

AD-A040 634

COMPUTER SCIENCES CORP ARLINGTON VA
DESCRIPTION OF THE REMOTE TERMINAL EXECUTIVE (REMTX) POSTPROCE--ETC(U)
JAN 77 W H POLLITT, M C FLYTHE

F/G 9/2

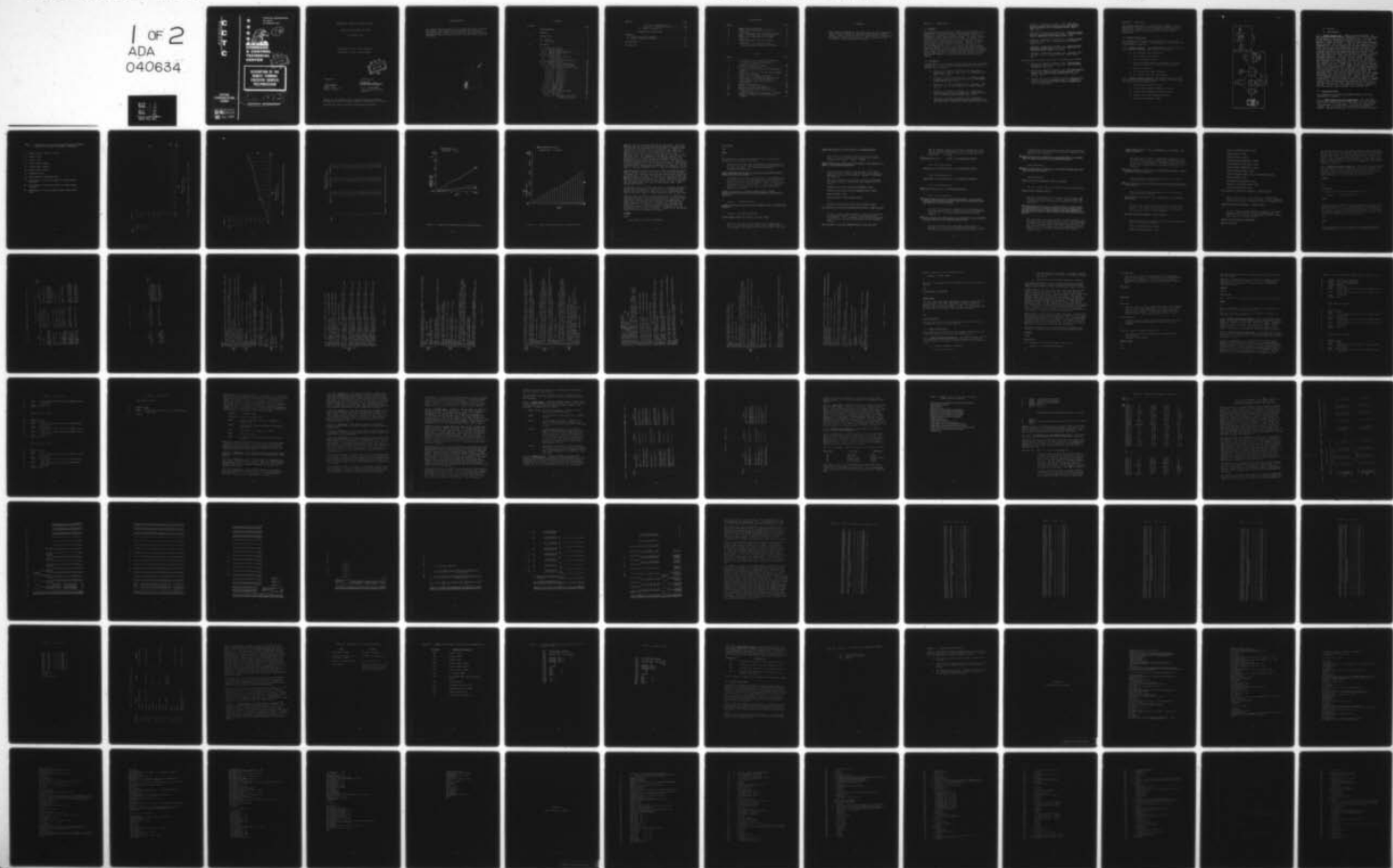
DCA100-74-C-0002

UNCLASSIFIED

CCTC-TM-144-77

NL

1 OF 2
ADA
040634



ADA 040634

C H C C

TECHNICAL MEMORANDUM
TM 144-77
10 JANUARY '1977'



**COMMAND
& CONTROL
TECHNICAL
CENTER**

12

DDC
JUN 16 1977

**DESCRIPTION OF THE
REMOTE TERMINAL
EXECUTIVE (REMTEX)
POSTPROCESSOR**

See 1473

TECHNICAL MEMORANDUM

DEFENSE
COMMUNICATIONS
AGENCY

Approved for public release;
distribution unlimited.

AD NO.

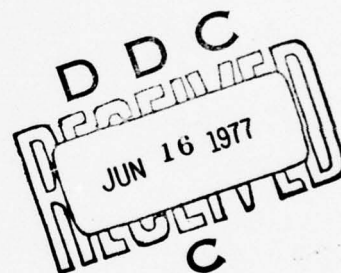
DDC FILE COPY

COMMAND AND CONTROL TECHNICAL CENTER

Technical Memorandum TM 144-77

10 January 1977

DESCRIPTION OF THE REMOTE TERMINAL
EXECUTIVE (REMTX) POSTPROCESSOR



REVIEWED BY:

JOHN E. ROLEY, JR.
Project Officer

APPROVED BY:

R. E. HARSHBARGER
Acting Deputy Director
NMCS ADP

Copies of this document may be obtained from the Defense
Documentation Center, Cameron Station, Alexandria, VA 22314.

Approved for public release; distribution unlimited.

ACKNOWLEDGMENT

This manual was prepared for the Command and Control Technical Center (CCTC) under the direction of the Chief for Development and Analysis with technical support provided by Computer Sciences Corporation under Contract Number DCA 100-74-C-0002.

DATE	2015 04 20	✓
TIME	10:00 AM	✓
BY	Jeff Soodan	✓
REASON		<input type="checkbox"/>
JUSTIFICATION		
BY		
DISTRIBUTION/AVAILABILITY CODES		
DATE		
AVAIL. DATE/SPECIAL		

A

CONTENTS

Section	Page
ACKNOWLEDGMENT	ii
ABSTRACT	vi
1. INTRODUCTION	1
1.1 General	1
1.2 References	1
2. DISCUSSION	3
2.1 System Description	3
2.1.1 REMTEX Hardware	3
2.1.2 REMTEX Support Software	3
2.1.3 REMTEX System Flow	5
2.2 System Operation	5
2.2.1 REMTEX Editor and User Interaction	5
2.2.2 REMTEX Processor Operation	19
2.3 System Maintenance	29
2.3.1 RXEDIT Program Maintenance	29
2.3.2 RXPROC Program Maintenance	32
2.3.2.1 RXPROC Main Routine Logic	32
2.3.2.1.1 Section A	32
2.3.2.1.2 Section B	36
2.3.2.1.3 Section C	36
2.3.2.1.4 Section D	36
2.3.2.1.5 Section E	37
2.3.2.1.6 Section F	37
2.3.2.1.7 Section G	37
2.3.2.1.8 Section H	37
2.3.2.1.9 Section I	37
2.3.2.2 PLOTTR Logic	38
2.3.2.3 NEWAX Logic	39
2.3.2.4 ERRMSG Logic	39
2.3.2.5 MORE Logic	42
2.3.3 REMTEX File Maintenance	42
2.3.3.1 TSS Files	42
2.3.3.2 BCD Files	42
2.3.3.2.1 Dictionary File and Summary Data File	44

Section	Page
2.3.3.2.2 Commands File	64
2.3.3.2.3 File Pointers File	69
2.4 REMTEX Flexibility	69
3. FINDINGS AND CONCLUSIONS	71
APPENDIX	
A. Source Listing of RXEDIT	73
B. Source Listing of RXPROC	83
DISTRIBUTION	103
DD Form 1473	105

ILLUSTRATIONS

Figure		Page
1	REMTEX System Flowchart	4
2	Sample Point Graph From the Online Printer	7
3	Sample Histogram From the Online Printer .	8
4	Sample Table From the Online Printer . . .	9
5	Sample Line Graph From the CalComp Plotter	10
6	Sample Histogram From the CalComp Plotter	11
7	Illustration of Exception Messages Occurring During Execution of RXEDIT . .	22

TABLES

Number		Page
1	Description of User Specified Portions of RXPROC Output as Displayed in Figures 2 Through 6	6
2	Messages Displayed by RXEDIT	20
3	JCL for Altering RXPROC Routines	33
4	Exception Messages Produced by Subroutine ERRMSG	40
5	Example of a JCL File for Executing RXPROC From the Terminal	43
6	Format for the Summary Data File	45
7	Comparison of an IDAGAM II Summary Table and Summary Data Created From It	47
8	Sample Summary Data File Produced by IDAGAM II	48
9	Sample Dictionary File	56
10	Format for Dictionary Items	63
11	Structure of the Dictionary File	65
12	Command Identification Keys for the Commands File	66
13	A Sample Commands File Produced by RXEDIT .	67
14	Sample File Pointers File Created by RXEDIT	70

ABSTRACT

This Technical Memorandum (TM) describes the newly developed REMote Terminal EXecutive (REMTEX) postprocessor software system. User interaction and input requirements are described and output formats are presented. Basic program and file maintenance instructions are included.

