomes nearly full.

Entering and exiting SET-UP clears the keyboard locked condition.

Calculating "Buffer Overflow" – If the host fails to respond to an XOFF from the terminal, the buffer continues to fill. When the 64-character capacity of the buffer is exceeded, a condition occurs called "buffer overflow." To determine if the buffer will overflow, use the following formulas.

No. of characters to overflow	=	32 – [3 × (receive speed transmit speed)]
Time to respond to XOFF	=	No. of characters to overflow × (bits per character + parity bit + 2) /receive speed

Example 1:
The terminal is transmitting 8-bit characters with no parity at 1200 baud and receiving at 1200 baud. The terminal sends an XOFF. The formula applies as follows.

No. of characters to overflow = 32 – [3 × (1200/1200)] = 29 characters

Time to respond to XOFF = 29 × (8+0+2)/1200 = 0.2416 second

Conclusion: The host must respond to the XOFF within 0.2416 second to avoid a buffer overflow.

Example 2:
The VT105 is transmitting 7-bit characters (with parity) at 300 baud and is receiving at 1200 baud. The terminal sends an XOFF.

No. of characters to overflow = 32 – [3 × (1200/300)] = 20 characters

Time to respond to XOFF = 20 × (7+1+2)/1200 = 0.1666 second

Conclusion: The host must respond within 0.1666 second to avoid a buffer overflow. If the buffer overflows, the VT105 will begin to discard incoming characters and the error character (▒) will be displayed.

Software that does not support the AUTO XON/XOFF feature can still use this terminal if the following conditions are met.

1. The software never sends an ESC code to the terminal.
2. The baud rate is limited to 4800 or less.
3. The software does not use smooth scrolling or split screen features.

Alternatively, if the AUTO XON/XOFF feature cannot be used, fill characters may be used after certain characters or character strings are sent to the terminal. Refer to Appendix E for fill characters required for specific control functions.

2.5.2 Reset and Self-Test

Reset and self-test routines initialize the terminal and erase the input buffer. If characters are received and placed in the buffer after either of these commands, the characters are destroyed.

To compensate for this, the host may act in one of two ways.

1. Immediately after sending the commands to perform either the reset or self-test functions, the host may act as if it had received XOFF from the terminal, thus sending no more characters until it receives XON. The terminal transmits XON after it completes the specified operation if the AUTO XOFF/XON feature is enabled.

2. When the first method cannot be implemented, a delay of 10 seconds or more may be used to allow the terminal time to complete the function. Future options may require a change in the time delay. This method, however, does not guarantee against the loss of data when an invoked function has detected an error.

The XOFF/XON synchronization scheme has an advantage over requiring the host to insert delays or filler characters in its data stream. Requiring a minimum of software support, XON/XOFF ensures that every character or command sent to the terminal is processed in correct order. It frees interface programs from all timing considerations and results in a more reliable operation.

2.5.3 NO SCROLL and CTRL S/CTRL Q

There are two other means of transmitting XOFF and XON – the NO SCROLL key, and CTRL S/CTRL Q. If the AUTO XON/XOFF feature is enabled, the terminal coordinates these operations so that the desired effect occurs. For example: If the buffer-filling condition causes an XOFF to be sent and the operator types the NO SCROLL key, a second XOFF is not sent. However, instead of sending an XON when the buffer empties, the terminal waits until the operator types the NO SCROLL key again before sending XON. Use of CTRL S and CTRL Q is synchronized with the NO SCROLL key.

If the AUTO XON/XOFF feature is disabled, the buffer-filling condition does not send an XOFF, the NO SCROLL key is disabled, and CTRL S and CTRL Q are transmitted as typed.

If the user transmits an XOFF to the host (by CTRL S or NO SCROLL), the host should not send any characters until the user transmits XON (by typing CTRL Q or the NO SCROLL key again).

CHAPTER 3

GRAPH DRAWING

3.1 INTRODUCTION

The graph drawing features of the VT105 terminal are provided by the waveform generator module, M7071. It gives the terminal the capability of plotting graphs, histograms, strip charts, graph markers, and vertical and horizontal lines.

This chapter describes how to:

- Select the graph drawing mode
- Establish the graph area
- Establish desired display
- Load graph data
- Generate shade lines, cursors, and grid
- Enter strip chart data.

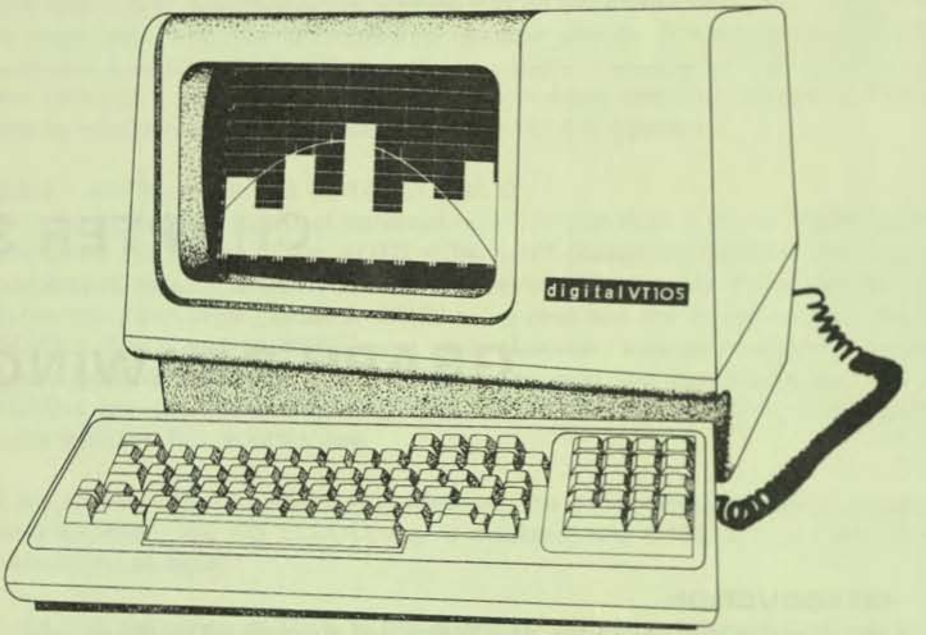
> **NOTE**
> **Should any of the features described in this chapter fail to perform properly, an interactive graphics test is provided in Appendix A to check the performance of the M7071 module in LOCAL mode.**

3.2 ENTERING THE GRAPH DRAWING MODE

Receiving an escape (033_8) followed by the number 1 (061_8) or typing ESC 1 in LOCAL mode switches the terminal to its graph drawing mode. The terminal will remain in this mode until ESC 2 is received by the terminal whether typed OFF LINE or sent by the host.

3.3 DEFINITIONS AND LIMITATIONS

GRAPH A graph is a series of points representing the variation in value of two variables: X and Y (Figure 3-1). For each horizontal value (x), there can be only one Y value; example, a sine wave.

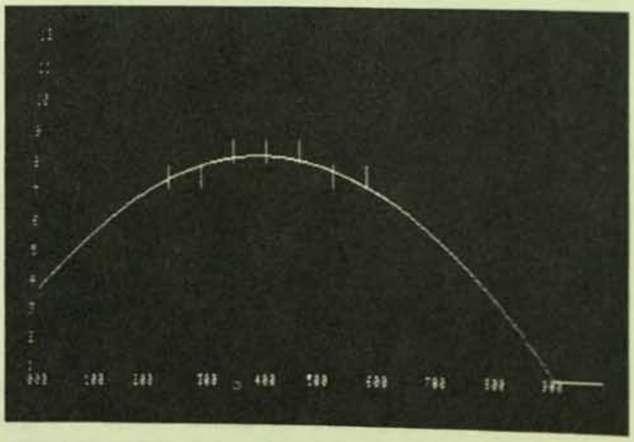


MR-3518

Figure 3-1 VT105 with Graph Display

GRAPH MARKER

A graph marker is a short vertical line that may be programmed to mark any point of the graph. Each marker represents a specific value of X and appears at intervals of Y = 240/16. As many as 512 markers can be placed on each graph or histogram for a total of 1024 markers. See Figure 3-2.



MR-3512

Figure 3-2 Graph with Graph Markers

HISTOGRAM (SHADED GRAPH)

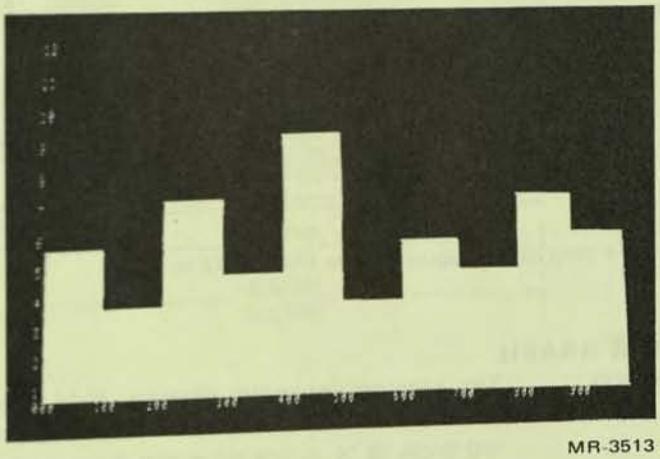
A histogram in the VT105 is a graphic display that has the area between the graph line and the bottom of the graph drawing field intensified. Two histograms can overlap and still be discernible. A bar graph is an example of a histogram, as shown in Figure 3-3.

SHADE LINE

A shade line is the line referenced for shading a graph. One shade line can be displayed for each of the two graphs. If no shade line is established, the graph can be shaded to the bottom of the graph drawing area. (See Histogram.)

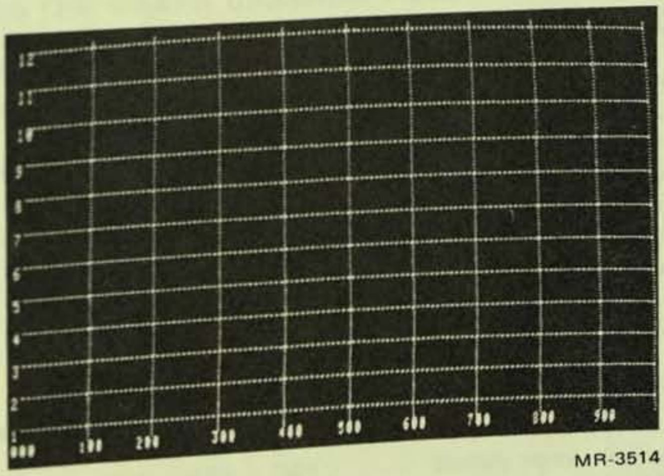
HORIZONTAL LINES/ VERTICAL LINES

Horizontal lines and/or vertical lines may be displayed in the graph area to represent set values of X or Y. As many as 512 vertical or 240 horizontal lines may be individually displayed on the screen. For example, a grid is displayed using specific values for horizontal and vertical lines, as shown in Figure 3-4.



MR-3513

Figure 3-3 Histogram Display (Shaded Graph)



MR-3514

Figure 3-4 Example of a Grid Display

STRIP CHART

A strip chart is a graph or histogram that permits new data to be added to its right side while shifting previous data to the left, as in Figure 3-5. Vertical and horizontal lines, if present, move from right to left as the strip chart moves, and wrap around the screen as they leave the left edge.

DUAL STRIP CHART

Dual strip chart is a feature that allows both Graph 0 and Graph 1 to pass across the screen.

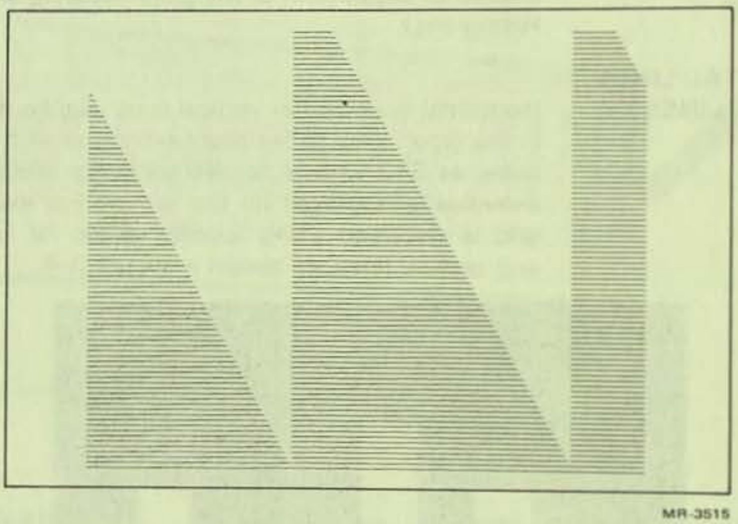


Figure 3-5 Strip Chart Display (Moves From Right to Left)

RECTANGULAR GRAPH DRAWING FIELD

The rectangular graph drawing field is one of two aspect ratios selectable within the VT105. It has a 20 × 10.9 cm (8 × 4-3/8 in) graph drawing field compatible with previous DIGITAL graph drawing terminals (e.g., VT55). See Figure 3-6.

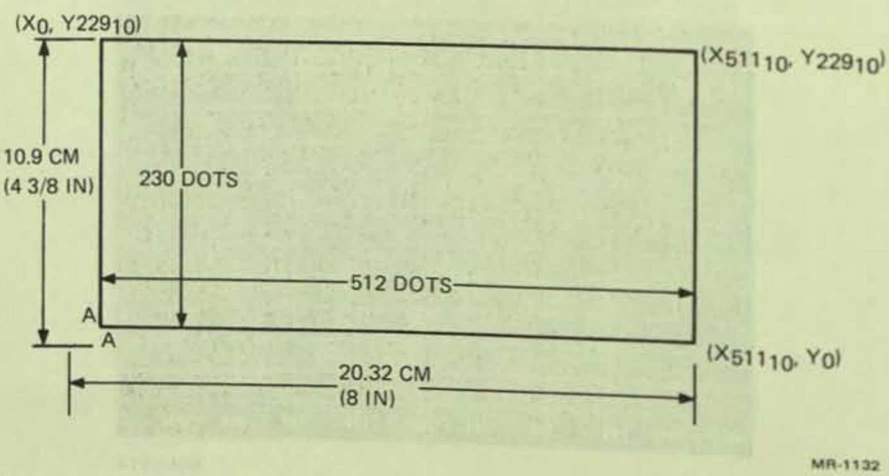


Figure 3-6 Rectangular Graph Drawing Field

SQUARE GRAPH DRAWING FIELD

The square graph drawing field is a selectable aspect ratio with a 16.5 × 11.5 cm (6.5 × 4.6 in) graph drawing field. It compresses the X-axis length and provides a greater area outside the field for alphanumeric labels, grid identifiers, or notes. See Figure 3-7.

GRAPH RESOLUTION

Up to 512 horizontal and 230 vertical points may be displayed per graph in the rectangular graph drawing field; 512 × 240 points, in the square graph drawing field.

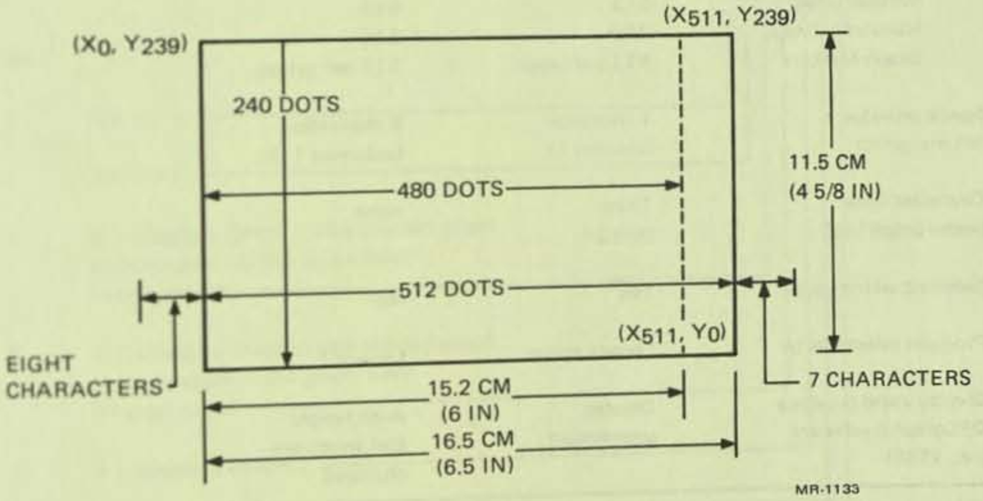


Figure 3-7 Square Graph Drawing Field

3.4 SELECTING THE GRAPH DRAWING FIELD

The VT105 has two selectable graph drawing fields or formats. Both formats display up to two 512-point graphs having single valued functions of X. Either or both graphs can be displayed as a histogram or a strip chart. Graphs and histograms can overlap and still be discernible, allowing the use of the entire field for both graphs, both histograms, or a graph and a histogram displayed together. Both formats display individually programmable horizontal and vertical lines over the entire graph field allowing a grid to represent any desired value. Graph markers can be displayed in either format. Table 3-1 compares the formats and how each is selected.

Example:

Character Sequence	Octal Code Sequence	Field
I space !	111 040 041	Enable square format
I space space	111 040 040	Enable rectangular format

Table 3-1 Comparison of Graph Drawing Formats

Feature	Rectangular Format	Square Format
Graph Field (maximum)	20 × 10.9 cm (8 × 4.3 in)	16.5 × 11.56 cm (6.5 × 4.625 in)
Graph Resolution (maximum)	512 × 230 points	512 × 240 points
Features		
Graphs	2	2
Shaded Graphs	2	2
Movable Shade lines	1 per graph	1 per graph
Strip Charts	2	2
Vertical Lines	512	512
Horizontal Lines	230	240
Graph Markers	512 per graph	512 per graph
Space provided in left margin for:	1 character (column 1)	8 characters (columns 1–8)
Character lines below graph field:	1 line (line 24)	none
Selected on Initialize	Yes	No
Program selectable by:	I space space	I space !
Display using previous DECgraphic software (i.e., VT55)	Display unchanged	Both height and width are changed

3.5 SELECTING DESIRED DISPLAY

Enabling graphs, histograms, strip charts, and baselines is accomplished by loading control register 0 in the VT105; enabling graph markers, vertical lines, and horizontal lines is accomplished by loading control register 1. The registers are loaded by sending a two or three character sequence from the keyboard or host computer. The number of characters depends on the type of graph or baseline desired. The control bits are encoded as 7-bit ASCII characters that are sent to the terminal.

3.5.1 Loading Control Register 0

To determine the required bits to set in register 0, refer to Table 3-2. A seven digit binary code transmits the desired display. The bits are set as desired, for example: Bit 2 is set to enable Graph 1; bit 0 is set to enable the display. The binary code created is 0100101 (045_8). For examples of loading register 0, see Table 3-3 and Table 3-4.

Table 3-2 Load Enable Register 0 Command

Rectangular or Square Format

1st character: A (101_8)
2nd character: variable (see below)
3rd character: variable (see below)

Explanation:
The second character is formed by setting bits where the bits have the following functions:

2nd Character (Register 0)

Binary Code 0 1 X X X X X

Bit	
6	Always 0.
5	Always 1.
4	If 1, displays Graph 1 as a shaded graph to the bottom of the graph field (Histogram 1).
3	If 1, displays Graph 0 as a shaded graph to the bottom of the graph field (Histogram 0)
2	If 1, displays Graph 1.
1	If 1, displays Graph 0.
0	If 1, enables all graphic information to be displayed.

NOTES

1. **Enabling Graph 0 and Histogram 0 (or Graph 1 and Histogram 1) at the same time intensifies the graph envelope.**

2. **Enabling Shade Line 0 (or Shade Line 1) automatically shades Graph 0 (or Graph 1); Histogram 0 (bit 3) and Histogram 1 (bit 4) do not need to be enabled when shade lines are used.**

Table 3-2 Load Enable Register 0 Command (Cont)

3rd Character (Register 0)

Binary Code
0 1 X X X X X X

Bit	
6	Always 0.
5	Always 1.
4	If 1, allows Graph 0 and 1 data to be shifted from right to left (dual strip chart feature).
3	If 1, allows Graph 0 or 1 data to be shifted from right to left (single strip chart enabled).
2	If 1, displays Graph 1 as a shaded graph, referenced to Shade Line 1.
1	If 1, displays Graph 0 referenced to Shade Line 0.
0	If 0, allows Shade Line 0 data to be loaded by the second word of the @ instruction; if 1, allows Shade Line 1 data to be loaded by the second word of the @ instruction.

NOTE
Bit 3 enables the single strip chart feature; either Graph 0 or Graph 1 data, but not both, can be incremented from right to left.

Bit 4 enables the dual strip chart feature. With Graph 0 enabled, Graph 0 data is entered, but the data does not shift at this time. Graph 1 is enabled and Graph 1 data is entered, then both Graph 0 and Graph 1 will shift their data one increment to the left.

Table 3-3 Examples of Selecting Graphs or Histogram

Function Enabled	Character Sequence*	Octal Code Sequence		Binary Code of 2nd Character	Decimal Value
Graph 0	A#	101	043	0100011	35
Graph 1	A%	101	045	0100101	37
Graphs 0 and 1	A'	101	047	0100111	39
Histogram 0	A)	101	051	0101001	41
Histogram 1	A1	101	061	0110001	49
Histograms 0 and 1	A9	101	071	0111001	57
Graph 0 and Histogram 1	A3	101	063	0110011	51
Graph 1 and Histogram 0	A–	101	055	0101101	45
Disable graphs and histograms	A space	101	040	0100000	32

*A third character is used to enable (or disable) shaded graphs (referenced to a shade line) and to enable strip charts. See Table 3-4.

Table 3-4 Examples of Selecting Shaded Graphs with Shade Lines and Strip Charts

Function Enabled	Third Character	Octal Code	Binary Code	Decimal Value
Load Shade Line 0*	"	042	0100010	34
Load Shade Line 1*	%	045	0100101	37
Enable Shaded Graphs with Shade Line 0 and 1	&	046	0100110	38
Enable single strip chart	(	050	0101000	40
Enable strip chart with shaded Graph 0 and Shade Line 0	)	052	0101010	42
Dual strip chart	0	060	0110000	48
Dual strip chart with shaded graphs and shade lines	6	066	0110110	54

*Loading or moving the shade line in the VT105 does not affect Graph 0 or Graph 1 data. See Paragraph 3.7 for loading shade line position.

Table 3-5 can be used to convert the binary codes created for register 0 to the program requirements.

Example:

Function	Octal Code	Character	Binary Code
Enable Graph 1	045	%	0100101

3.5.2 Enabling Graphs and Histograms (Shaded Graphs)

The second character in a sequence for loading register 0 selects the graph or histogram to be displayed. The letter A (101_8) allows register 0 to be loaded.

Table 3-5 Graph Drawing Characters

Character	Octal Code	Binary Code	Decimal Value
SPACE	040	0100 000	32
!	041	0100 001	33
"	042	0100 010	34
#	043	0100 011	35
$	044	0100 100	36
%	045	0100 101	37
&	046	0100 110	38
' (apostrophe)	047	0100 111	39
(	050	0101 000	40
)	051	0101 001	41
*	052	0101 010	42
+	053	0101 011	43
, (comma)	054	0101 100	44
– (minus)	055	0101 101	45
.	056	0101 110	46
/	057	0101 111	47
0	060	0110 000	48
1	061	0110 001	49
2	062	0110 010	50
3	063	0110 011	51
4	064	0110 100	52
5	065	0110 101	53
6	066	0110 110	54
7	067	0110 111	55
8	070	0111 000	56
9	071	0111 001	57
:	072	0111 010	58
;	073	0111 011	59
<	074	0111 100	60
=	075	0111 101	61
>	076	0111 110	62
?	077	0111 111	63

3.5.3 Enabling Strip Charts and Shade Lines

The third character in the load register 0 sequence enables shaded graphs and strip charts. Some of the common functions enabled by the third character are listed in Table 3-4.

3.5.4 Loading Control Register 1

The characters required to enable graph markers and grid lines are formed by setting the appropriate bits in the diagrams in Table 3-6, then finding the character for the code created in Table 3-5.

3.5.5 Enabling Graph Markers, Vertical Lines, and Horizontal Lines

Turning graph markers and grid lines ON and OFF is accomplished by loading register 1. The character sequence is the same in either graph drawing format, except a third character establishes the desired format. The exclamation mark, !(041_8), enables the square format; a SPACE (040_8) enables the rectangular format. The rectangular format is enabled also on initializing the terminal.

Table 3-7 shows examples of enabling graph markers, horizontal lines, and vertical lines.

Table 3-6 Load Enable Register 1 Command

1st character: I (111_8) 73_{10}
2nd character: variable (see below)
3rd character: variable (see below)

2nd Character (Register 1)

Binary Code
0 1 X X X X X

Bit	
6	The most significant bit is always 0.
5	Always 1.
4	If 1, clears the static RAMs containing any vertical lines, horizontal lines, Graph 0, Graph 1, and any graph markers. It does not clear the load enable registers – Register 0 and Register 1.
3	If 1, allows Graph 1 markers to be displayed.
2	If 1, allows Graph 0 markers to be displayed.
1	If 1, allows vertical lines to be displayed.
0	If 1, allows horizontal lines to be displayed.

Table 3-6 Load Enable Register 1 Command (Cont)

3rd Character (Register 1)

Binary Code
0 1 X X X X X

Bit	
6	Always 0.
5	Always 1.
2–4	(unused)
1	If 1, enable Interactive Graphics Test (See Appendix A.)
0	If 0, the rectangular graph drawing field is selected; if 1, the square field is selected. The rectangular field is also enabled by an initialize signal (INIT) during power-up.

Table 3-7 Examples of Selecting Graph Markers, Horizontal Lines, and Vertical Lines

> **NOTE**
> **Remember to bit map all options correctly when sending out a command. Setting a desired feature may reset other features if all options are not considered.**

Function Enabled	Character Sequence	Octal Sequence* 1st	Octal Sequence* 2nd	Binary Code of 2nd Char.	Decimal Value
Clear graph drawing memories	I0	111	060	0110000	48
Enable horizontal and vertical lines	I#	111	043	0100011	35
Enable horizontal lines only	I!	111	041	0100001	33
Enable vertical lines only	I"	111	042	0100010	34
Enable Graph 0 markers	I$	111	044	0101100	36
Enable Graph 1 markers	I(	111	050	0101000	40
Enable Graph 0 and 1 markers	I,	111	054	0101100	44
Enable grid and Graph 0 markers	I'	111	047	0100111	39
Enable grid and Graph 1 markers	I+	111	053	0101011	43
Enable grid and Graph 0 and 1 markers	I/	111	057	0101111	47
Disable lines and graph markers	I space	111	040	0100000	32

*A third character is required to establish the square format, to change formats, or to set up the Interactive Graphics Test.

Table 3-7 Examples of Selecting Graph Markers, Horizontal Lines, and Vertical Lines (Cont)

To load coordinates for the function enabled by register 1, refer to the following paragraphs.

Function	Paragraph
Load graph markers	3.10
Load vertical lines	3.11
Load horizontal lines	3.12

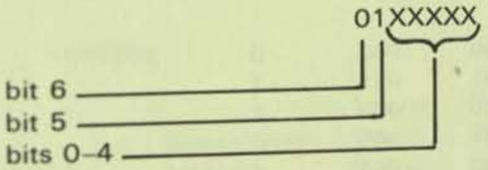
3.6 FORMING GRAPH DRAWING DATA CHARACTERS

In order to represent a horizontal or vertical address of a point on a 240 $\times$ 512 point graph, at least 9 binary bit positions are required.

Examples: $239_{10} = 011101111_2$
$511_{10} = 111111111_2$

Keys typed from the keyboard, or transmitted from the host computer, normally only contain 7 digits (7-bit ASCII characters). Therefore, two keys must be typed, or two codes transmitted, to fully describe an X or a Y value. The first key (character) transmits the lower five bits of the binary data value; the second key transmits the remaining bits, or upper data value.

The graph drawing data characters can be standardized to the 32 characters listed in Table 3-8 if bits 5 and 6 of each character are always 1 and 0, respectively, as in the following format.



The data value is then transmitted in two parts as in the diagram below.

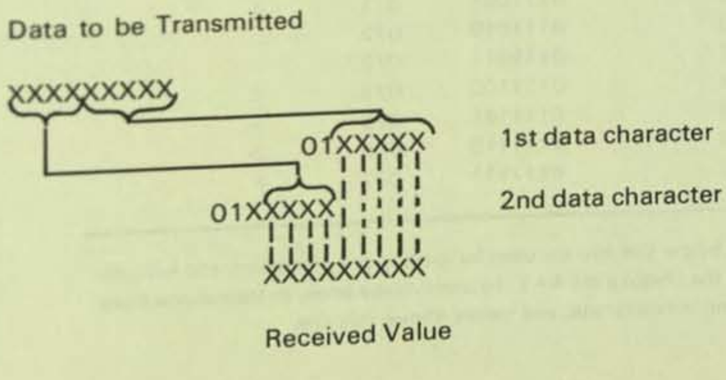


Table 3-8 Graph Drawing Data Characters

Decimal Value Lower data 1st data	 Upper data 2nd data	Binary Code	Octal Code	Data Character
	Load shade line and graph data; erase other lines.			
0	0	0100000	040	SPACE
1	32	0100001	041	!
2	64	0100010	042	"
3	96	0100011	043	#
4	128	0100100	044	$
5	160	0100101	045	%
6	192	0100110	046	&
7	224	0100111	047	' (apostrophe)
8	256	0101000	050	(
9	288	0101001	051	)
10	320	0101010	052	*
11	352	0101011	053	+
12	384	0101100	054	, (comma)
13	416	0101101	055	– (minus)
14	448	0101110	056	.
15	480	0101111	057	/
	Load horizontal, vertical lines, and markers *			
16	0	0110000	060	0
17	32	0110001	061	1
18	64	0110010	062	2
19	96	0110011	063	3
20	128	0110100	064	4
21	160	0110101	065	5
22	192	0110110	066	6
23	224	0110111	067	7
24	256	0111000	070	8
25	288	0111001	071	9
26	320	0111010	072	:
27	352	0111011	073	;
28	384	0111100	074	<
29	416	0111101	075	=
30	448	0111110	076	>
31	480	0111111	077	?

*Upper data values below this line are used for loading graph markers and horizontal or vertical lines; they require bit 4=1. To erase these lines, or load shade lines, graph data or starting X-coordinate, use values above this line.