SECTION 1. INTRODUCTION

1.1 General

The REMote Terminal EXECutive (REMTEX) software package is a generalized, dictionary-driven, postprocessor system. Its purpose is to retrieve, reformat, and display output data produced by other HIS 600/6000 programs. The output formats available through REMTEX are point graph, histogram, and tabular on the online printer and line graph and histogram on an offline CalComp flat bed plotter. REMTEX is coded in Series 6000 FORTRAN (FORTRAN Y) and will execute at any HIS 600/6000 computer site. To utilize the complete capability of REMTEX, a user's site also should have a CalComp flat bed plotter and one or more interactive remote terminals. Also, users located at Worldwide Military Command and Control System (WWMCCS) sites can utilize the Worldwide Data Management System (WWDMS) to produce additional tabular output reports.

1.2 References

Information on the programming conventions used for the REMTEX program may be found in the following Honeywell Information Systems, Inc., manuals:

- a. Honeywell Information Systems, Inc., Bulk Media Conversion, Order No. DD11, Rev. 0, Wellesley Hills, Mass., April 1974.
- b. Honeywell Information Systems, Inc., Control Cards Reference Manual, Order No. DD31, Rev. 0, Wellesley Hills, Mass., March 1974.
- c. Honeywell Information Systems, Inc., FORTRAN, Order No. DD02, Rev. 0, Wellesley Hills, Mass., January 1975.
- d. Honeywell Information Systems, Inc., GCOS Time-Sharing System General Information, Order No. DD22, Rev. 0, Wellesley Hills, Mass., April 1974.
- e. Honeywell Information Systems, Inc., GCOS Time-Sharing System Programmer's Reference Manual, Order No. DD17, Rev. 0, Wellesley Hills, Mass., March 1974.

- f. Honeywell Information Systems, Inc., GCOS Time-Sharing System Terminal/Batch Interface Facility, Order No. DD21, Rev. 0, Wellesley Hills, Mass., April 1974.
- g. Honeywell Information Systems, Inc., General Comprehensive Operating Supervisor (GCOS), Order No. DD19, Rev. 0, Wellesley Hills, Mass., April 1974.
- h. Honeywell Information Systems, Inc., General Loader, Order No. DD10, Rev. 0, Wellesley Hills, Mass., March 1974.
- i. Honeywell Information Systems, Inc., General Macro Assembler Program (GMAP), Order No. DD08, Rev. 0, Wellesley Hills, Mass., July 1974.
- j. Honeywell Information Systems, Inc., Utility, Order No. DD12, Rev. 0, Wellesley Hills, Mass., April 1974.

The following documents were used as references for REMTEX:

- a. California Computer Products, Inc., CalComp Basic Software CAL EDIT User's Guide, No. 1058, Anaheim, California, May 1975.
- b. California Computer Products, Inc., CalComp Graphics Functional Software USAS FORTRAN/Business User's Manual, No. 1011, Anaheim, California, Revised October 1969.
- c. Command and Control Technical Center, Institute for Defense Analysis Ground-Air Model II (IDAGAM II), Computer System Manual CMS UM 201-75, Washington, D.C., October 1976.

SECTION 2. DISCUSSION

This section describes the functioning of REMTEX, the user interaction required for the operation of REMTEX, basic maintenance information for the REMTEX programs and files, and several operational techniques to utilize system flexibility.

2.1 System Description

This subsection presents the environment and the system flow of REMTEX. A graphic representation of the system flow is depicted in figure 1.

2.1.1 REMTX Hardware. The REMTEX system requires the following minimum hardware items for full operation:

- a. HIS 600/6000 series computer
- b. 38K available core memory
- c. Two 9-track tape drives
- d. Sufficient disk space to accommodate all files (space required will vary from user to user)
- e. One online printer
- f. One interactive remote terminal
- g. One CalComp 7000 flat bed plotter.

2.1.2 REMTX Support Software. The REMTEX system requires the following GCOS software support packages for full operation plus maintenance capability:

- a. Time-Sharing Subsystem (TSS)
- b. Series 6000 FORTRAN (FORTRAN Y) Compiler
- c. General Macro Assembler Program (GMAP)
- d. Remote Card Input Subsystem (CARDIN)
- e. Bulk Media Conversion (BMC)

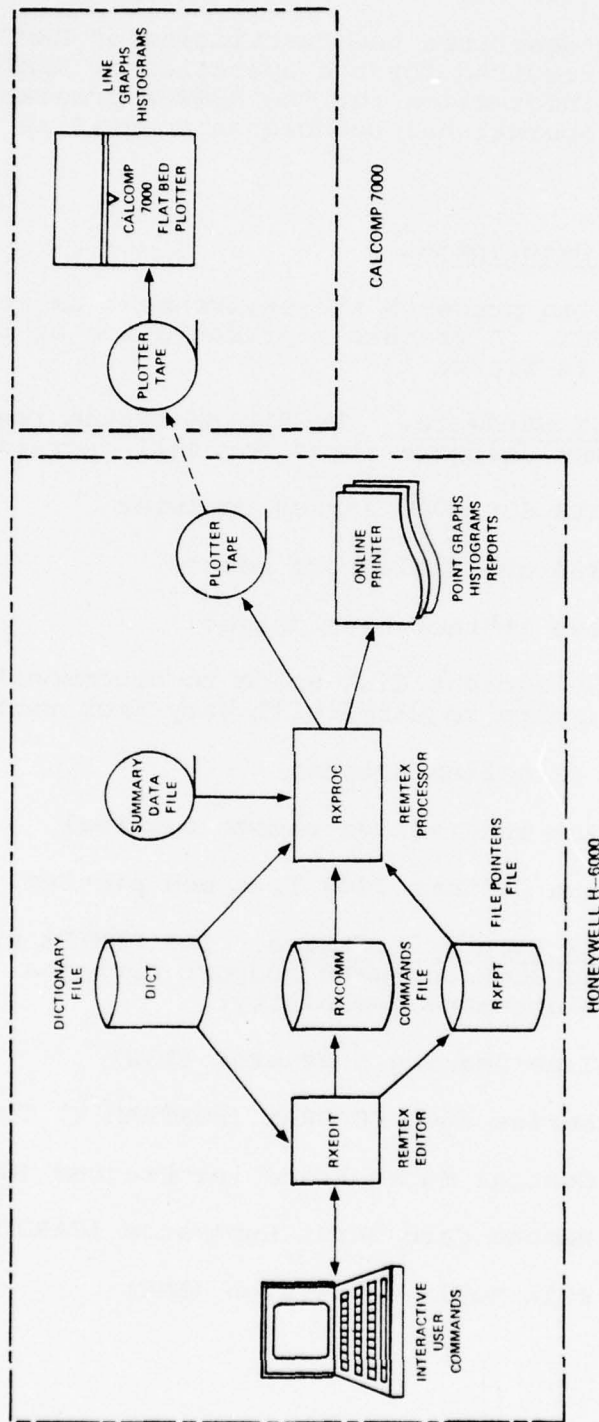


Figure 1. REMTEX System Flowchart

f. Utility

g. TEXT EDITOR.

2.1.3 REMTEX System Flow. Before initiating REMTEX, the user should provide computer operations with the reel numbers of the magnetic tapes that he intends to use. The user then should obtain an interactive terminal and log on to the computer. Refer to figure 1. When the user calls the REMTEX Editor program, the Editor will ask for the locations of the Commands file, the Dictionary file, and the Summary Data file. After these files are successfully attached and their locations are stored on the File Pointers file, the Editor will initiate a dialog that prompts the user as he enters the commands describing the reports he wants as output. When the user terminates the dialog, the Editor will detach all files and cease execution. The user will then place the REMTEX Processor program in execution through CARDIN. When the user has copied his job number, he may log off the terminal. The Processor will attach the File Pointers file and read the locations of the Commands file, the Dictionary file, and the Summary Data file. These files, as well as the Plotter tape, will then be attached. Based on the reporting requirements in the Commands file and the legal variable definitions in the Dictionary file, the Processor will examine the data from the Summary Data file and reformat and output the retrieved data. The possible output formats are point graphs, histograms, and tables on the online printer and line graphs and histograms on the CalComp 7000. Table 1 defines each output portion that is flagged by a circled number in figures 2 through 6. Figures 2 through 6 depict each type of REMTEX output. When the Processor has completed execution, the user may pick up his online output as part of his execution listing. At this time, the user should submit a request to have the plotter tape processed on the CalComp 7000.