3.6.1 Selecting Upper and Lower Data Characters

The value of the lower data ranges from 0 to 31_{10}; the upper data value increases in increments of 32 decimal units. Together they can describe any value of the graph drawing field. To find the characters or code to transmit a desired location, perform the following.

1. Select the upper data value closest, but not exceeding, the desired value in Table 3-8. This is the second data character which is transmitted last.

   Example:

Value To be Transmitted	Nearest Upper Data Value	Second Data Character	Binary Code	Octal Code
200_{10}	192_{10}	&	0100110_2	046_8

   NOTE

   If a horizontal line, vertical line, or graph marker is to be loaded, use values in the lower half of the upper data column; i.e., 192_{10} is 066_8 or the character 6.

2. Find the remainder of the value to be transmitted in the lower data value column. This will be the first data character transmitted.

   Example:

Remaining Value	Lower Data Value	First Data Character	Binary Code	Octal Code
8	8	(	0101000_2	050_8

3.6.2 Load Data Sequences

The data to be transmitted is initially preceded by a "load character," as described in Table 3-9. In the above example, loading a shade line at line 200_{10} is transmitted by @(& or equivalent program. Storing a horizontal line at line 200_{10} is transmitted by D(6 or equivalent program.

For multiple data entries, the load character does not need to be repeated. This allows data for a graph to be loaded into memory without repeating the character B or J. Exceptions to this procedure are lóading a shade line and loading the starting X-coordinate.

Table 3-9 Load Data Sequences

Function	Load Character	Character Sequence	Range
Load shade line	@	@ 1st data 2nd data	0–239
Load Graph 0 data	B	B 1st data 2nd data	0–239
Load Graph 1 data	J	J 1st data 2nd data	0–239
Load Graph 0 marker	C	C 1st data 2nd data	0–511
Load Graph 1 marker	K	K 1st data 2nd data	0–511
Load horizontal line	D	D 1st data 2nd data	0–239
Load vertical line	L	L 1st data 2nd data	0–511
Load starting X-coordinate	H	H 1st data 2nd data	0–511

3.6.3 Frequent Data Entry Errors

Largest Data Character Transmitted First – The low-order bits of the data value (lower data value in Table 3-8) must be transmitted first; if reversed, the point will typically exceed the range limits of the graph drawing field and not be displayed, or, it may appear near the edge of the field.

Example:

Desired Shade line	Data Transmitted	Characters Should be	Characters Reversed	New Values
40	8 + 32	@(!	@!(	1+256
70	6 + 64	@&"	@"&	2+192

Zero Valued Characters Not Transmitted – When transmitting data with two characters, if the upper data value is equal to the point or line desired, a SPACE (040_8) (equal to zero) must be the first data character. If missed, the line or point will be much less than desired.

Example:

Shade line Desired	Data Required	Characters Should be	Characters In error	Wrong Value
160	0+160	@ SPACE %	@%	5

3.7 LOADING THE SHADE LINE

A movable shade line can be displayed for both Graph 0 and Graph 1. One shade line can be entered for Graph 0, and one for Graph 1, on any of the 230 (or 240, square format) horizontal lines available. To transmit positions within this range, a load character and two data characters are used, as shown in Table 3-10. Positions can be created by following Paragraph 3.6 for forming graph drawing data characters.

Table 3-10 Loading Shade Line Position

	Previous VT55 Format	New VT105 Format Load Shade Line
1st Character:	@ NOP	@ (100_8) 64_{10}
2nd Character:	none	01XXXXX
3rd character:	none	01XXXXX

NOTE
Shade Line 0 or Shade Line 1 is loaded depending on which shade line is selected by loading register 0. See Paragraph 3.5.3.

3.8 LOADING GRAPH MEMORIES

The M7071 has two graph memories – Graph 0 and Graph 1. Each graph may plot up to 512-horizontal points. Each horizontal point may have only one value using up to 230 vertical points with the rectangular format (or 240 vertical points with the square format). The graph will begin entering data at X = 0 (040_8) unless a starting X-address is specified. (See Paragraph 3.13.)

The data to be entered is initially preceded by the letter B (102_8) for Graph 0, or J (112_8) for Graph 1. Each pair of data characters describes a Y-data point.

A Y-value is entered for each value of X, using two data characters, as described in Paragraph 3.6. As the X-address is incremented from 0 to 511, the values of Y can sequentially be loaded into memory. The X-register will automatically increment after each pair of Y-data characters are stored in memory, except when dual strip charts are enabled. (See Paragraph 3.14.) The letters B or J do not need to be repeated for each pair of Y-data characters. Table 3-8 may be used to determine the Y-data characters. The formation of graph data characters is illustrated in Table 3-11.

NOTE
For each value of Y, the lower data value must be transmitted first, then the upper data value.

Table 3-11 Loading Graph Data

	Graph 0	Graph 1
1st character:	B (102_8)	J (112_8)
2nd character:	variable (see below)	variable
3rd character:	variable (see below)	variable

Explanation of Second Character:

Binary Code
0 1 X X X X X

Bit	
6	The most significant bit is always 0.
5	Always 1.
0–4	The remaining bits form the low-order 5 bits of an 8-bit Y-value.

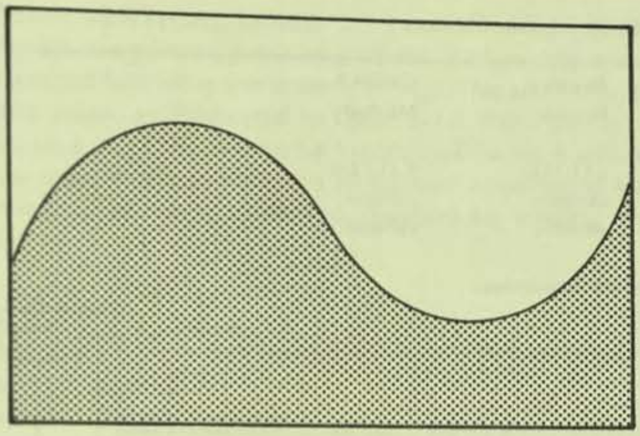
Explanation of Third Character:

Binary Code
0 1 X X X X X

Bit	
6	The most significant bit is always 0.
5	Always 1.
4	(unused)
3	(unused)
0–2	The remaining bits form the high-order 3 bits of the 8-bit Y-value.

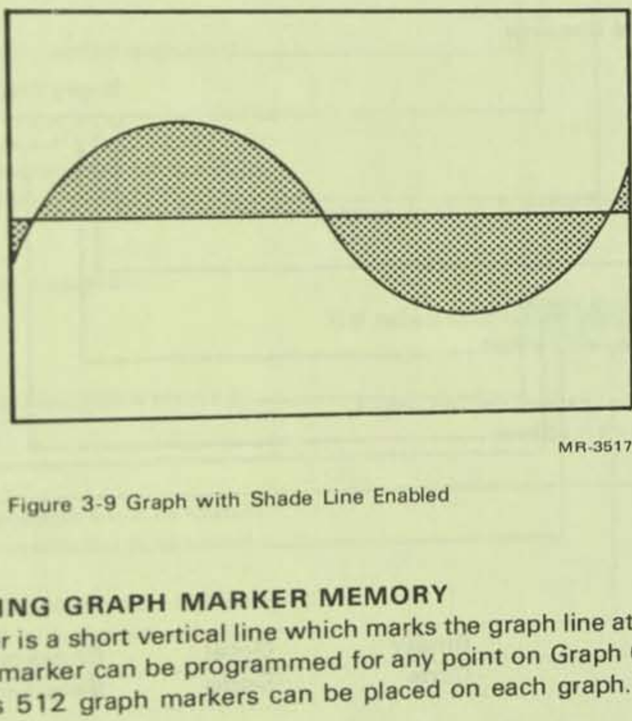
3.9 HISTOGRAM DATA

Data for a histogram (shaded graph) is entered by loading the appropriate graph memory (Paragraph 3.8). Histogram 0 and/or Histogram 1 is enabled by loading register 1 (Paragraph 3.5). Enabling a histogram will shade points between the graph envelope and the bottom of the graph field. Shading occurs from the graph data to graph line 0 (Figure 3-8). With a shade line enabled, the graph is shaded above and below this line, as in Figure 3-9.



MR-3516

Figure 3-8 Histogram without Shade Line Enabled



MR-3517

Figure 3-9 Graph with Shade Line Enabled

3.10 LOADING GRAPH MARKER MEMORY

A graph marker is a short vertical line which marks the graph line at a desired value of X. A graph marker can be programmed for any point on Graph 0 and on Graph 1. As many as 512 graph markers can be placed on each graph.

Loading graph marker memory is accomplished by sending pairs of data characters following the letter C (103_8) for Graph 0, or K (113_8) for Graph 1. Each pair of data characters represents the lower data value and the upper data value of an X-address, as illustrated in Table 3-12. Note that bit 4 of the third character determines whether the graph marker will be loaded or erased.

Table 3-12 Load Graph Marker Memory

	Graph 0 Marker	Graph 1 Marker
1st character:	C (103_8)	K (113_8)
2nd character:	variable	variable
3rd character:	variable	variable

Explanation of Second Character:

Binary Code
0 1 $\underbrace{XXXXX}$

Bit	
6	The most significant bit is always 0.
5	Always 1.
0–4	The remaining bits form the low-order 5 bits of a 9-bit X-address.

Explanation of Third Character:

Binary Code
0 1 X $\underbrace{XXXX}$

Bit	
6	The most significant bit is always 0.
5	Always 1.
4	If 1, causes graph marker to be loaded; if 0, causes marker to be erased.
0–3	The remaining bits form the high-order 4 bits of the 9-bit X-address.

Example:

Function	Binary Code	Octal Code	Character Sequence
Load graph 0 marker at location 100	1000011 0100100 0110011	103 044 063	C$3
Erase graph 0 marker at location 100	1000011 0100100 0100011	103 044 043	C$#

Table 3-8 may be used to determine the characters required to load or erase a specific graph marker. Once stored in memory, graph markers are enabled and disabled by loading register 1. (See Paragraph 3.5.4.)

3.11 DISPLAYING VERTICAL LINES

Vertical lines may be programmed for any of the 512 points along the X-axis. Vertical lines are loaded following the letter L (114_8). The second and third characters form an X-data value, as illustrated in Table 3-13. Note that bit 4 of the third character must equal a 1 for the line to be loaded; a 0 in bit 4 will erase that line. Table 3-8 may be used to determine the characters required to load or erase a specific line. Vertical lines are enabled and disabled by loading register 1. (See Paragraph 3.5.4.)

Table 3-13 Load Vertical Line Coordinate

1st character:	L (114_8)
2nd character:	variable (see below)
3rd character:	variable (see below)

Explanation of Second Character:

Binary Code
0 1 X X X X X

Bit	
6	The most significant bit is always 0.
5	Always 1.
0–4	The remaining bits from the low-order 5 bits of a 9-bit X-position.

Explanation of Third Character:

Binary Code
0 1 X X X X X

Bit	
6	The most significant bit is always 0.
5	Always 1.
4	If 1, causes vertical line to be loaded; if 0, causes vertical line to be erased.
0–3	The remaining bits form the high-order 4 bits of a 9-bit X-position.

3.12 DISPLAYING HORIZONTAL LINES

A horizontal line is loaded into memory by two data characters following the letter D (104_8). The second and third characters form a Y-data value, as illustrated in Table 3-14. Up to 230 horizontal lines may be displayed in the rectangular format; 240, in the square format. Note that bit 4 in the third character must equal a 1 to load a horizontal line; bit 4=0 will erase the line. Table 3-8 can be used to determine the characters required to load or erase a specific horizontal line. Horizontal lines are enabled and disabled by loading register 1. (See Paragraph 3.5.4.)

Table 3-14 Load Horizontal Line Coordinates

1st character: D (104_8)
2nd character: variable (see below)
3rd character: variable (see below)

Explanation of Second Character:

Binary Code
0 1 X X X X X

Bit	
6	The most significant bit is always 0.
5	Always 1.
0–4	The remaining bits form the low-order 5 bits of an 8-bit Y-position.

Explanation of Third Character:

Binary Code
0 1 X X X X X

Bit	
6	The most significant bit is always 0.
5	Always 1.
4	If 1, causes horizontal line to be loaded; if 0, causes horizontal line to be erased.
3	Unused
0–2	The remaining bits form the 3 high-order bits of the 8-bit Y-position.

3.13 LOAD THE STARTING X-COORDINATE

A starting X-coordinate may be loaded by two data characters following the letter H (110_8), as illustrated in Table 3-15. Any value of X, from 0 to 511, may be used. The data characters required for the desired X starting address can be determined from Table 3-8.

> **NOTE**
> **The lower data value of X must be transmitted first, then the upper data value.**

Table 3-15 Load Starting X Coordinate

1st character: H (110_8)
2nd character: variable (see below)
3rd character: variable (see below)

Explanation of Second Character

Binary Code

$0\ 1\ \underbrace{X\,X\,X\,X\,X}$

Bit	
6	The most significant bit is always 0.
5	Always 1.
0–4	The remaining bits form the low-order 5 bits of a 9-bit X-position.

Explanation of Third Character

Binary Code

$0\ 1\ X\ \underbrace{X\,X\,X\,X}$

Bit	
6	The most significant bit is always 0.
5	Always 1.
4	Unused
0–3	The remaining bits form the high-order 4 bits of a 9-bit X-position.

3.14 ENTERING STRIP CHART DATA

3.14.1 Single Strip Chart Data

Data for a single strip chart is entered by loading either Graph 0 or Graph 1 memory.

1. Enable the desired graph and single strip chart feature by loading register 0. (See Table 3-2.)

   Example:

   Enable Graph 0, Single Strip Chart, and Shade Line 0. Enter:

Character Sequence	Octal Sequence	Decimal Sequence
A#*	101_8 043_8 052_8	65_{10} 35_{10} 42_{10}

2. If desirable, load the starting X-coordinate at the right margin. Enter:

H??	110_8 077_8 077_8	72_{10} 63_{10} 63_{10}

3. Now, enter data into Graph 0 memory; type B plus any sequence of two data characters. If started at the right margin, this will cause the graph to move from right to left; if not, the graph will fill the screen first, then move data from right to left with each new data word.

B(data)(data)	102_8 (data)(data)	66_{10}(data)(data)

3.14.2 Dual Strip Chart Data

To set up the dual strip chart feature:

1. Enable both graphs and the dual strip chart feature; set bit 4 of the 3rd character in register 0. (See Table 3-2.)

   Example:

   Load Graph 0 and 1, dual strip chart, and Shade Line 0 and 1. Enter:

Character Sequence	Octal Sequence	Decimal Sequence
A'6	101_8 047_8 066_8	54_{10} 39_{10} 54_{10}

2. Load the starting X-coordinate at the right margin. Enter:

Character Sequence	Octal Sequence	Decimal Sequence
A'6	101_8 047_8 066_8	65_{10} 39_{10} 54_{10}
H??	110_8 077_8 077_8	72_{10} 63_{10} 63_{10}

3. Enter Graph 0 data; enter a B and two data characters. (The data is entered, but the graph will not move at this time.)

   B(data)(data) 102_8(data)(data) 66_{10}(data)(data)

4. Enter Graph 1 data; enter a J plus two data characters. The graph will now display and shift both Graph 0 and Graph 1 data points one increment to the left.

   J(data)(data) 112_8(data)(data) 74_{10}(data)(data)

NOTES

1. **Load starting X-coordinate, desired graph markers, and vertical lines before enabling either single- or dual-strip chart mode. The exact position of these points may vary once a strip chart is enabled.**

2. **Any graph markers and vertical lines enabled will wrap around as the strip chart moves.**

3. **When the strip chart mode is disabled, any displayed graphics shifts. The X-coordinate that moved during strip chart mode returns to its normal location.**

4. **When in dual-strip chart mode, the last available graph position (X=511) is not displayed. The switching action between Graph 0 and Graph 1 in that position is eliminated from the display.**

5. **A graph drawing programmer's card is included in Appendix C.**

CHAPTER 4

INSTALLATION, INTERFACE INFORMATION, AND SPECIFICATIONS

4.1 UNPACKING

The VT105 is packaged within a reinforced carton containing the following items.

Video monitor
Detached keyboard
Power cord
Set-up Label
User guide

Instructions are printed on the carton for unpacking the unit.

4.2 SITE CONSIDERATIONS

The VT105 consists of a video monitor and detachable keyboard that may be placed on a desk or table top. The dimensions of the VT105 are shown in Figure 4-1.

Specifications for operating environment are provided in Paragraph 4.8.

4.3 INSTALLATION

To install the VT105 perform the following steps.

1. Remove the VT105 from the shipping carton and place it in the desired work area.

2. Place the keyboard in front of the terminal and plug the keyboard coiled cord into the keyboard receptacle located on the rear of the terminal (Figure 4-2).

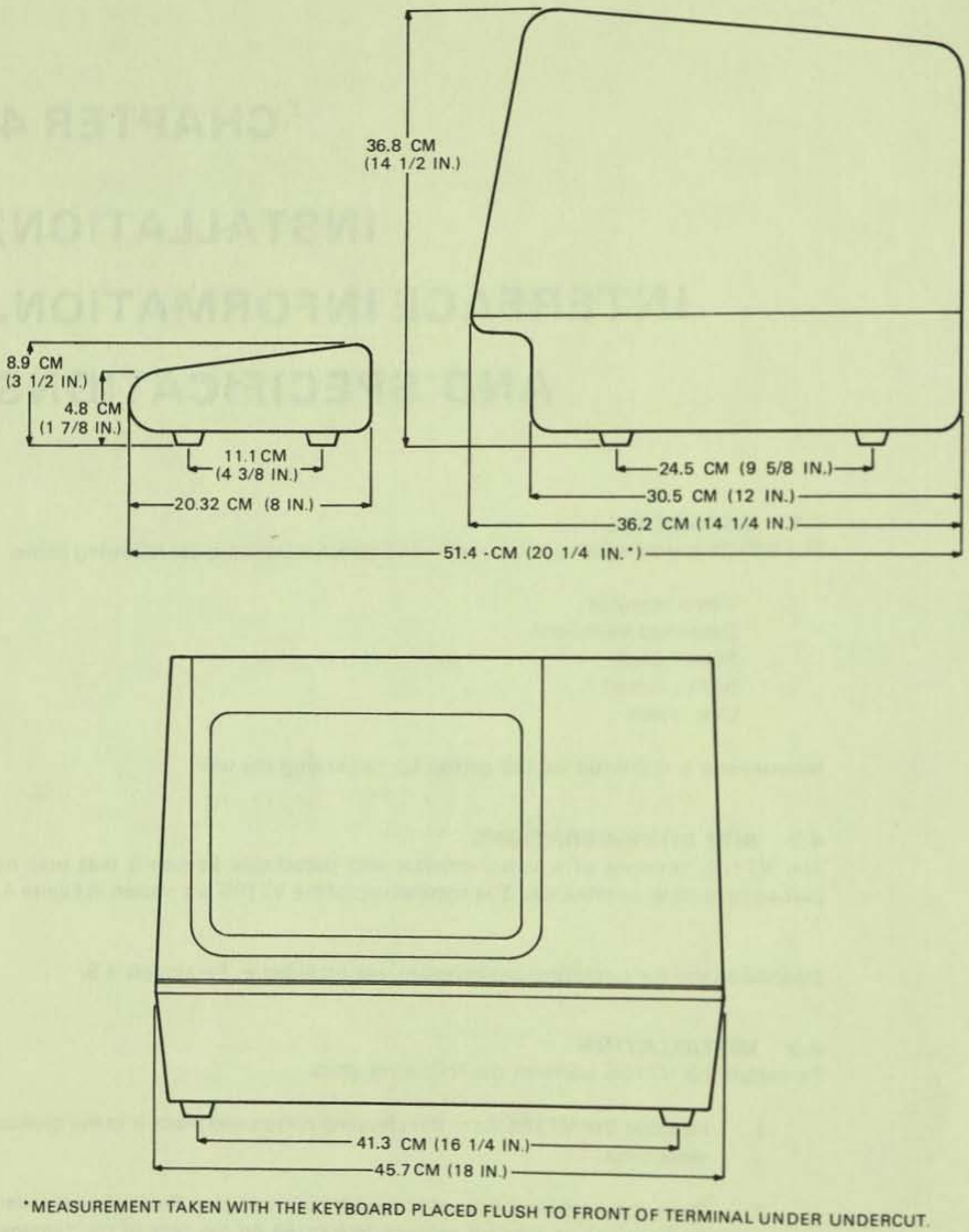


Figure 4-1 VT105 Terminal Dimensions

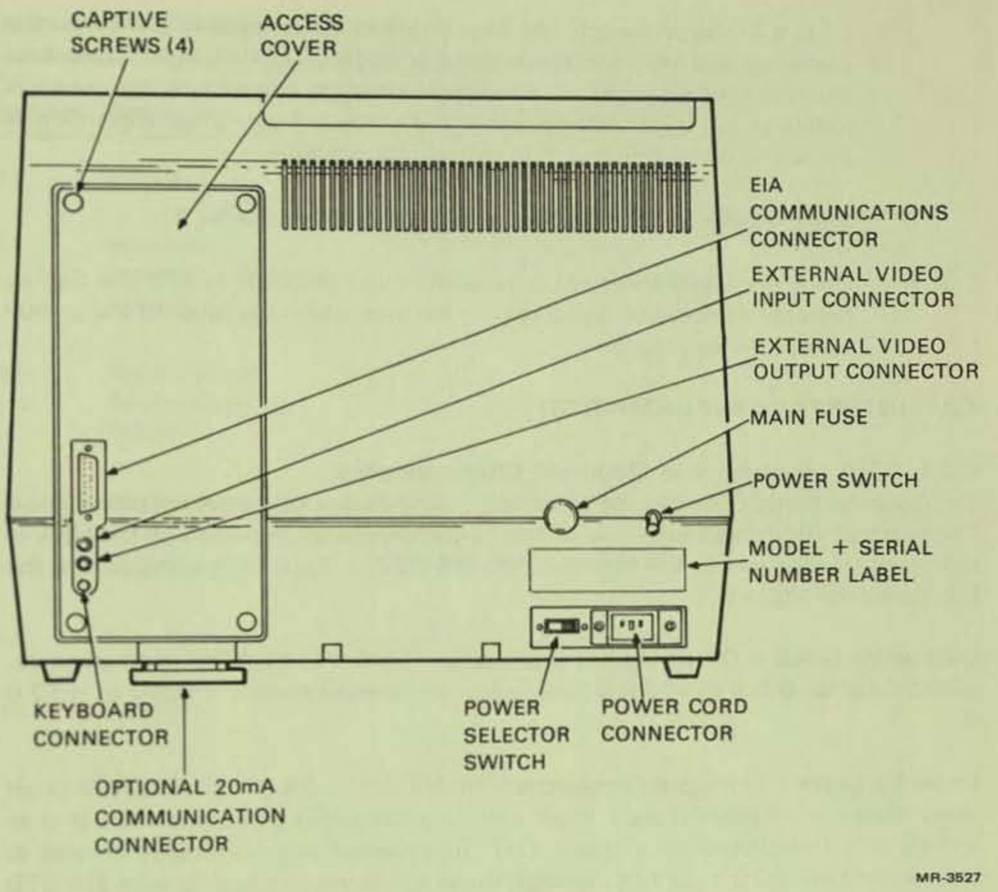


Figure 4-2 VT105 Rear View

3. Verify that the power selector switch shows the correct wall outlet voltage (115 V is standard in the U.S.).

Switch Position	Voltage Range
115	90 – 128 Vac RMS
230	180 – 256 Vac RMS

4. Verify that the power switch is OFF.

Position	AC Power
Up	ON
Down	OFF

5. Connect the communications cable to the appropriate communications receptacle. Refer to Paragraph 4.4 for interface information.

6. Connect the power cord to the power cord receptacle on the rear of the terminal and plug the other end of the power cord into a nearby wall outlet.

7. Turn the power switch on. The terminal automatically performs the power-up self-test and either the ON LINE or LOCAL light located on the keyboard is turned on. After approximately one minute, the cursor is visible in the upper-left corner of the screen. If any other indication is shown, refer to Chapter 1 to isolate the problem.

8. Set the desired Set-up features as outlined in Chapter 1.

9. Once the installation setup procedure is complete, record the Set-up features selected on the Set-up label and attach the label to the underside of the keyboard.

4.4 INTERFACE INFORMATION

4.4.1 EIA Interface and Electrical Characteristics

The basic terminal operates on full duplex, asynchronous communication lines. The terminal interfaces to the line with a 25-pin connector mounted on the back of the terminal that meets EIA specification RS-232-C. Table 4-1 summarizes the EIA connector signals.

Output Voltages – On all signals designated "from VT105," the mark or unasserted state is –6.0 V to –12.0 V; the space or asserted state is +6.0 V to +12.0 V.

Input Voltages – On signals designated "to VT105," –25.0 V to +0.75 V or an open circuit is interpreted as a mark, ON, or unasserted state, and +25.0 V to +2.25 V is interpreted as a space, OFF, or asserted state. Voltages greater in magnitude than ±25 V are not allowed. These levels are compatible with EIA STD RS-232-C and CCITT Recommendation V.28.

Table 4-1 EIA RS-232-C Connector Signals

Pin Number	Description	Source/ Destination	Remarks
1	Protective ground	Ground	Do not use
2	Transmitted data	From VT105	Mark state when not sending.
3	Received data	To VT105	
4	Request to send	From VT105	Asserted at all times
5	Clear to send	To VT105	Ignored
6	Data set ready	To VT105	Ignored
7	Signal ground	Ground	Terminal chassis ground (common return)
8	Carrier detect	To VT105	Ignored
9	(not used)		

Table 4-1 EIA RS-232-C Connector Signals (Cont)

Pin Number	Description	Source/ Destination	Remarks
10	(not used)		
11	Speed select (secondary request to send)	From VT105	Unasserted
12	Speed indicator (secondary carrier detect)	To VT105	Ignored
13	(not used)		
14	(not used)		
15	Transmit clock	To VT105	Ignored
16	(not used)		
17	Receive clock	To VT105	Ignored
18	(not used)		
19	Speed select (Same as pin 11)	From VT105	Unasserted
20	Data terminal ready	From VT105	Asserted at all times except: 1. terminal is not powered up. 2. terminal is in LOCAL mode. 3. during the 3.5s interval following a SHIFT-BREAK
21	(not used)		
22	Ring indicator	To VT105	Ignored
23	Speed select (Same as pin 11)	From VT105	Unasserted
24	(not used)		
25	(not used)		

4.4.2 Optional 20 mA Current Loop Interface

In most current loop applications, the VT105 is connected in a passive configuration – that is, current is supplied to the VT105. In this mode, the transmitter and receiver are both passive, both optically isolated, and the transmitter goes to the mark state when power is turned off.

Conversion from active to passive (or vice versa) requires moving a slide switch.

In active mode, either the transmitter or the receiver or both may be connected so that the VT105 is the source of the 20 mA of current. In active mode, isolation is not present, and the transmitter goes to the space state when power is turned off.

Figure 4-3 shows the 20 mA current loop interface connector mounted to the access cover and the individual pin assignment.

20 mA Electrical Characteristics – The electrical characteristics of the 20 mA current loop interface are shown below.

	Min	Max
Transmitter		
Open circuit voltage	5.0 V	50 V
Voltage drop marking	–	4.0 V
Spacing current	–	2.0 mA
Marking current	20 mA	50 mA
Receiver		
Voltage drop marking	–	2.5 V
Spacing current	–	3.0 mA
Marking current	15 mA	50 mA

In addition to the above specifications for passive operation, active mode places the transmitter or receiver in series with a source of 17 V ±5 percent and 660 ohms.

4.5 EXTERNAL VIDEO CONNECTION

The VT105 may be easily interfaced to external video devices. The video devices may act either as a slave to the VT105 when connected to the composite video output or provide synchronized video data to the VT105. The external video connectors are the two female BNC connectors located on the back of the terminal just below the EIA connector. The upper connector, J8, is the video input; the lower connector, J9, is the video output.

4.5.1 Composite Video Output (J9)

The composite video output provides RS170-like output (see note) generated by combining the video signal with a composite sync signal. This output contains all video data appearing on the VT105 screen *except* that video which comes from J8. The output has the following nominal characteristics. See Figure 4-4.

1. Output impedance = 75 ohms, dc-coupled
2. Sync level = 0 V
3. Black level = approximately 0.3 V with a 75 ohms load
4. White level = approximately 1.0 V with a 75 ohm load

5. The composite sync waveform conforms to EIA RS170 standards. The vertical interval is composed of six equalizing pulses, six vertical sync pulses, and six more equalizing pulses. The timing is as follows.

Equalizing pulse width	=	2.33 μs$\pm$50 ns
Vertical sync width	=	27.28 μs$\pm$200 ns
Horizontal sync width	=	4.71 μs$\pm$50 ns
Horizontal blank time	=	11.84 μs$\pm$50 ns (80 column mode)
	=	12.34 μs$\pm$50 ns (132 column mode)
Front Porch	=	1.54 μs$\pm$50 ns
Horizontal period	=	63.56 μs$\pm$50 ns

NOTE
The use of decoupling is not in strict agreement with RS170. To agree with RS170 the output load requires a 10 μF capacitor in series with the output. This presents no problem with most monitors as they are ac-coupled.