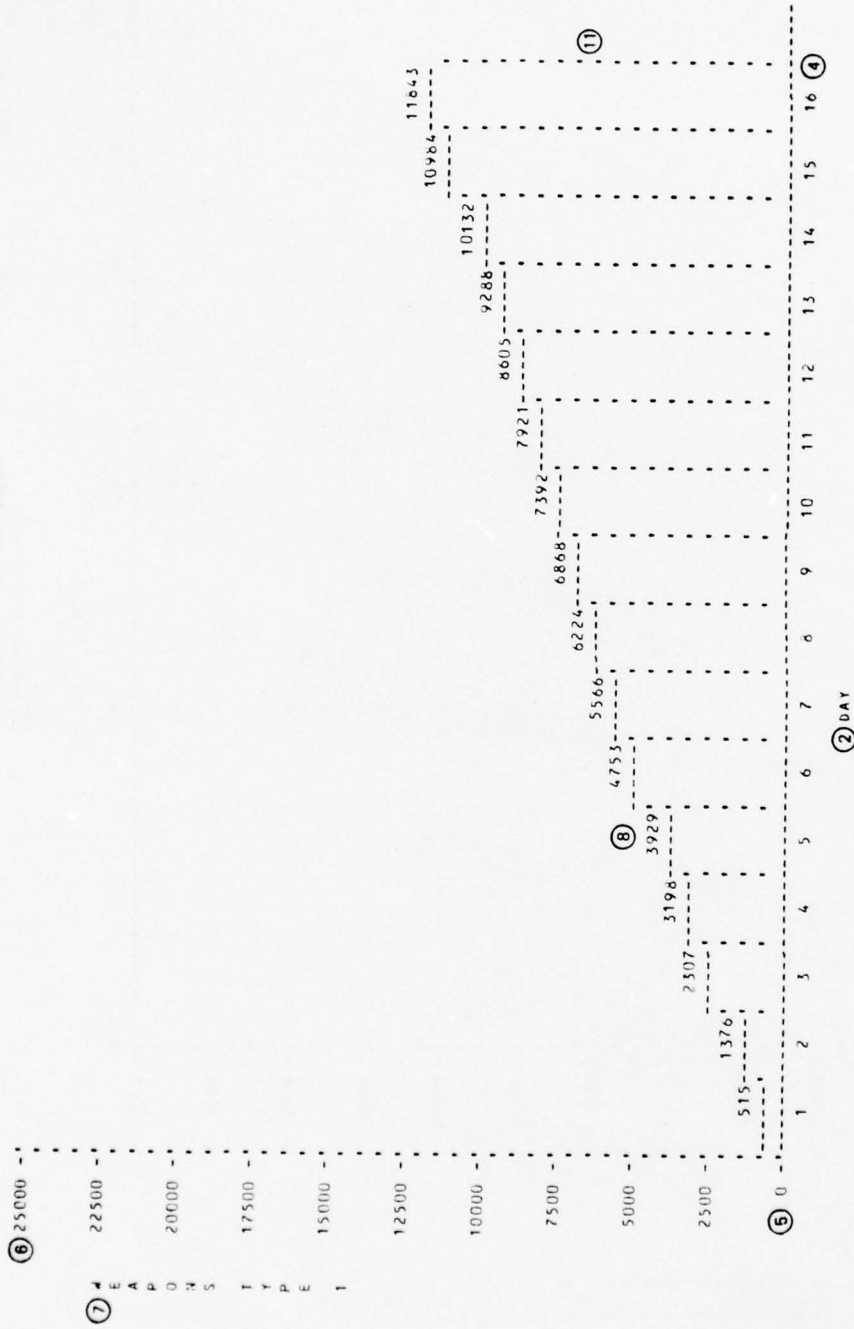
2.2 System Operation

This subsection describes the REMTEX operation and user interaction in detail.

2.2.1 REMTEX Editor and User Interaction. The user must provide the location of the Summary Data file, the Commands file, and the Dictionary file. The user also must specify the processing to be performed. This may be accomplished through user interaction with the REMTEX edit program, RXEDIT. Appendix A contains a source code listing of RXEDIT.

Table 1. Description of User Specified Portions of RXPROC Output as Displayed in Figures 2 Through 6

1. Report title, lines 1, 2 and 3
2. X-axis title
3. Y-axis title
4. X-axis scale (upper)
5. Y-axis scale (lower)
6. Y-axis scale (upper)
7. Histogram title
8. Data values on histogram bars
9. Point graph of Y-axis variable #1 versus X-axis variable
10. Point graph of Y-axis variable #2 versus X-axis variable
11. Histogram of Y-axis variable versus X-axis variable



(1) RED WEAPONS LUSSES
 REPORT DATE - 19 JAN 77

Figure 3. Sample Histogram From the Online Printer

① RED WEAPONS LOSSES

 REPORT DATE - 19 JAN 77

② DAY	WEAPONS TYPE 1	WEAPONS TYPE 2	WEAPONS TYPE 3
1	515.0	84.0	125.0
2	1376.0	220.0	351.0
3	2307.0	368.0	592.0
4	3198.0	507.0	810.0
5	3929.0	622.0	966.0
6	4753.0	753.0	1188.0
7	5566.0	882.0	1387.0
8	6226.0	987.0	1553.0
9	6868.0	1090.0	1715.0
10	7392.0	1173.0	1846.0
11	7921.0	1258.0	1979.0
12	8605.0	1367.0	2144.0
13	9288.0	1475.0	2309.0
14	10132.0	1611.0	2517.0
15	10984.0	1748.0	2727.0
16	11843.0	1887.0	2938.0
17	12745.0	2032.0	3159.0
18	13496.0	2152.0	3345.0
19	14385.0	2295.0	3564.0
20	15414.0	2462.0	3811.0