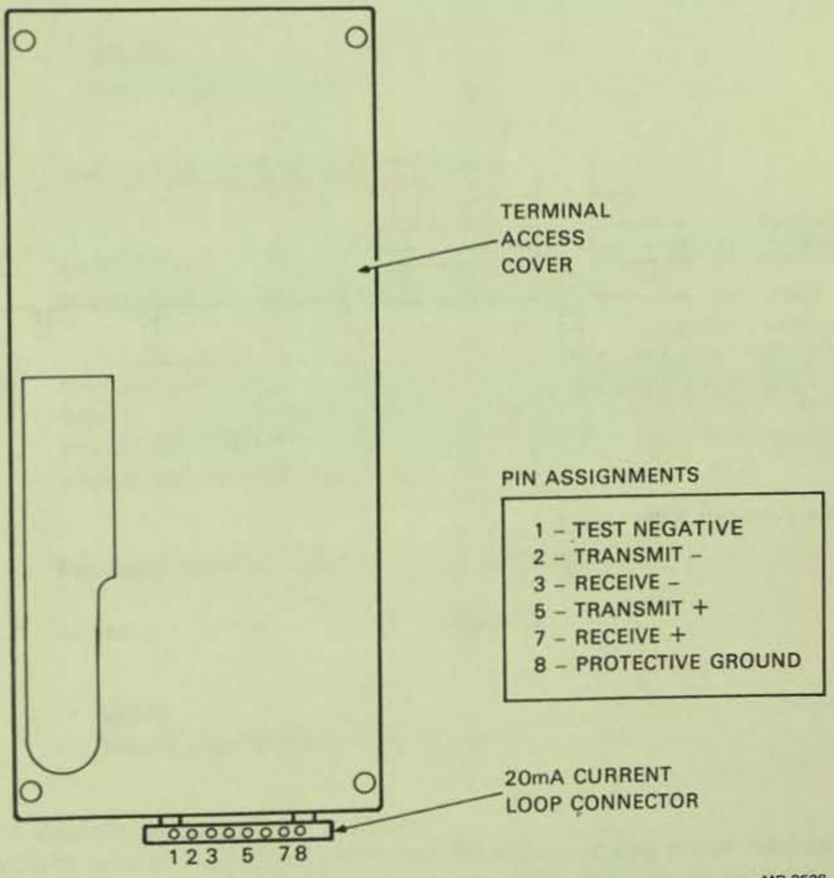


Figure 4-3 20 mA Current Loop Connection

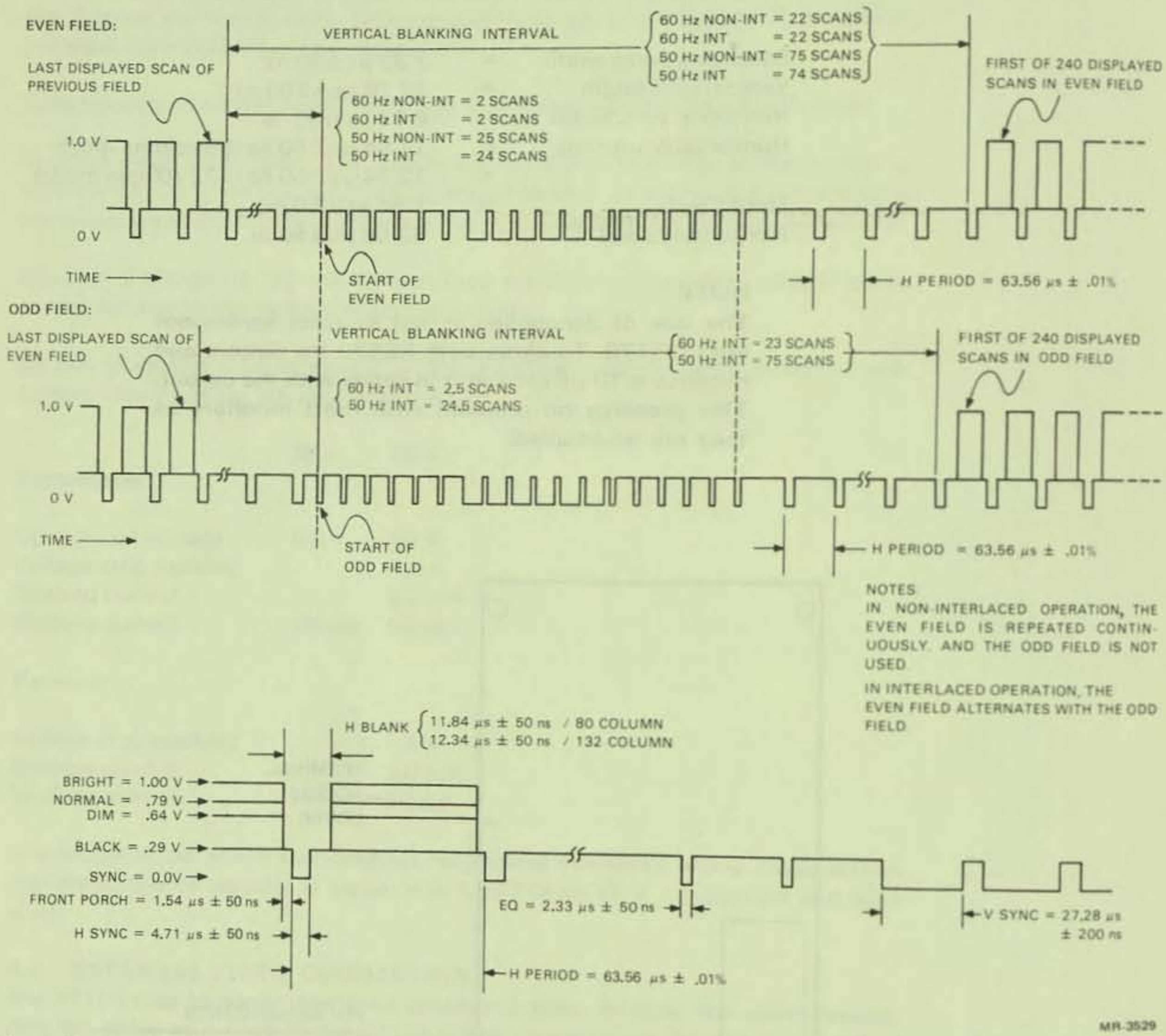


Figure 4-4 Composite Video Output

4.5.2 Video Input (J8)

An analog signal applied to the video input is ORed with the internal video signal. The beam intensity at any point on the screen corresponds to the intensity of the signal that would make the beam brighter. A video signal on this input affects only the internal screen and does not appear on the composite video output, J9. This input has the following nominal characteristics.

Input impedance	= 75 ohms, dc-coupled
Black level	= 0 V
White level	= 1.0 V
Maximum continuous input	= ±2.0 V

NOTE
The external video source must be synchronized to the VT105 by referencing the composite sync on the composite video output.

4.6 WAVEFORM GENERATOR INSTALLATION

The waveform generator module (M7071) provides the graph drawing capabilities of the VT105. This module is installed from the rear of the terminal as follows.

NOTE
Power must be OFF.

1. Remove the terminal access cover.

2. Slide the module into the card cage assembly and plug it into the jack provided on the 54-13384 expander board.

3. Connect one end of an IC connector cable (70-08612) to XE90 on the M7071 module. Connect the other end to J2 on the terminal controller board, as shown in Figure 4-5. The red line on the cable must be as shown and connects as follows.

Terminal Controller Board	M7071 Module
J2, pin 1	XE88, pin 1

NOTE
Pins 9 and 10 of J2 are left open.

4.6.1 Waveform Generator Checkout

The M7071 module has an interactive test feature to display known test patterns on the video screen by typing keys on the keyboard. Refer to Appendix A to run this test procedure in entirety.

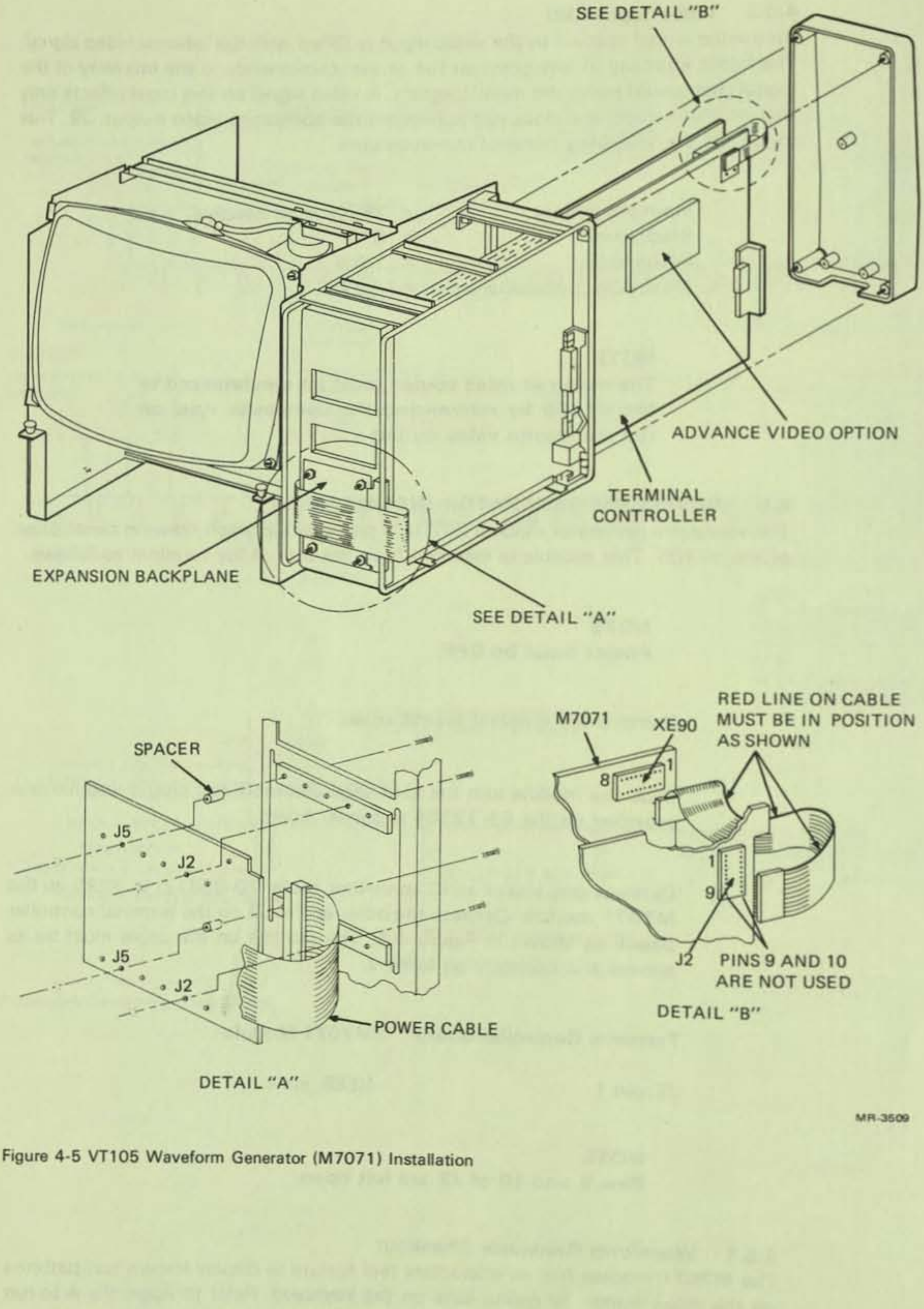


Figure 4-5 VT105 Waveform Generator (M7071) Installation

4.7 CUSTOMER EQUIPMENT CARE

The keyboard keys are the only moving parts of the terminal and require no preventive maintenance by the owner. Rubbing the keys with a dry or barely moist cloth should suffice to clean them. Do not remove the keycaps to clean them more thoroughly; damage may result to the switch contacts if they are replaced incorrectly.

The terminal cabinet surfaces may be cleaned with soap and water or any mild detergent. Cleaners with solvents should not be used. Avoid using excessive water to clean the terminal.

Keep the ventilation slots clear so as not to cause the terminal to overheat.

4.8 VT105 SPECIFICATIONS

Dimensions

Monitor	Height: 36.83 cm (14.5 in) Width: 45.72 cm (18 in) Depth: 36.20 cm (14.25 in)
Keyboard	Height: 8.89 cm (3.5 in) Width: 45.72 cm (18 in) Depth: 20.32 cm (8 in) Minimum table depth: 51.4 cm (20.25 in)

Weight

Monitor	13.6 kg (30 lb)
Keyboard	2.0 kg (4.5 lb)
Shipping Weight	19 kg (42 lb)

Environment

Operating	Temperature: 10° to 40° C (50° to 104° F) Relative humidity: 10% to 90% Maximum wet bulb: 28° C (82° F) Minimum dew point: 2° C (36° F) Altitude: 2.4 km (8,000 ft)
Non-Operating	Temperature: -40° to 66° C (40° to 151° F) Relative humidity: 0 to 95% Altitude: 9.1 km (30,000 ft)

Power

Line Voltage	90–128 V RMS single phase, 2 wire 180–256 V RMS single phase, 2 wire (switch-selectable)
Line Frequency	47–63 Hz
Current	3.0 A RMS maximum at 115 V RMS 1.5 A RMS maximum at 230 V RMS

Power (Cont)

Input Power	250 VA apparent, 150 W maximum
Current limiting	3 A normal blow fuse
Power cord	detachable, 3 prong, 1.9 m (6 ft) 120 Vac power cord (DEC 17-00083-09) 240 Vac power cord (DEC 17-00083-10)

Display

CRT	305 mm (12 in) diagonal measure, P4 phosphor
Format	24 lines × 80 characters or 14 lines × 132 characters (selectable)

Graph Drawing

No. of fields	Two, noninterlaced graph drawing fields.
Rectangular field	20.3 × 10.94 cm (8 × 4.375 in) horiz × vertical
Square field	16.5 × 11.56 cm (6.5 × 4.625 in) horiz × vertical
Graph resolution	rectangular format: 230 dots × 512 dots square format: 240 dots × 512 dots
No. of graphs, shaded graphs, or stripcharts	Two single-valued functions of X, individually controlled
Strip charts	Single or dual strip charts; right data entry; right-to-left movement.
Shade lines	Two movable shade lines; one per graph.
Graph markers	512 per graph; total of 1024, individually controlled.
Vertical lines	512; individually controlled.
Horizontal lines	230 (or 240) individually controlled in rectangular (or square) format.
Special features	Individual blanking and unblanking of separate graphs; overlapping areas of shaded graphs are discernible.
Test feature	Operator controlled by "key-in" instructions. Test patterns displayed.

Alphanumerics

Character	7 × 9 dot matrix with descenders for lowercase

Alphanumerics (Cont)

Character size	3.35 mm × 2.0 mm (0.132 inch × 0.078 inch) in 80 column mode 3.35 mm × 1.3 mm (0.132 inch × 0.051 inch) in 132 column mode
Character Set	96 character displayable ASCII subset (upper and lowercase, numeric and punctuation).
Cursor type	Keyboard selectable, blinking block character or blinking underline.

Keyboard

General	83-key detachable unit with a 1.9 m (6 ft) coiled cord attached.
Key Layout	65-key arrangement and sculpturing similar to standard typewriter keyboard with an 18-key auxiliary keypad.
Auxiliary Keyboard	18-key numeric pad with period, comma, minus, enter, and four general purpose function keys.
Visual Indicators	Seven LEDs: three LEDs are dedicated to ON LINE, LOCAL and KBD LOCKED; four LEDs are user programmable.
Audible Signals	Key-click: sounds when keys are typed. Bell: sounds upon receipt of BEL code; and sounds eight characters from right margin. Multiple bell: sounds upon detection of error in Set-up save or recall operation.

Communication

Type	EIA
Speeds	Full duplex: 50, 75, 110 (two stop bits), 134.5, 150, 200, 300, 600, 1200, 1800, 2000, 2400, 3600, 4800, 9600, 19,200
Code	ASCII
Character Format	Asynchronous 7- or 8-bit characters; keyboard selectable. (Note: if 8-bit characters are selected, the 8th bit is always a space.)
Parity	Even, odd, or none; keyboard selectable.
Synchronization	Keyboard selectable via automatic generation of XON and XOFF control codes.

CHAPTER 5

OPTIONS

The VT105 has available options that are designed to enhance the basic terminal and allow its use in a wide range of applications. The options currently available are:

VT1XX-AA 20 mA Current Loop Option
VT1XX-AB Advanced Video Option

Contact the nearest DIGITAL Sales Office for further information and instructions on how to order the options.

5.1 20 mA CURRENT LOOP OPTION – VT1XX-AA

The 20 mA Current Loop option allows the VT105 to connect directly to the host computer over a short distance without using a modem. The specifications for the 20 mA Current Loop option are located in Chapter 4.

5.1.1 20 mA Current Loop Option Installation

Use the following procedure to install the 20 mA Current Loop option:

1. Remove power from the terminal by disconnecting the ac plug.
2. Unplug the keyboard.
3. Unplug any connectors from the composite video input/output jacks.
4. Disconnect the communications cable.
5. With a blade-type screwdriver loosen the four captive screws holding the access cover; then remove the cover.
6. Set the TRANS switch to the NORMAL position (Figure 5-1). (If the terminal must provide current on the receiver line, set the switch to the ACT position.)

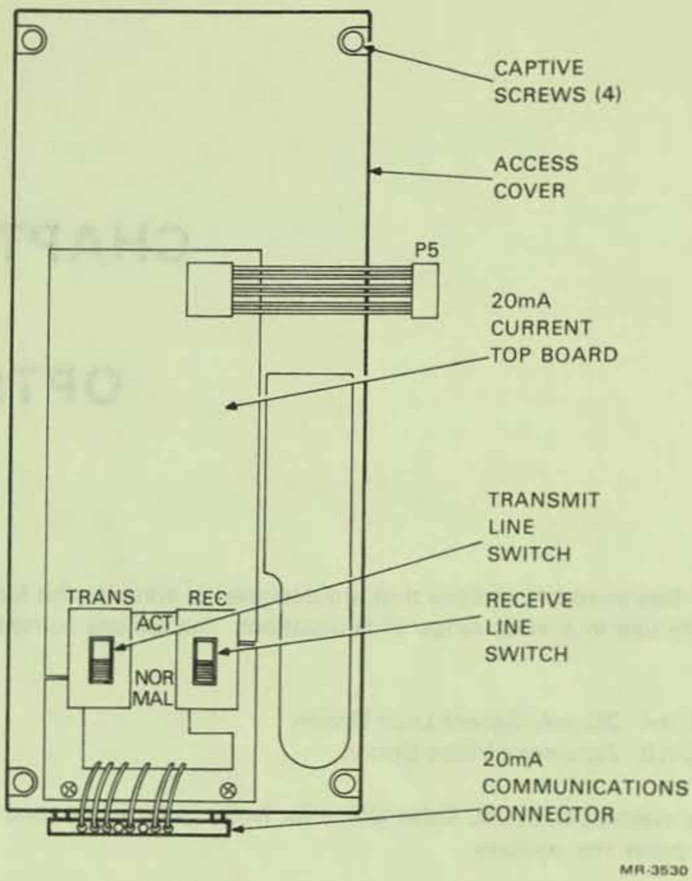


Figure 5-1 20 mA Current Loop Option

7. Set the REC switch to the NORMAL position (Figure 5-1). (If the terminal must provide current on the receive line, set the switch to the ACT position.)

8. Connect P5 to J5 on the terminal controller board (Figure 5-2).

9. Perform the 20 mA Current Loop option checkout.

5.1.2 20 mA Current Loop Option Checkout

The VT105 contains an internal test called the data loopback test. In the data loopback test the VT105 transmit and receive lines are connected to each other via a special external connector. A predefined set of characters are then transmitted. The terminal receives the characters and compares them to the characters transmitted. If the characters do not match, an error is then flagged.

Use the following procedure to check out the operation of the 20 mA current loop option.

1. Disconnect the terminal from the communications line.

2. Remove the terminal access cover containing the 20 mA current loopback board (Figure 5-2); place both of the switches in the NORMAL position. Reinstall the access cover.

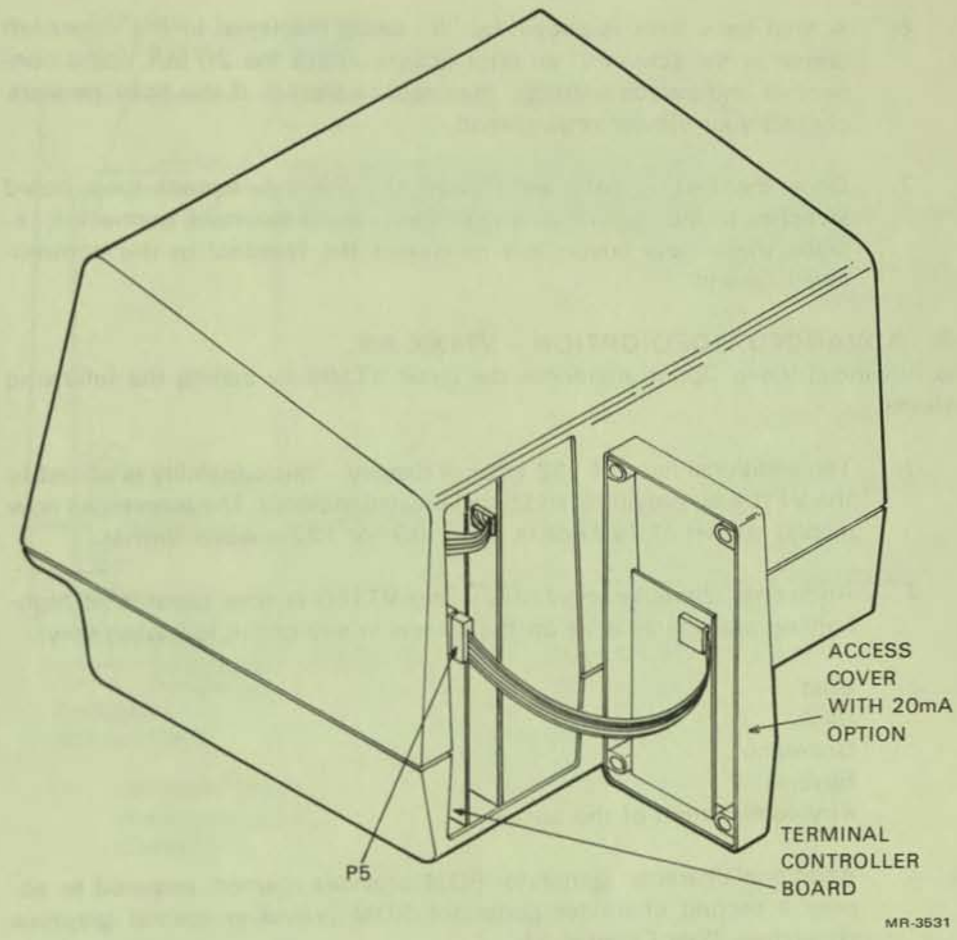


Figure 5-2 Terminal Access Cover with 20 mA Option Installed

3. Connect the 20 mA loopback connector (PN 70-15503-00) to the Mate-N-Lok connector mounted to the bottom of the access cover.

4. Place the terminal in ANSI mode (Set-up B).

5. Type the following sequence to perform the data loopback test.

   ESC [2;2y

   When the test is performed, the screen clears and the message WAIT is displayed in the upper-left corner of the screen. The entire test takes approximately 6 seconds to run.

   NOTE
   The data loopback test can be repeated indefinitely by typing: ESC[2;10y.

   This test ends only when an error is detected or the terminal power is turned off.

6. A loop back error is shown by "8" being displayed in the upper-left corner of the screen. If an error occurs, check the 20 mA board connectors and switch settings, then repeat step 5. If the error persists, contact your service organization.

7. Once the test is complete, return the 20 mA current loop board switches to the original positions, remove the loopback connector, replace the access cover, and reconnect the terminal to the communications line.

5.2 ADVANCED VIDEO OPTION – VT1XX-AB

The Advanced Video Option enhances the basic VT105 by adding the following features.

1. Ten additional lines of 132 column display – this capability is added to the VT105 by providing additional display memory. The screen can now display a total of 24 lines in either 80- or 132-column format.

2. Additional character attributes – the VT105 is now capable of highlighting any character(s) on the screen in any of the following ways:

   Bold
   Blink
   Underline
   Reverse
   Any combination of the above

3. Additional character generator ROM provides memory required to access a second character generator ROM providing special graphics characters. (See Chapter 2.)

5.2.1 Advanced Video Option Installation

Use the following procedure to install the Advanced Video Option.

1. Remove power from the terminal by disconnecting the ac plug.

2. Unplug the keyboard.

3. Unplug any connectors from the composite video input/output jacks.

4. Disconnect the communications cable.

5. With a blade-type screwdriver, loosen the four captive screws holding the access cover (Figure 5-3).

6. If the 20 mA current loop option is installed, gently pull the access cover away from the terminal about 5 cm (2 in); then reach in and disconnect J5 from the terminal controller board. The terminal controller board is located in the leftmost slot in the card cage (viewed from the rear).

7. Remove the access cover.

8. Remove graphic cable from J2 on the terminal controller board; remove other cables if present.

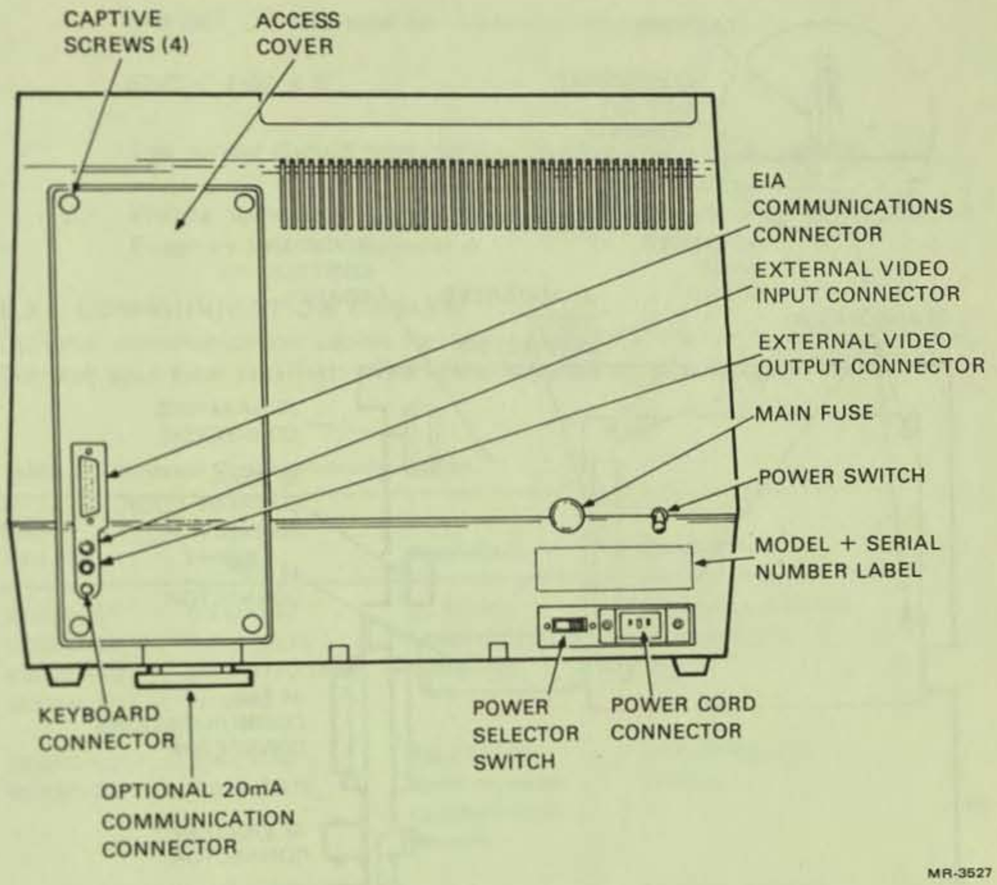


Figure 5-3 VT105 Rear View

9. Remove the terminal controller board by gently but firmly pulling the board straight out.

10. Place the terminal controller board on a flat surface with the component side up.

11. Locate the four mounting holes drilled in the terminal controller board shown in Figure 5-4. Mount a standoff in each hole.

12. Grasp the advanced video board by the edges and carefully align the connector pins with the connector on the terminal controller board. Gently but firmly mount the Advanced Video board onto the terminal controller board.

13. Reinstall the terminal controller board. The terminal controller board must be inserted into the leftmost slot in the card cage.

14. Reinstall all cables.

15. Reinstall the access cover.

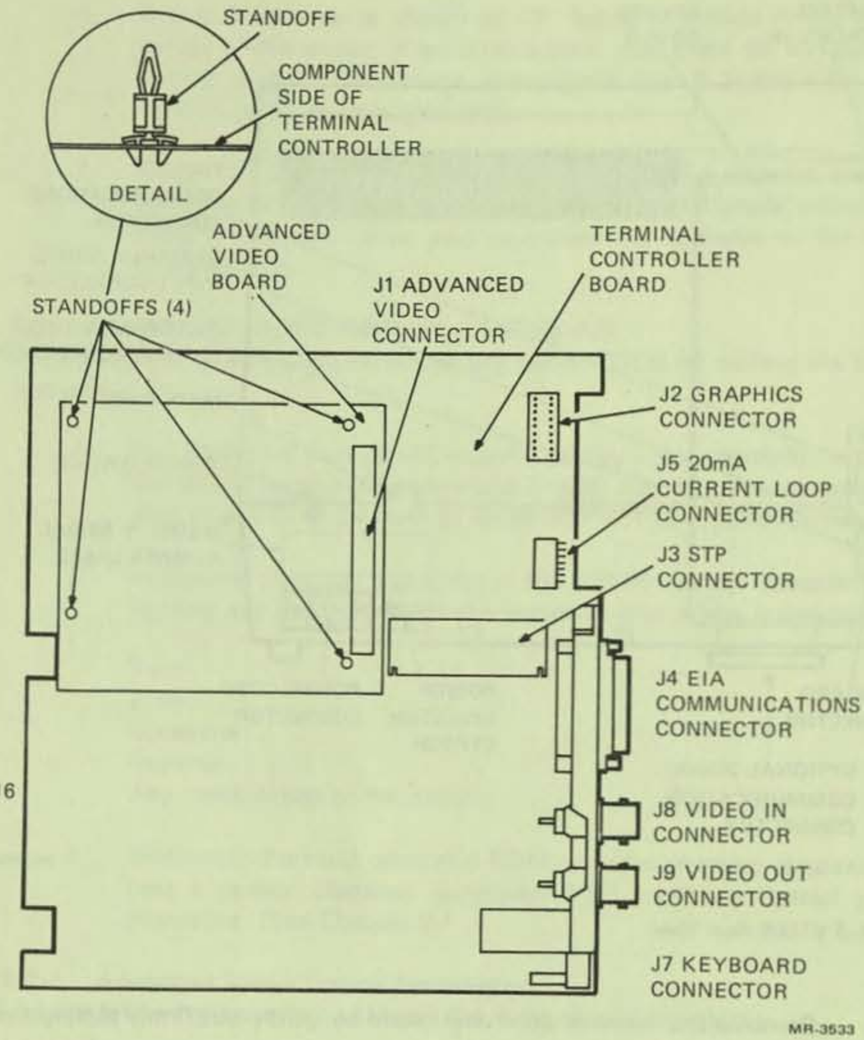


Figure 5-4 Advanced Video Option Installation

5.2.2 Advanced Video Option Checkout

Use the following procedure to check out the operation of the Advanced Video option.