Figure 4. Sample Table From the Online Printer

① RED WEAPONS LOSSES
REPORT DATE - 19 JAN 77

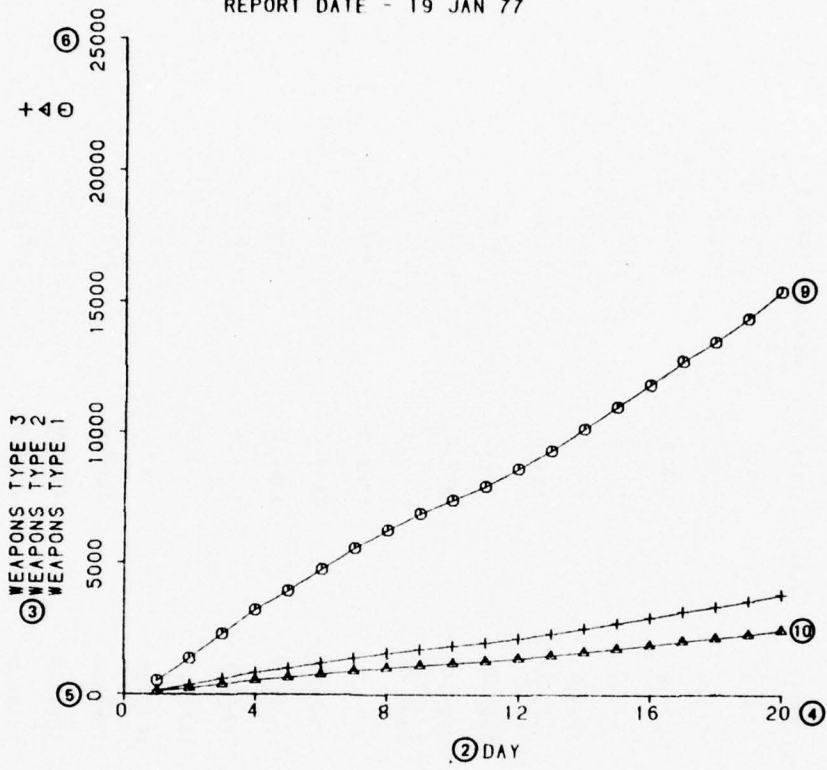


Figure 5. Sample Line Graph From the CalComp Plotter

① RED WEAPONS LOSSES

REPORT DATE - 19 JAN 77

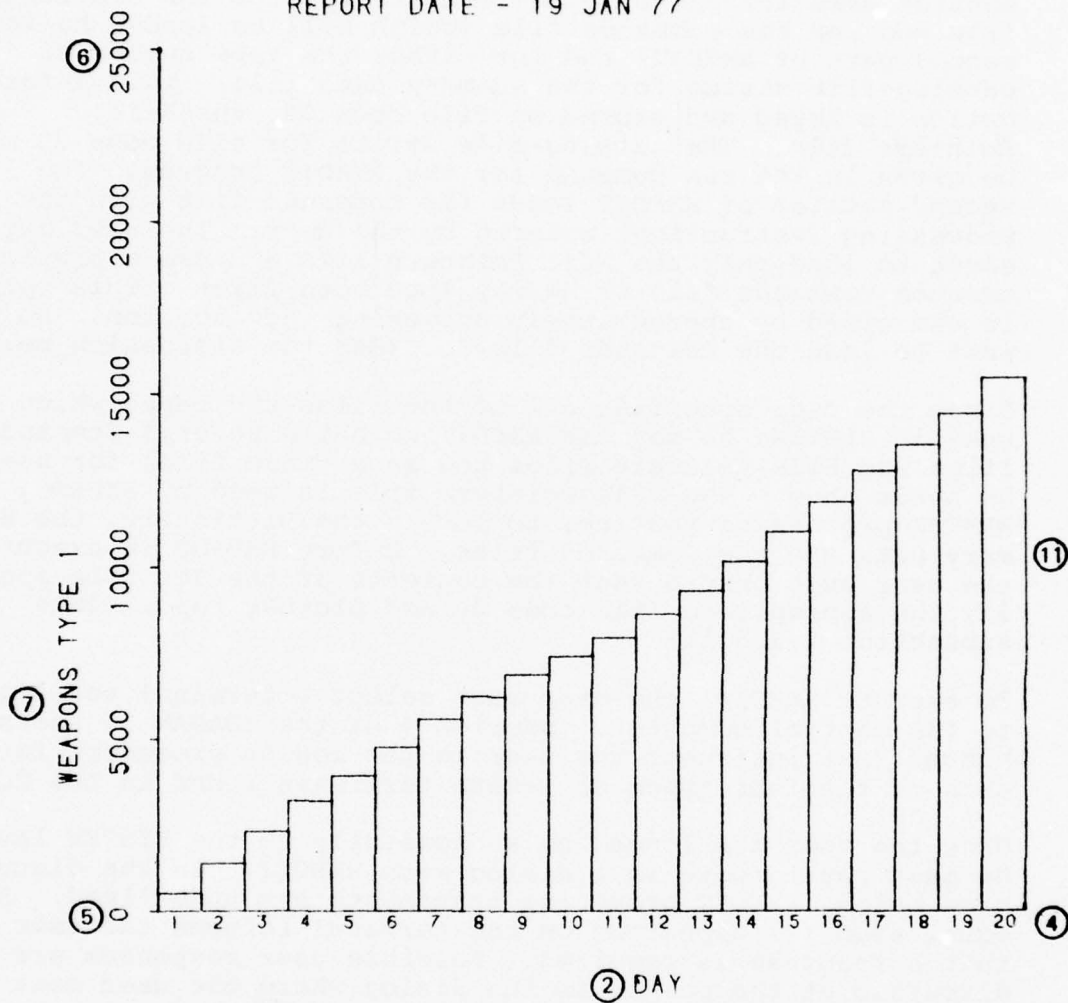


Figure 6. Sample Histogram From the CalComp Plotter

RXEDIT contains two major sections of inquiries. The first section asks for the catalog-file strings for the Dictionary file and for the Commands file (which will be loaded by the second part of RXEDIT) and for either the tape number or catalog-file string for the Summary Data file. This information is keyed and stored on file code 30, the File Pointers file. The catalog-file string for file code 30 must be given in the run command for the RXEDIT program. The second section of RXEDIT loads the Commands file with the processing instructions entered by the user. The user may elect to load only the File Pointers file and use a previously created Commands file or he may load both files. This option is exercised by appropriately answering the question, "Do you want to load the Commands file?". (See the discussion below.)

Since the user specifies all of the files and tapes which are used in REMTEX, he may use RXEDIT to build several Commands files and File Pointers files and save these files for use as he needs them. The File Pointers file is used by RXPROC, the REMTEX processing routine, to locate the Dictionary, the Summary Data and the Commands files. Before RXPROC is executed, the user must ensure that the contents of the JCL file specify the appropriate file code 30 and plotter tape. (See subsection 2.3.3.1.)

To execute RXEDIT, the user must select a terminal and log on to the central computer. Section 4 of the IDAGAM II Users Manual (UM) instructs the user in the log on procedure for each of the four types of remote terminals found in the CCTC.

Once the user has logged on successfully to the SYSTEM level, he must participate in a dialog with RXEDIT. In the discussion below, RXEDIT prompting statements are underlined. An equal sign (=) appearing on the terminal informs the user that a response is required. Possible user responses are discussed at the points in the dialog where the user must provide an answer. Circled numbers, e.g. ②, refer to the illustration in figures 2 through 6. Beginning at the SYSTEM level, the dialog proceeds as follows:

SYSTEM?

YFORT

The system to be used is FORTRAN Y.

OLD OR NEW

NEW

READY

*
—

RUN catalog-file string for RXEDIT H* file # catalog-file
string for File Pointers file "30"

This input from the user attaches the File Pointers file
to file code 30 and loads and executes RXEDIT from the
H* FILE. Example: RUN 674IDP00/CSC/RXEDITHS #
674IDP00/CSC/RXFPT"30".

NOTIFY COMPUTER OPERATIONS TO OBTAIN ALL NECESSARY MAGNETIC
TAPES FROM THE TAPE LIBRARY.

No response is required from the user. This message is
a reminder to the user that he must notify computer oper-
ations in advance to obtain from the tape library any
magnetic tapes required by RXPROC. One tape is required
by RXPROC for the plotter data. If the input Summary
Data file is on tape, that tape is also required by
RXPROC for execution.

REMTEX IS EXECUTING - PLEASE RESPOND WHEN = APPEARS.
ENTER CATALOG-FILE STRING FOR THE DICTIONARY, 36 CHARACTERS
MAXIMUM.

=

Example: 674IDP00/CSC/DICT.

ENTER CATALOG-FILE STRING FOR THE COMMANDS FILE, 36 CHARACTERS
MAXIMUM.

=

Example: 674IDP00/CSC/RXCOMM.

ENTER INPUT SOURCE "T", "TAPE", "D", OR "DISK".

=

The user must specify the medium of the Summary Data
file. If the user cannot be sure that a tape drive will
be available, his summary data should be input from disk.

ENTER TAPE NUMBER OF INPUT SOURCE - 5 INTEGER DIGITS.

=

This prompting statement appears only if the input source is tape, in which case the user must enter a 5-digit reel number. Example: 99999.

ENTER CATALOG-FILE STRING OF INPUT SOURCE - CAT-1/CAT-2/.../FILE, -- 36 CHARACTERS MAXIMUM.

=

This prompting statement appears only if the input source is disk, in which case the user must enter the appropriate catalog-file string. Example: 674IDP00/CSC/SUMMDATA.

The user responses to the above prompting statements are summarized and displayed by the computer at this point:

CATALOG-FILE STRING FOR THE DICTIONARY = text

CATALOG-FILE STRING FOR THE COMMANDS FILE = text

INPUT SOURCE = text

TAPE NUMBER OF INPUT SOURCE = text

or

CATALOG-FILE STRING FOR THE INPUT SOURCE = text.

DO YOU WISH TO CHANGE ANY CATALOG FILE NAMES - 'YES' OR 'NO'.

=

The YES response causes RXEDIT to return to the prompting statement "ENTER CATALOG-FILE STRING FOR THE DICTIONARY". The NO response causes RXEDIT to load the File Pointers file and to continue with the following dialog:

DO YOU WISH TO LOAD THE COMMANDS FILE - 'YES' OR 'NO'.

=

The NO response causes the program to detach all files and to terminate immediately. The YES response causes the program to continue the dialog, enabling the Commands file to be loaded.

① ENTER LINE 1 OF REPORT TITLE - 36 CHARACTERS MAXIMUM.

=

Text, self-explanatory.

ENTER LINE 2 OF REPORT TITLE - 36 CHARACTERS MAXIMUM.

=

Text, self-explanatory.

ENTER LINE 3 OF REPORT TITLE - 36 CHARACTERS MAXIMUM.

=

Text, self-explanatory.

② ENTER X-AXIS TITLE - 36 CHARACTERS MAXIMUM.

=

Text, self-explanatory.

③ ENTER Y-AXIS TITLE 1, 36 CHARACTERS MAXIMUM. IF YOU HAVE ALREADY TRANSMITTED YOUR LAST Y-AXIS TITLE, ENTER ONE BLANK AND EXECUTE A CARRIAGE RETURN.

=

Y-axis title names will continue to be requested until the user either enters a blank character or reaches the limit of six. At least one nonblank title must be entered.

④ ENTER X-AXIS SCALE (UPPER END)--'D', 'DEFAULT', OR A POSITIVE INTEGER, WHICH IS LESS THAN OR EQUAL TO 100.

=

The user either inputs a maximum X range value or defaults to a value to be determined by RXPROC. If the default option is selected for the X-axis scale, the

Y-axis scales (upper and lower) will also be determined by RXPROC and the following Y-axis scale prompting statements will be omitted.

- ⑤ ENTER Y-AXIS SCALE (LOWER END), AN INTEGER WHICH IS GREATER THAN OR EQUAL TO ZERO AND LESS THAN ONE MILLION.

=

Self-explanatory.

- ⑥ ENTER Y-AXIS SCALE (UPPER END), AN INTEGER GREATER THAN ZERO AND LESS THAN ONE MILLION.

=

Self-explanatory.

ENTER X-AXIS VARIABLE NAME, 'DAY' OR 'INDEX'.

=

The user selects one of the legal X-axis variable names.

ENTER Y-AXIS VARIABLE NAME 1.

=

The user selects one of the legal Y-axis variable names listed in the Dictionary. A Y-axis variable name must be compatible with the previously selected X-axis variable name.

ENTER QUALIFIER VALUE 1. IF X-AXIS VARIABLE IS 'DAY', ENTER '0' FOR TOTALS OR A POSITIVE INTEGER TO SET AN INDEX VALUE; IF X-AXIS VARIABLE IS 'INDEX', ENTER A POSITIVE INTEGER TO SPECIFY THE DAY NUMBER. THE VALUE MUST BE LESS THAN OR EQUAL TO 100.

=

The combination of X-axis variable, Y-axis variable, and qualifier value is illegal if not found in the Dictionary, in which case this interactive program will again ask for the X-axis variable name. The Y-axis variable name and the qualifier value requests will be repeated until the number of Y-axis variable names equals the number of Y-axis titles.

ENTER OUTPUT FORMAT - "H", "HISTOGRAM", "G", "GRAPH", "B",
"BOTH", "TABLE".

=

The user selects either a histogram, a graph, or both or he selects a table. If the user selects BOTH, only the first Y-axis variable will appear on the histogram. If the user selects TABLE, a tabular report will be produced on the online printer; no other report will be produced.

⑦ ENTER HISTOGRAM Y-AXIS TITLE - 36 CHARACTERS MAXIMUM. BLANK
DEFAULTS TO Y-AXIS TITLE 1.

=

This prompting statement appears only if the output format includes histogram.

⑧ DO YOU WISH TO DISPLAY DATA VALUES ABOVE EACH HISTOGRAM BAR--
"YES" OR "NO".

=

This prompting statement appears only if the output format includes histogram.

ENTER OUTPUT DESTINATION - "C", "CALCOMP", "P", "PRINTER",
"B", "BOTH".

=

The user must select either the CalComp 7000 flat bed plotter, the H-6080 online printer, or both for his reports. This prompting statement does not appear if the selected output format was TABLE.

DISPLAY OF REPORT DESIRED - 'YES' OR 'NO'.

=

The NO response will suppress the following summary display of the report request:

LINE 1 OF REPORT TITLE = text

LINE 2 OF REPORT TITLE = text

LINE 3 OF REPORT TITLE = text

X-AXIS TITLE = text

Y-AXIS TITLE n = text

X-AXIS SCALE (UPPER END) = text

Y-AXIS SCALE (LOWER END) = text

Y-AXIS SCALE (UPPER END) = text

X-AXIS VARIABLE NAME = text

Y-AXIS VARIABLE NAME n AND ITS QUALIFIER = text

OUTPUT FORMAT = text

HISTOGRAM Y-AXIS TITLE = text

DISPLAY OF HISTOGRAM DATA = text

OUTPUT DESTINATION = text.

DO YOU WISH TO KEEP THIS REPORT? -- 'YES' OR 'NO'.

=

The NO response will cause RXEDIT to eliminate this request and continue with the dialog. The YES response retains the request and loads it onto the Commands file.

ADDITIONAL REPORTS DESIRED? - 'YES' OR 'NO'.

=

The YES response causes RXEDIT to initiate a new report request, beginning with the dialog "ENTER LINE 1 OF REPORT TITLE". The NO response causes RXEDIT to terminate after the following dialog:

REMTEX EDIT PROGRAM HAS COMPLETED.

HAVE A NICE DAY.

*

-

During the execution of the RXEDIT program, various messages may be displayed on the screen, depending on the entries made by the user. Some of these messages are merely informative, others require that the user take some action. Table 2 contains a list of messages which may appear. This table also gives an interpretation of each message, indicates what action the user should take and refers to an example of that kind of message in figure 7.

2.2.2 REMTEX Processor Operation. After the user has loaded one or more File Pointer files and one or more Commands files, the user must initiate execution of the REMTEX Processor (RXPROC). The user may accomplish this by returning to the system level at his terminal and initiating the dialog described below. HIS system dialog is underlined.

SYSTEM?

*
—

CARD

OLD OR NEW

O Catalog-file string for the RXPROC JCL file.

Example: O 674IDP00/CSC/RXJCL.

READY

*
—

Before placing the RXPROC JCL into execution, the user should examine it and alter it if the plotter tape reel number or the File Pointers file catalog-file string is different from the one he is using for this REMTEX execution. The JCL may be altered interactively on the terminal after listing. The JCL may be listed by the following dialog.

LIST

Text

*
—

If the alteration of the RXPROC JCL is permanent, the user should capture the new JCL through the following dialog:

Table 2. Messages Displayed by RXEDIT (Part 1 of 2)

Message	Interpretation	User Action	Example in Figure 7
ILLEGAL CATALOG-FILE STRING.	The user entered a catalog file string for a file which was not accessible. It may be nonexistent or busy.	Enter new catalog-file string when asked.	1
ILLEGAL ENTRY.	Entry was not one which matched the instructions for that prompting statement.	Read prompting statement to see what entries are legal.	2
COMPLETE DICTIONARY IS NOT IN STORAGE. THERE IS ONLY ROOM FOR N ITEMS. PROGRAM TERMINATED.	The parameter NDP is smaller than the number of Dictionary items.	Check the size of the Dictionary list, making sure that the last entry is 'ZZZZZ'. Then adjust NDP in both RXEDIT and RXPROC to accommodate the Dictionary.	11
THE FIRST Y-AXIS TITLE CANNOT BE BLANK.	The user has entered a blank Y-axis title.	Enter a nonblank title.	4
ILLEGAL X-AXIS VARIABLE USED.	The user entered a name other than "DAY" or "INDEX" for the X-axis variable name.	Enter a legal name.	6
ILLEGAL Y-AXIS VARIABLE USED.	The user entered a Y-axis variable name which was not in the Dictionary.	Enter a legal name.	7
ILLEGAL COMBINATION OF Y-AXIS VARIABLE AND QUALIFIER VALUE. PROGRAM WILL ALLOW YOU n(k2) MORE TRIES AT A LEGAL COMBINATION.	The user entered a combination of Y-axis variable, X-axis variable and qualifier which was not found in the Dictionary.	Try again to enter a legal combination. You have a maximum of 3 tries.	8
YOU HAVE FAILED TO REQUEST A VALID REPORT. THE ENTRIES MADE FOR THIS REPORT WILL BE IGNORED. THE ENTRIES YOU HAVE MADE ARE AS FOLLOWS:	On his third try, the user was unable to enter a legal combination of Y-axis variable name, X-axis variable name, and qualifier.	The program lists the entries made by the user so that he can identify his error. Then the user is asked if he wishes to enter additional reports.	9
YOUR SELECTION WILL CAUSE A TABLE TO BE PRODUCED BY THE PRINTER. NO PLOTS WILL BE PRODUCED.	The user has specified the output format to be a table.	This message is informative only and does not indicate an error by the user or that the user should take any action.	10

Table 2. (Part 2 of 2)

Table 2. (Part 2 of 2)