1. Turn the terminal power on and verify that no error is detected during the power-up self-test.

2. Press the SET-UP key. The words "SET-UP A" should blink in boldface, the words "TO EXIT PRESS SET-UP" should be underlined, and the ruler should contain alternating normal and reverse video fields.

3. Place the terminal in the 132 column mode and then in the LOCAL mode.

4. Exit SET-UP and type the following sequence.

   ESC < ESC # 8

   The screen should now display 24 lines × 132 columns.

5. Ensure waveform generator is operational. Perform the Interactive Graphics test in Appendix A.

5.3 COMMUNICATION CABLES

Optional communication cables for use with the VT105 are listed in Table 5-1. Contact your local DIGITAL sales office for ordering information.

Table 5-1 Optional Communications Cables

Cable Part No.	Length	Connectors	Purpose
BC03M-01 BC03M-25 BC03M-AO BC03M-XX*	0.3 m (1 ft) 7.6 m (25 ft) 30.5 m (100 ft) variable	2 - RS-232 female Cinch connectors	Null modem; connects terminal to a line unit.
BC05D-10** BC05D-25**	3 m (10 ft) 7.6 m (25 ft)	RS-232 male Cinch connector to female Cinch connector	Connects terminal to a modem
BC05F-9 BC05F-15† BC05F-XX*	2.7 m (9 ft) 4.5 m (15 ft) variable	2 - Mate-N-Lok male connectors	Connects terminal with a 20 mA option installed to a line unit.

*The -XX indicates that other lengths are available but do not come preassembled.
**When using a DF01-A acoustic coupler, pin 23 of this cable must be disconnected.
†A BC05F-15 cable is shipped with the VT1XX-AA 20 mA current loop option.

APPENDIX A

VT105 INTERACTIVE GRAPHIC TEST PROCEDURE

A.1 INTRODUCTION

The interactive test feature of the VT105 is a series of test patterns that can be displayed by the user or Field Service engineer. One or all the patterns may be selected to be displayed; they are not dependent on each other.

The tests described in this procedure are run in the rectangular format.

NOTES

1. **The tests may also be run in the square format; however, the test patterns are slightly different.**

2. **Do not type the SPACE BAR unless the word SPACE is spelled out.**

3. **Remember to use the SHIFT key for uppercase symbols; the CAPS LOCK key is only used for uppercase letters.**

4. **If at any time the wrong character is entered, initialize the registers and memories by typing the following sequence.**

   A SPACE SPACE
   I 0
   I SPACE "

   Then reenter the test data.

A.2 SET-UP INTERACTIVE GRAPHIC TEST

Function	Type
Enter VT105 LOCAL mode.	
Set terminal to auto-repeat mode (Set-up B).	
Enter graphic mode:	ESC 1
Enable rectangular format:	I SPACE SPACE
Enable test:	I SPACE "

A.3 TESTING PROCEDURE

Enable the rectangular format of the interactive test.

A.3.1 Test Graph 0, Histogram 0, and Graph 0 Markers

Function	Type	Figure
Enable Graph 0	A#	A-1
Enable Histogram 0	A)	A-2
Enable Graph 0 markers	I$	A-3
Disable Graph 0 markers	I SPACE	
Disable Histogram 0 and Graph 0	A SPACE	

A.3.2 Test Graph 1, Histogram 1, and Graph 1 Markers

Function	Type	Figure
Enable Graph 1	A%	A-1
Enable Histogram 1	A1	A-2
Enable Graph 1 markers	I(	A-3
Disable Graph 1 markers	I SPACE	
Disable Histogram 1 and Graph 1	A SPACE	

A.3.3 Test Horizontal Lines

Function	Type	Figure
Enable display	A!	
Enable horizontal lines	I!	A-4
Disable horizontal lines	I SPACE	

A.3.4 Test Vertical Lines

Function	Type	Figure
Enable display	A!	
Enable vertical lines	I"	A-5
Disable vertical lines	I SPACE	

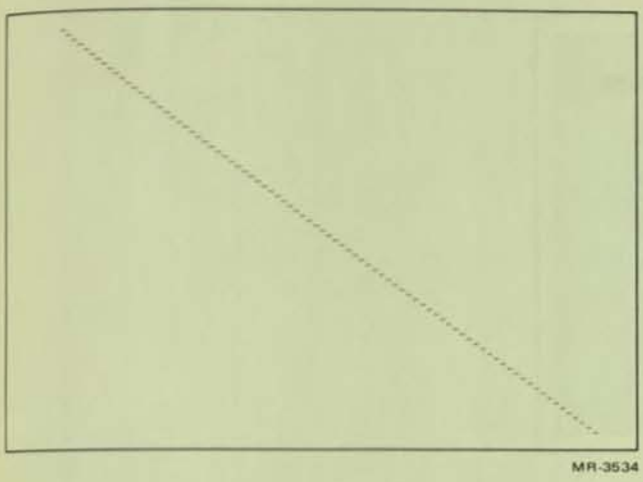


Figure A-1 Graph Test Pattern

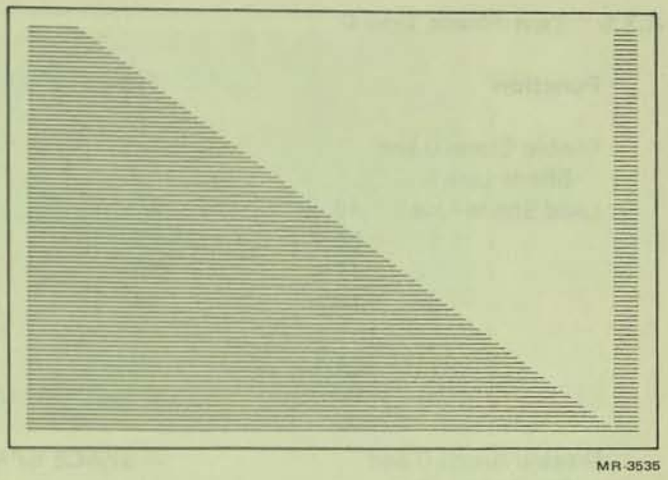


Figure A-2 Histogram Test Pattern

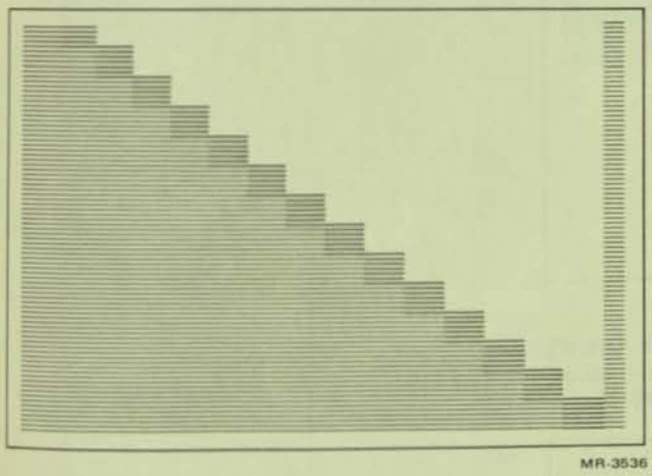


Figure A-3 Graph Marker Test Pattern

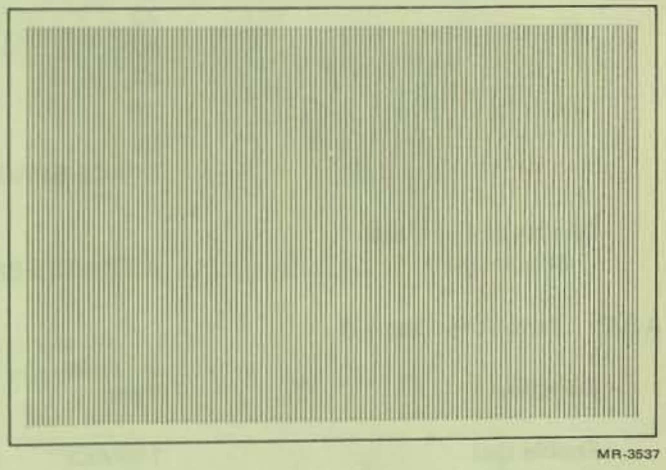


Figure A-4 Horizontal Line Test Pattern
(will look like vertical lines)

A.3.5 Test Shade Line 0

Function		Type	Figure
Enable Graph 0 and Shade Line 0		A#"	A-6
Load Shade Line:	49	@11	
	82	22	
	115	33	
	148	44	
	181	55	
	214	66	
	0	SPACE SPACE	
Disable Graph 0 and Shade Line 0		A SPACE SPACE	

A.3.6 Test Shade Line 1

Function	Type	Figure
Enable Graph 1 and Shade Line 1	A%%	A-6
Load shade lines	@11	
	22	
	33	
	44	
	55	
	66	
	SPACE SPACE	
Disable Graph 1 and Shade Line 1	A SPACE SPACE	

A.3.7 Test Strip Chart 0

Function	Type	Figure
Enable test	I SPACE"	
Enable Graph 0, Histogram 0, and Strip chart 0	A+(	A-2
Load X at right margin	H??	
Enable Load Graph 0	B	A-7

NOTE
Any sequence of two numbers will cause graph to move; i.e., the SPACE BAR enters a 0 as in Figure A-7, 11 enters data at line 49, 22 enters data at line 82, etc. Hold number key down if in auto-repeat mode.

Disable Graph 0, Histogram 0 and Strip chart 0	A SPACE SPACE

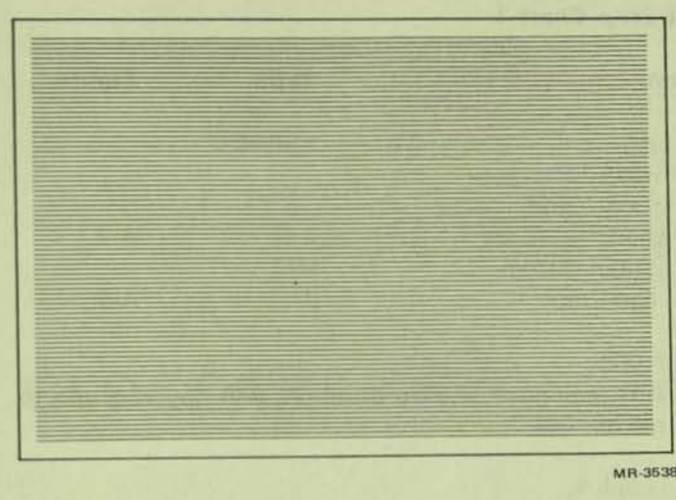


Figure A-5 Vertical Line Test Pattern
(will look like horizontal lines)

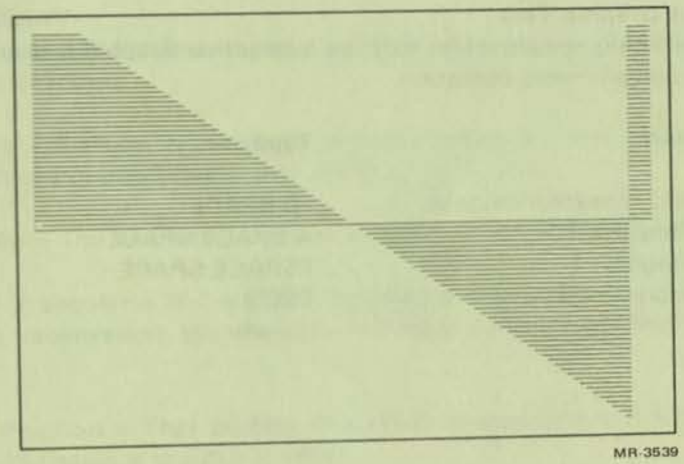


Figure A-6 Shade Line Test Pattern
(shade line will shift upward)

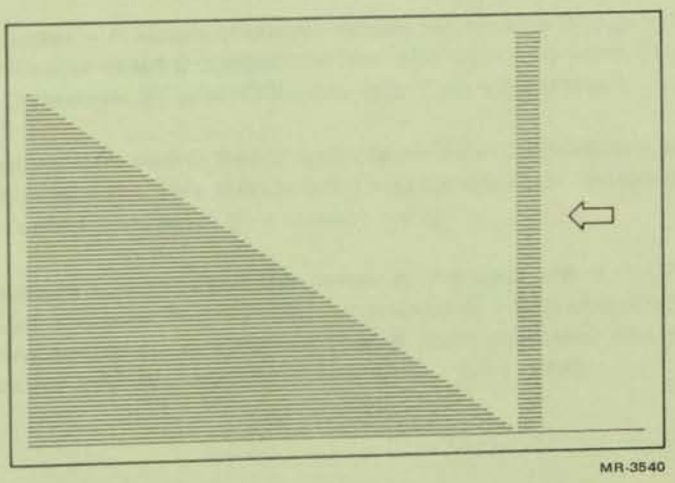


Figure A-7 Strip Chart Test Pattern
(Graph should shift right to left)

A.3.8 Test Strip Chart 1

Function	Type	Figure
Enable test	I SPACE"	
Enable Graph 1, Histogram 1 and Strip chart 1	A5(	A-2
Load X at right margin	H??	
Enable Load Graph 1	J	A-7

NOTE
Any sequence of two numbers will cause graph to move.

Disable Graph 1, Histogram 1 and Strip chart 1	A SPACE SPACE	

A.3.9 Exit Graphic Test

Type the following sequences to exit the Interactive Graphic test and return the terminal to alphanumeric operation.

Function	Type
Exit test; initialize memories	I 0 SPACE
Clear Register 0	A SPACE SPACE
Clear Register 1	I SPACE SPACE
Exit graphic mode	ESC 2

APPENDIX B

GLOSSARY AND NOTATION

B.1 GLOSSARY

Active Position – The character position on the visual display that is to display the next graphic character.

ANSI Mode – A mode in which the terminal recognizes and responds only to control functions in accordance with ANSI specifications.

Aspect Ratio – The width to height ratio of the image on a video picture tube.

Character – A sequence of 7 or 8 bits that represent a control or graphic entity. In serial-by-bit transmission, the character is transferred from low-order bit to high-order bit.

Character Position – That portion of a visual display which is displaying or is capable of displaying a graphic symbol.

Control – The term "control" refers to a "control function" (that includes a control character, an escape sequence, a control sequence, or control string).

Control Character – A single character whose occurrence in a particular context initiates, modifies, or stops a control function. The value of a control character is in the range of 0 through 37 and 177 octal in a 7-bit environment.

Control Function – An action that affects the recording, processing, transmission, or interpretation of data. This term refers to either a control character, an escape sequence, a control sequence, or a control string.

Control Sequence – A string of characters that begins with a control sequence introducer (CSI) and ends with the first occurrence of a final character (100 – 176 octal). A control sequence may contain zero or more parameter characters (060 – 077 octal) and/or intermediate characters (040 – 057 octal).

Control Sequence Introducer (CSI) – A prefix to a control sequence that provides supplementary control functions. The CSI for the VT105 is ESC[(033 133 octal).

Control String – A string of characters that is used to perform a control function and has an opening and closing control character. These characters are not presently used, are not printed, and are not displayed by the terminal.