Message	Interpretation	User Action	Example in Figure 7
<p>AT LINE 0850 FILE 30 NOT IN APT. ACCESS CALLED. FUNCTION?</p>	<p>this is a TSS message which occurs when the user does not attach file "30" when he begins execution of RKEDIT.</p>	<p>In response to "FUNCTION?", the user should enter: AF, catalog-file string for file pointers file "30". The TSS will again ask "FUNCTION?". The user should enter a null response because there are no more files to be attached.</p>	3
<p>...PROG. L1 (ERR 032) ILLEGAL CHAR IN DATA BELOW. ERROR IN COLUMN OF DATA. TREAT ILLEGAL CHAR AS ZERO.</p>	<p>The user has entered alpha characters for an entry which should have been numeric.</p>	<p>Continue giving responses to the interactive program but ask for a list of commands at the end to check for errors.</p>	5

```

READY
*run
NOTIFY COMPUTER OPERATIONS TO OBTAIN ALL NECESSARY MAGNETIC TAPES FROM
THE TAPE LIBRARY.
REMTX IS EXECUTING - PLEASE RESPOND WHEN = APPEARS.
ENTER CATALOG-FILE STRING FOR THE DICTIONARY, 36 CHARACTERS MAXIMUM.
=674idp00/csc/dixt
ILLEGAL CATALOG-FILE STRING
ENTER CATALOG-FILE STRING FOR THE DICTIONARY, 36 CHARACTERS MAXIMUM.
=674idp00/csc/dict
ENTER CATALOG-FILE STRING FOR THE COMMANDS FILE, 36 CHARACTERS MAXIMUM.
=674idp00/csc/rxcom
ENTER INPUT SOURCE "T", "TAPE", "D", OR "DISK"
=tape
ILLEGAL ENTRY
ENTER INPUT SOURCE "T", "TAPE", "D", OR "DISK"
=tape
ENTER TAPE NUMBER OF INPUT SOURCE - 5 INTEGER DIGITS
=20550
CATALOG-FILE STRING FOR THE DICTIONARY = 674IDP00/CSC/DICT
CATALOG-FILE STRING FOR THE COMMANDS FILE = 674IDP00/CSC/RXCOM

INPUT SOURCE = TAPE
TAPE NUMBER OF INPUT SOURCE = 20550
DO YOU WISH TO CHANGE ANY CATALOG FILE NAMES - 'YES' OR 'NO'
=no
at line # 850 not in aft. access called-
file 30
FUNCTION? af,674idp00/csc/hfnew"30"
SUCCESSFUL.
FUNCTION?

```

①

②

③

Figure 7. Illustration of Exception Messages Occurring During Execution of RXEDIT (Part 1 of 7)

DO YOU WISH TO LOAD THE COMMANDS FILE - 'YES' OR 'NO'
 =yes
 ENTER LINE 1 OF REPORT TITLE - 36 CHARACTERS MAXIMUM.
 =red weapons type 1,2, and 3
 ENTER LINE 2 OF REPORT TITLE - 36 CHARACTERS MAXIMUM.
 =which are lost each day of battle
 ENTER LINE 3 OF REPORT TITLE - 36 CHARACTERS MAXIMUM.
 =report date 27 dec 76
 ENTER X-AXIS TITLE - 36 CHARACTERS MAXIMUM.
 =
 ENTER Y-AXIS TITLE 1, 36 CHARACTERS MAXIMUM. IF YOU
 HAVE ALREADY TRANSMITTED YOUR LAST Y-AXIS TITLE, ENTER ONE BLANK
 AND EXECUTE A CARRIAGE RETURN.
 =
 THE FIRST Y-AXIS TITLE CANNOT BE BLANK.
 ENTER Y-AXIS TITLE 1, 36 CHARACTERS MAXIMUM. IF YOU
 HAVE ALREADY TRANSMITTED YOUR LAST Y-AXIS TITLE, ENTER ONE BLANK
 AND EXECUTE A CARRIAGE RETURN.
 =weapons type 1
 ENTER Y-AXIS TITLE 2, 36 CHARACTERS MAXIMUM. IF YOU
 HAVE ALREADY TRANSMITTED YOUR LAST Y-AXIS TITLE, ENTER ONE BLANK
 AND EXECUTE A CARRIAGE RETURN.
 =weapons type 2
 ENTER Y-AXIS TITLE 3, 36 CHARACTERS MAXIMUM. IF YOU
 HAVE ALREADY TRANSMITTED YOUR LAST Y-AXIS TITLE, ENTER ONE BLANK
 AND EXECUTE A CARRIAGE RETURN.
 =weapons type 3
 ENTER Y-AXIS TITLE 4, 36 CHARACTERS MAXIMUM. IF YOU
 HAVE ALREADY TRANSMITTED YOUR LAST Y-AXIS TITLE, ENTER ONE BLANK
 AND EXECUTE A CARRIAGE RETURN.
 =

④

Figure 7. (Part 2 of 7)

```

ENTER X-AXIS SCALE (UPPER END)--'D', 'DEFAULT', OR A POSITIVE INTEGER,
WHICH IS LESS THAN OR EQUAL TO 100.
=10-
***PROG. L# (ERR #32)
..... 1510
ILLEGAL CHAR IN DATA BELOW
ERROR IN COLUMN 6 OF
10-
TREAT ILLEGAL CHAR AS ZERO
ENTER Y-AXIS SCALE (LOWER END), AN INTEGER WHICH IS GREATER THAN OR
EQUAL TO ZERO AND LESS THAN ONE MILLION.
=0
ENTER Y-AXIS SCALE (UPPER END), AN INTEGER
GREATER THAN ZERO AND LESS THAN ONE MILLION.
=10000
ENTER X-AXIS VARIABLE NAME, 'DAY' OR 'INDEX'.
=index
ILLEGAL X-AXIS VARIABLE USED.
ENTER X-AXIS VARIABLE NAME, 'DAY' OR 'INDEX'.
=index
ENTER Y-AXIS VARIABLE NAME 1
=rw01
ENTER QUALIFIER VALUE 1. IF X-AXIS VARIABLE IS 'DAY',
ENTER '0' FOR TOTALS OR A POSITIVE INTEGER TO SET AN INDEX VALUE;
IF X-AXIS VARIABLE IS 'INDEX', ENTER A POSITIVE INTEGER TO SPECIFY
THE DAY NUMBER. THE VALUE MUST BE LESS THAN OR EQUAL TO 100.
=1
ILLEGAL Y-AXIS VARIABLE USED.
ENTER Y-AXIS VARIABLE NAME 1
=rw11
ENTER QUALIFIER VALUE 1. IF X-AXIS VARIABLE IS 'DAY',
ENTER '0' FOR TOTALS OR A POSITIVE INTEGER TO SET AN INDEX VALUE;

```

⑤

⑥

⑦

Figure 7. (Part 3 of 7)

7 (Cont'd) IF X-AXIS VARIABLE IS 'INDEX', ENTER A POSITIVE INTEGER TO SPECIFY THE DAY NUMBER. THE VALUE MUST BE LESS THAN OR EQUAL TO 100.
 =0
 ILLEGAL COMBINATION OF Y-AXIS VARIABLE, X-AXIS VARIABLE, AND QUALIFIER VALUE. PROGRAM WILL ALLOW YOU 2 MORE TRIES AT A LEGAL COMBINATION.
 ENTER X-AXIS VARIABLE NAME, 'DAY' OR 'INDEX'.
 =index
 ENTER Y-AXIS VARIABLE NAME 1
 =rwl
 ENTER QUALIFIER VALUE 1. IF X-AXIS VARIABLE IS 'DAY',
 ENTER '0' FOR TOTALS OR A POSITIVE INTEGER TO SET AN INDEX VALUE;
 IF X-AXIS VARIABLE IS 'INDEX', ENTER A POSITIVE INTEGER TO SPECIFY THE DAY NUMBER. THE VALUE MUST BE LESS THAN OR EQUAL TO 100.
 =0

8 ILLEGAL COMBINATION OF Y-AXIS VARIABLE, X-AXIS VARIABLE, AND QUALIFIER VALUE. PROGRAM WILL ALLOW YOU 1 MORE TRIES AT A LEGAL COMBINATION.
 ENTER X-AXIS VARIABLE NAME, 'DAY' OR 'INDEX'.
 =index
 ENTER Y-AXIS VARIABLE NAME 1
 =rwl
 ENTER QUALIFIER VALUE 1. IF X-AXIS VARIABLE IS 'DAY',
 ENTER '0' FOR TOTALS OR A POSITIVE INTEGER TO SET AN INDEX VALUE;
 IF X-AXIS VARIABLE IS 'INDEX', ENTER A POSITIVE INTEGER TO SPECIFY THE DAY NUMBER. THE VALUE MUST BE LESS THAN OR EQUAL TO 100.
 =0

9 YOU HAVE FAILED TO REQUEST A VALID REPORT.
 THE ENTRIES MADE FOR THIS REPORT WILL BE IGNORED.
 THE ENTRIES YOU HAVE MADE ARE AS FOLLOWS -
 LINE 1 OF REPORT TITLE = RED WEAPONS TYPE 1, 2, AND 3
 LINE 2 OF REPORT TITLE = WHICH ARE LOST EACH DAY OF BATTLE
 LINE 3 OF REPORT TITLE = REPORT DATE 27 DEC 76
 X-AXIS TITLE = DAY
 Y-AXIS TITLE 1 = WEAPONS TYPE 1

Figure 7. (Part 4 of 7)

Y-AXIS TITLE 2 = WEAPONS TYPE 2
 Y-AXIS TITLE 3 = WEAPONS TYPE 3
 X-AXIS SCALE (UPPER END) = 10-
 Y-AXIS SCALE (LOWER END) = 0
 Y-AXIS SCALE (UPPER END) = 10000
 X-AXIS VARIABLE NAME = INDEX
 Y-AXIS VARIABLE NAME 1 AND ITS QUALIFIER = RWL1 0
 Y-AXIS VARIABLE NAME 2 AND ITS QUALIFIER = 000000 000000
 Y-AXIS VARIABLE NAME 3 AND ITS QUALIFIER = 000000 000000
 ADDITIONAL REPORTS DESIRED? - "YES" OR "NO"
 =yes
 ENTER LINE 1 OF REPORT TITLE - 36 CHARACTERS MAXIMUM.
 =total blue people type three
 ENTER LINE 2 OF REPORT TITLE - 36 CHARACTERS MAXIMUM.
 =in all sectors of region 2
 ENTER LINE 3 OF REPORT TITLE - 36 CHARACTERS MAXIMUM.
 =report date 27 dec 76
 ENTER X-AXIS TITLE - 36 CHARACTERS MAXIMUM.
 =
 ENTER Y-AXIS TITLE 1, 36 CHARACTERS MAXIMUM. IF YOU
 HAVE ALREADY TRANSMITTED YOUR LAST Y-AXIS TITLE, ENTER ONE BLANK
 AND EXECUTE A CARRIAGE RETURN.
 =people
 ENTER Y-AXIS TITLE 2, 36 CHARACTERS MAXIMUM. IF YOU
 HAVE ALREADY TRANSMITTED YOUR LAST Y-AXIS TITLE, ENTER ONE BLANK
 AND EXECUTE A CARRIAGE RETURN.
 =
 ENTER X-AXIS SCALE (UPPER END)--'D', 'DEFAULT', OR A POSITIVE INTEGER,
 WHICH IS LESS THAN OR EQUAL TO 100.
 =20
 ENTER Y-AXIS SCALE (LOWER END), AN INTEGER WHICH IS GREATER THAN OR
 EQUAL TO ZERO AND LESS THAN ONE MILLION.
 =10000

Figure 7. (Part 5 of 7)

```

ENTER Y-AXIS SCALE (UPPER END), AN INTEGER
GREATER THAN ZERO AND LESS THAN ONE MILLION.
=70000
ENTER X-AXIS VARIABLE NAME, 'DAY' OR 'INDEX'.
=day
ENTER Y-AXIS VARIABLE NAME 1
=bps3
ENTER QUALIFIER VALUE 1. IF X-AXIS VARIABLE IS 'DAY',
ENTER '0' FOR TOTALS OR A POSITIVE INTEGER TO SET AN INDEX VALUE;
IF X-AXIS VARIABLE IS 'INDEX', ENTER A POSITIVE INTEGER TO SPECIFY
THE DAY NUMBER. THE VALUE MUST BE LESS THAN OR EQUAL TO 100.
=2
ENTER OUTPUT FORMAT - "H", "HISTOGRAM", "G", "GRAPH", "B", "BOTH", "TABLE"
=table
YOUR SELECTION WILL CAUSE A TABLE TO BE PRODUCED BY THE PRINTER. NO
PLOTS WILL BE PRODUCED.
DISPLAY OF REPORT DESIRED - 'YES' OR 'NO'
=no
DO YOU WISH TO KEEP THIS REPORT? -- 'YES' OR 'NO'
=no
ADDITIONAL REPORTS DESIRED? - "YES" OR "NO"
=no
REMTX EDIT PROGRAM HAS COMPLETED

HAVE A NICE DAY

*
*run
NOTIFY COMPUTER OPERATIONS TO OBTAIN ALL NECESSARY MAGNETIC TAPES FROM
THE TAPE LIBRARY.
REMTX IS EXECUTING - PLEASE RESPOND WHEN = APPEARS.

```

10

Figure 7. (Part 6 of 7)

```

ENTER CATALOG-FILE STRING FOR THE DICTIONARY, 36 CHARACTERS MAXIMUM.
=674idp00/csc/dict
ENTER CATALOG-FILE STRING FOR THE COMMANDS FILE, 36 CHARACTERS MAXIMUM.
=674idp00/csc/rxcom
ENTER INPUT SOURCE "T", "TAPE", "D", OR "DISK"
=tape
ENTER TAPE NUMBER OF INPUT SOURCE - 5 INTEGER DIGITS
=20550
CATALOG-FILE STRING FOR THE DICTIONARY = 674IDP00/CSC/DICT
CATALOG-FILE STRING FOR THE COMMANDS FILE = 674IDP00/CSC/RXCOM

INPUT SOURCE = TAPE
TAPE NUMBER OF INPUT SOURCE = 20550
DO YOU WISH TO CHANGE ANY CATALOG FILE NAMES - 'YES' OR 'NO'
=no
at line # 850
file 30 not in aft. access called-

FUNCTION? af,674idp00/csc/hfnew"30"
SUCCESSFUL.

FUNCTION?
DO YOU WISH TO LOAD THE COMMANDS FILE - 'YES' OR 'NO'
=yes
⑪ { COMPLETE DICTIONARY IS NOT IN STORAGE.
    THERE IS ONLY ROOM FOR 100 ITEMS. PROGRAM TERMINATED.

```

RESAVE file name for the RXPROC JCL file.

Example: RESAVE RXJCL

*
—

The user is now ready to execute RXPROC through the following dialog.

RUN

CARD FORMAT, DISPOSITION

N

SNUMB ddddT

The user must write down the SNUMB in order to claim his job at the control desk upon completion of REMTEX execution. When the user claims his output, he also may submit the plotter tape produced by RXPROC, if any, to the CalComp 7000 plotter through the normal job submission procedure.

*
—

BYE

LINE TERMINATED

The user has completed his dialog and his terminal has been disconnected from the H-6080 computer.

2.3 System Maintenance

This subsection describes the maintenance procedures for the RXEDIT and RXPROC programs and the REMTEX files.

2.3.1 RXEDIT Program Maintenance. The REMTEX Editor (RXEDIT) is coded in HIS TSS FORTRAN using structured programming. Each interactive sequence is coded according to the following logic:

- a. Display prompting statement.
- b. Read the user reply.

- c. Test the reply for validity; if invalid, display an error message and display the prompting statement again.
- d. Display the next pertinent prompting statement.

The format statements used to display the prompting statements contain English language phrases, making the RXEDIT program self-documenting and, consequently, easy to follow.

RXEDIT contains six major sections. The first section contains a set of inquiries about the files and tapes to be used in RXEDIT and RXPROC. The keyed replies are loaded onto the File Pointers file for input to RXPROC. The second section reads the items from the Dictionary file into core memory. The third section is a set of inquiries regarding the report which the user is requesting. If the user so elects, a display of this report is given by the fourth section of code. If the user wishes to implement this report, the fifth section loads the keyed requests onto the Commands file. Sections 3, 4, and 5 are executed for each report requested by the user. Finally, the files are detached and the program is terminated in the sixth section.

RXEDIT calls a time-sharing FORTRAN subroutine called ADJUST. This routine takes a six-character word which is left justified and right justifies its contents. This is accomplished by shifting characters to the right, one character for each trailing blank in the original word. The right justified word will have leading blanks if any shifting was necessary. Subroutine ADJUST is on the same file as RXEDIT.

To alter the RXEDIT code, the maintenance programmer must log on to a terminal and follow the steps presented below:

SYSTEM?

YFORT

OLD OR NEW

OLD catalog-file string for RXEDIT source code

Example: OLD 674IDP00/CSC/RXEDIT.

*

—

LIST mmmm-nnnn

The user may elect to display portions of the RXEDIT code on his terminal. In this case, lines mmmm through nnnn will be displayed. (The initial alter number precedes the hyphen and the final alter number follows it.)

mmmm code

.
.
.

nnnn code

*
—

mmmm code

The user inputs an alter number followed by TSS FORTRAN code. If the alter number does not exist, the number and its line of code are entered in proper numerical sequence into RXEDIT. If the alter number does exist, the new line of code replaces the old line of code.