Cursor – A visual representation of the active position for the next character, i.e., blinking reverse-video or blinking underline.

Cursor Control – An editor function that moves the active position of the cursor.

Default – A value that is assumed when no explicit value is specified.

Display – The active area of the screen, i.e., the area inside the scrolling region, or the entire screen, depending on the origin mode.

Editor Function – A control that affects the layout or positioning of previously entered or received information. This information is to be interpreted and executed without remaining in the data stream. (See format effector.)

Escape Character (ESC) – A control character that provides supplementary characters (code extension).

Escape Sequence – A string of characters that begins with ESC (033 octal) and ends with the first occurrence of a final character (060 – 176 octal). An escape sequence may have zero or more intermediate characters (040 – 057) preceding the final character.

Final Character – A character whose bit combination terminates a control function. (See example below.)

1. The final character in an escape sequence is from 60_8 to 176_8 inclusive.

2. The final character in a control sequence is from 100_8 to 176_8 inclusive.

Format Effector – A control that affects the layout or positioning of information on the screen. It may remain in the data stream after processing. (See editor function.)

Graphic Character – A character, other than a control character, with a visual representation.

Graph Marker – A short vertical line programmed to mark a point on a graph.

Histogram – A graphic display that is shaded between the graph data and the bottom of the graph drawing field.

Home – The character position at the origin. See origin mode (DECOM); i.e., upper-left corner.

Interactive Graphic Device – A display that allows the user to interact directly with the system for creating, manipulating, and designing graphs using a display console or other device.

Intermediate Character – A character whose bit combination precedes a final character in an escape or control sequence. This character "I" in notation (Paragraph B.2) is from 40_8 to 57_8 inclusive.

Numeric Parameter – A string of characters that represents a number Pn. Pn has a range of 0 (60_8) to 9 (71_8).

Parameter – 1) A string of one or more characters that represent a single value; 2) The value so represented.

Parameter String – A string of characters that represent one or more parameter values.

Pixel – A single picture element in a graphic display.

Selective Parameter – A string of bit combinations from a specified list of subfunctions, designated by Ps. Each Ps character has a range of 0 to 9 (60_8 – 71_8); each character is separated by a semicolon.

Shade Line – The line referenced for shading a graph.

Strip Chart – A graph that moves previous data as new data is added.

VT52 Mode – A mode in which the terminal recognizes and responds only to control functions for VT52 (VT55) DECscope terminals.

B.2 CONTROL FUNCTION NOTATION

The abbreviations or notation used for defining the control functions for the VT105 are in accordance with ANSI standard X3.41 – 1974 and DEC STD 138 which define control function terminology.

B.2.1 Control Sequence Notation

A control sequence is defined by the notation:

ESC[IPn;Pn...PnF

where:

1. ESC[is the control sequence introducer (033 133_8) for a 7-bit character environment.
2. P is a parameter character within the range of 060 to 077_8. Zero or more parameter characters may be included in a control sequence. Parameter characters are separated by a semicolon (073_8).
   a. Pn is a numeric parameter within the range of 0 through 9 (060_8 – 071_8).
   b. Ps is a selective parameter from a specified list.

3. I is an intermediate character within the range of 040 – 057_8. Zero or more intermediate characters may be included in a control sequence.

4. F is a final character within the range of 100 – 176_8.

B.2.2 Escape Sequence Notation

The format of an escape sequence, as defined in American National Standard X 3.41 1974 and used in the VT105, is:

ESC I . . . I F

where:

1. ESC is the introducer control character (33_8) that is named Escape.

2. I . . . I are the intermediate bit combinations that may or may not be present. "I" characters have a range of 40_8 to 57_8 in both 7- and 8-bit character formats.

3. F is the final character. "F" characters have a range of 60_8 to 176_8 using 7 bits; 200_8 to 237_8 using 8 bits.

NOTE

The occurrence of control characters (0_8 to 37_8) within a control or escape sequence is technically an error condition. Recovery is possible by executing the function specified by the control character and then executing the control or escape sequence. The exceptions are:

1. **If the character ESC occurs, the current control function is aborted, and the new control function commences.**

2. **If the character CAN (30_8) or the character SUB (32_8) occurs, the current control function is aborted.**

APPENDIX C
GRAPH DRAWING PROGRAMMER'S CARD

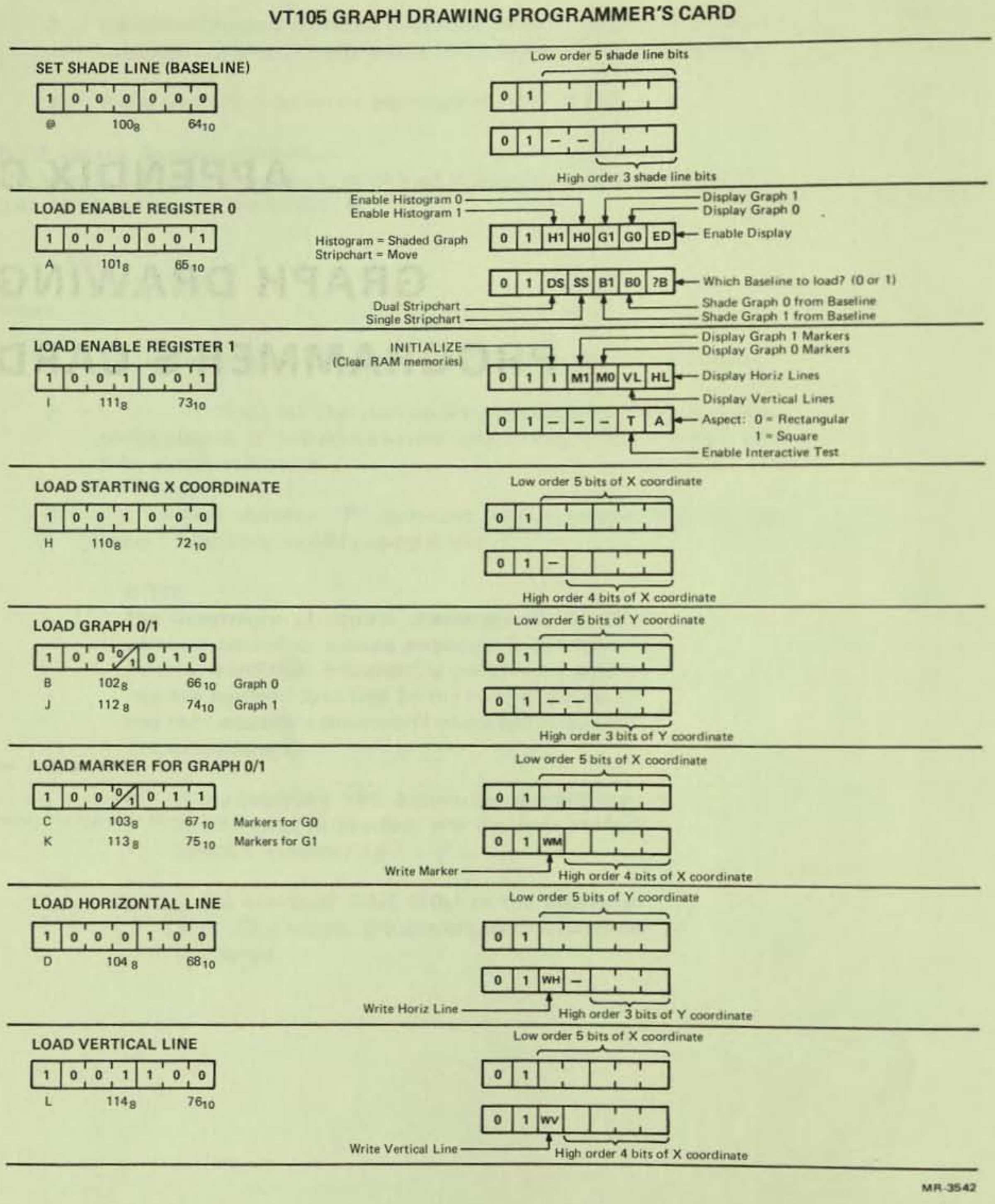


Figure C-1 Graph Drawing Programmer's Card

APPENDIX D

7-BIT ASCII CODE

Table D-1 7-Bit ASCII Code/Character Chart

Octal Code	Char	Octal Code	Char	Octal Code	Char	Octal Code	Char
000	NUL	040	SP	100	@	140	`
001	SOH	041	!	101	A	141	a
002	STX	042	"	102	B	142	b
003	ETX	043	#	103	C	143	c
004	EOT	044	$	104	D	144	d
005	ENQ	045	%	105	E	145	e
006	ACK	046	&	106	F	146	f
007	BEL	047	'	107	G	147	g
010	BS	050	(	110	H	150	h
011	HT	051	)	111	I	151	i
012	LF	052	*	112	J	152	j
013	VT	053	+	113	K	153	k
014	FF	054	,	114	L	154	l
015	CR	055	–	115	M	155	m
016	SO	056	.	116	N	156	n
017	SI	057	/	117	O	157	o
020	DLE	060	0	120	P	160	p
021	DC1	061	1	121	Q	161	q
022	DC2	062	2	122	R	162	r
023	DC3	063	3	123	S	163	s
024	DC4	064	4	124	T	164	t
025	NAK	065	5	125	U	165	u
026	SYN	066	6	126	V	166	v
027	ETB	067	7	127	W	167	w
030	CAN	070	8	130	X	170	x
031	EM	071	9	131	Y	171	y
032	SUB	072	:	132	Z	172	z
033	ESC	073	;	133	[	173	{
034	FS	074	<	134	\	174	|
035	GS	075	=	135	]	175	}
036	RS	076	>	136	∧	176	~
037	US	077	?	137	_	177	DEL

APPENDIX E

FILL CHARACTER REQUIREMENTS

E.1 RECEIVE SPEED VS. FILL CHARACTERS

NOTE
This appendix is only for systems that cannot use synchronization control codes XON and XOFF.

Fill characters are required to help keep the terminal synchronized with the host computer when the XON/XOFF control codes are not used. Table E-1 shows the fill character requirements for every receive speed at which the terminal is capable of operating. No entry in a column indicates that fill characters are not required in that specific case.

Table E-1 Fill Character Requirements

Control Function		NEL, IND, LF, RI (Smooth scroll feature selected)	DECALN	DECCOLM	ED	NEL, IND, LF, RI (Jump scroll feature selected)	EL	DECINLM	All others except DECTST and RIS
Receive	**19200**	384	243	243	90	60	4	2	2
Baud	**9600**	192	122	122	45	30	2	1	1
Rate	**4800**	96	61	61	23	15	1	1	
	3600	72	46	46	17	11	1		
	2400	48	30	30	11	7			
	2000	40	25	25	9	6			
	1800	36	23	23	9	6			
	1200	24	15	15	6	4			
	600	12	8	8	3	2			
	300	6	4	4	1	1			
	200	4	3	3	1	1			
	150	3	2	2	1				
	134.5	3	2	2	1				
	110	2	1	1	1				
	75	2	1	1					
	50	1	1	1					

VT105 Graphic Terminal
User's Guide
EK-VT105-UG

Reader's Comments

Your comments and suggestions will help us in our continuous effort to improve the quality and usefulness of our publications.

What is your general reaction to this manual? In your judgment is it complete, accurate, well organized, well written, etc.? Is it easy to use? ______

What features are most useful? ______

What faults or errors have you found in the manual? ______

Does this manual satisfy the need you think it was intended to satisfy? ______

Does it satisfy *your* needs? ______ Why? ______

☐ Please send me the current copy of the *Technical Documentation Catalog,* which contains information on the remainder of DIGITAL's technical documentation.

Name ______ Street ______
Title ______ City ______
Company ______ State/Country ______
Department ______ Zip ______

Additional copies of this document are available from:

Digital Equipment Corporation
Accessories and Supplies Group
Cotton Road
Nashua, NH 03060
Attention: Documentation Products
Telephone: 1-800-258-1710

Order No. EK-VT105-UG

For service outside the U.S. and Canada, contact your local Digital Sales Office or authorized Digital representative.

digital

I'm interested. Please provide me with more information on the following:

- ☐ On-Site Service
- ☐ Off-Site Service
- ☐ Flat Rate Repair
- ☐ Educational Services
- ☐ Customer Spares
- ☐ Terminal

Name ____________________

Address ____________________

____

____

Phone No. ____________________

Fold Here

Do Not Tear – Fold Here and Staple

digital

No Postage Necessary if Mailed in the United States

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 33 MAYNARD, MA.

POSTAGE WILL BE PAID BY ADDRESSEE

Digital Equipment Corporation
Communications Development and Publishing
200 Forest Street (MR1-2/T17)
Marlboro, MA 01752

INSTALLATION, WARRANTY, AND SERVICE INFORMATION

INSTALLATION/WARRANTY

For customers who have purchased directly from DIGITAL, reference the sales agreement for installation and warranty terms purchased with this terminal.

For customers who have purchased, leased, or rented from a vendor other than DIGITAL, contact your vendor for information regarding installation and warranty terms purchased with this terminal.

DIGITAL SERVICES

A wide range of maintenance and customer services are available from DIGITAL for your terminal. Through use of these services, you can design a plan which meets your service needs, from complete DIGITAL maintenance to complete self-maintenance. Vendors supplying DIGITAL products may use these services as factory backup support.

- **On-Site Service**
  DIGITAL offers responsive, low cost, factory-level maintenance performed at your site by trained Terminals Service Specialists through:
  - Service Agreements which cover all your maintenance needs to include priority response; all labor, materials and travel for a fixed monthly charge.
  - Per Call Service which is provided on a "time" and "materials" basis and can serve as a back-up to your own in-house maintenance programs.
- **Off-Site Service**
  For those customers who have a significant level of troubleshooting expertise, but require Field Service assistance for the repair of components, DIGITAL has established a worldwide network of Product Repair Centers (PRCs) and the Customer Returns Area (CRA). Through a wide array of service product offerings this logistics network offers cost effective services to include:
  - Module Mailer™
  - Fixed Price Exchange
  - Product Refurbishment
- **Spare Parts**
  In an effort to further assist those customers who choose to perform their own computer maintenance, DIGITAL's Customer Spares organization is committed to providing thorough and timely spares support through:
  - Spares Inventory Planning
  - Component/Subassembly Spares
  - Maintenance Test Equipment
  - Maintenance Documentation Service
  - Emergency Spare Parts
- **Training**
  DIGITAL offers hardware maintenance courses through the Educational Services group at any of our 17 worldwide training centers; or depending on your specific training requirements, courses can be provided in your own facilities.
- **Terminal Supplies**
  DIGITAL offers a variety of supplies to enhance terminal reliability and to make operation easier. Everything from furniture accessories, (cabinets, tables, etc.) to terminal supplies, such as paper, ribbons, diskettes, cassettes, labels, and many other items, as illustrated in the Supplies Brochures, are available through DIGITAL and may be obtained by dialing the toll-free number below. All orders may be placed via the toll-free number (800-258-1710) and will be processed within 24 hours (U.S. only – with the exception of Hawaii and Alaska).

To obtain further information concerning any of the customer services available, fill out the attached card or write:

DIGITAL EQUIPMENT CORPORATION
129 Parker Street
Maynard, MA 01754

ATTN: Customer Services Marketing – PK3-2/S25

digital