DELE mmmm-nnnn

To delete lines of code, the user enters the word DELE, followed by the alter numbers of those lines to be deleted.

*
—

RESA file name for RXEDIT source code

The updated version of RXEDIT will now replace the old version.

Example: RESA RXEDIT.

RXEDIT SAVED

*
—

BYE

This maintenance technique is applicable to all files that are TSS format.

Whenever the RXEDIT routine is altered, it should be recompiled and a new H* file created. This is accomplished on the terminal, beginning at the system level, through the following dialog:

SYSTEM?

YFORT

OLD OR NEW

OLD catalog-file string for the revised RXEDIT source code

READY

*
—

RUN = catalog-file string for the RXEDIT H* # catalog-file string for the File Pointers file"30".

The next time the user wishes to run RXEDIT, he may do so from the newly created H* file.

2.3.2 RXPROC Program Maintenance. The RXPROC program consists of the main routine, three FORTRAN subroutines, PLOTTR, NEWAX, and ERRMSG, and one GMAP subroutine, MORE. Appendix B contains a complete source code listing of RXPROC. Each of these routines resides on a separate source code file and object code file. The JCL setup, shown in table 3, will allow a maintenance programmer to alter the RXPROC routines.

2.3.2.1 RXPROC Main Routine Logic. The source code of the main routine of RXPROC is divided into nine sections, lettered A through I, by comment cards. The RXPROC logic will be discussed by section.

2.3.2.1.1 Section A. This section performs the initial, nonrepetitive housekeeping functions. The plotter tape is opened and assigned to file code 25 through the CalComp Basic Software Package Subroutine PLOTS. The File Pointers file (file code 30) is opened and searched sequentially. When the key DIC is found, the Dictionary file is attached via the catalog-file string and assigned to file code 10. When the key COM is found, the Commands file is attached via the

Table 3. JCL for Altering RXPROC Routines (Part 1 of 3)

```

$      SNUMB   Installation-dependent
$      IDENT   Installation-dependent
$      USERID Installation-dependent
$      OPTION  FORTRAN
$      FORTY   DECK,COMDK
$      PRMFL   S*,R,S,catalog-file string for RXPROC source
              code file
$      PRMFL   C*,W,S,catalog-file string for RXPROC object
              code file
$      FILE    K*,X1S,10L
$      UPDATE  LIST
      .
      .
      .
      alter deck for RXPROC
      .
      .
$      CONVERT NSPIN
$      FILE    IN,X1S
$      PRMFL   OT,W,S,catalog-file string for RXPROC source
              code file
$      FORTY   DECK,COMDK
$      PRMFL   S*,R,S,catalog-file string for PLOTTR source
              code file
$      PRMFL   C*,W,S,catalog-file string for PLOTTR object
              code file
$      FILE    K*,X1S,10L
$      UPDATE  LIST
      .
      .
      .
      alter deck for PLOTTR
      .
      .
$      CONVERT NSPIN
$      FILE    IN,X1S
$      PRMFL   OT,W,S,catalog-file string for PLOTTR source
              code file
$      FORTY   DECK,COMDK
$      PRMFL   S*,R,S,catalog-file string for NEWAX source
              code file

```

Table 3. (Part 2 of 3)

```

$      PRMFL  C*,W,S,catalog-file string for NEWAX object
        code file
$      FILE   K*,XLS,10L
$      UPDATE LIST
        .
        .
        .
alter deck for NEWAX
        .
        .
        .
$      CONVERT NSPIN
$      FILE   IN,XLS
$      PRMFL  OT,W,S,catalog-file string for NEWAX source
        code file
$      FORTY  DECK,COMDK
$      PRMFL  S*,R,S,catalog-file string for ERRMSG source
        code file
$      PRMFL  C*,W,S,catalog-file string for ERRMSG object
        code file
$      FILE   K*,XLS,10L
$      UPDATE LIST
        .
        .
        .
alter deck for ERRMSG
        .
        .
        .
$      CONVERT NSPIN
$      FILE   IN,XLS
$      PRMFL  OT,W,S,catalog-file string for ERRMSG source
        code file
$      GMAP   DECK,COMDK
$      PRMFL  G*,R,S,catalog-file string for MORE source
        code file
$      PRMFL  C*,W,S,catalog-file string for MORE object
        code file
$      FILE   K*,XLS,10L
$      UPDATE LIST
        .
        .
        .

```

Table 3. (Part 3 of 3)

```
alter deck for MORE
      .
      .
$      CONVERT NSPIN
$      FILE     IN,X1S
$      PRMFL    OT,W,S,catalog-file string for MORE source
              code file
$      ENDJOB
```

catalog-file string and assigned to file code 15. If the key CFS is found, the Summary Data file is attached via the catalog-file string and assigned to file code 20. If the key TPN is found, the Summary Data file is attached via the tape reel number and Subroutine MORE and is assigned to file code 20. If no tape drive is available, RXPROC will notify the operator and RXPROC will be placed in HOLD status. When all of the records in the File Pointers file have been read and processed, RXPROC reads the Dictionary data. The Y-variable information in each record is stored in the following arrays:

DNAM(n,1) - Y variable name
DNAM(n,2) - legal X variable name
DQUAL - legal Y variable qualifier (indexed or nonindexed)
DFORM - index of the format statement for decoding the record
DTYP - data type, 0 = integer, 1 = real
DTAB - record key value
DENT - Y variable data field location within the record.

When the Y variable name ZZZZZZ is read in, all of the legal Y variables have been read. The next record contains the number of format statements on the Dictionary file and the remaining records contain the format statements, which are read in and stored in array FORM.

2.3.2.1.2 Section B. This section performs recurring house-keeping. All arrays and variables are reinitialized for each report.

2.3.2.1.3 Section C. This section reads the commands that constitute a report request. Each command is interpreted and the appropriate action is performed. When the End-of-Report command (ERP) is read, RXPROC transfers to Section D to process the report.

2.3.2.1.4 Section D. This section compares the variable names requested for the report with the legal variable combinations specified in the Dictionary. If any requested variable is illegal, the report will be omitted.

2.3.2.1.5 Section E. This section reads the Summary Data file and compares each key against the group of variables that were requested for this report. When a match is found, the data is retrieved from its data field within the record and stored in the YDAT data array. The location of the data field and the form of the data (real or integer) are determined by Dictionary information. Also, the maximum value for X is obtained and saved. When the Summary Data file has been searched completely, RXPROC transfers to Section F.

2.3.2.1.6 Section F. This section sets the maximum X and Y scale values (user default) or, alternatively, tests the X and Y values against the scales requested by the user. If this section sets the maximum Y scale value, the value will be one digit followed by one or more zeros, such as 6000. Also, the Summary Data file is rewound to position it for the next report, if any.

2.3.2.1.7 Section G. This section converts the report values in array YDAT to printer positions and stores them in array PTDATA.

2.3.2.1.8 Section H. This section sets up those reports to be plotted, if any, and calls Subroutine PLOTTR to place the report onto the plotter tape.

2.3.2.1.9 Section I. This section sets up and outputs print page images. First the Y scale increments are set. If the report format is a histogram, a Y-axis label is set from the user request. Then each X scale value is set, the corresponding bar is constructed from array PTDATA and, optionally, the Y data value is placed on top of the bar. If over 16 X values are required, one printer page of the histogram will be output and a new print page will be initiated for the remaining bars.

If the report format is a point graph, the Y-axis labels will be set from the user request. Next, the X scale increments will be set. Finally, the data points will be extracted from array PTDATA and placed into the appropriate positions within the print page image, and the point graph will be written to the printer.

If the report format is a table, no histogram or point graph will be output. First, the report title, page number, and column headers are set up. Next, the data lines are set up. The first column contains the X data values in ascending

increments. The remaining columns contain as many corresponding Y values as the user has requested. If the data values exceed the capacity of the printer page, the page will be output and the title and header information will be regenerated. When all requested report formats have been output, RXPROC will return to Section B to begin the next report.

2.3.2.2 PLOTTR Logic. Subroutine PLOTTR (NUM) is called by RXPROC for one of three purposes: (1) to write commands on the plotter tape for drawing a line graph, (2) to write commands on the plotter tape for drawing a histogram, or (3) to close out the plotter tape. The input argument NUM is used with (1) and (2) to indicate the index of the plot whose commands are being written to tape. This value is used to position the pen on the CalComp plotter. When PLOTTR is called to close out the plotter tape, NUM is set to zero.

PLOTTR uses two routines which are part of the CalComp Basic Software Package, SYMBOL and LINE. The SYMBOL routine produces annotation, and LINE produces a line plot of pairs of data points. PLOTTR also uses the routine BAR, which draws bars with or without hatching. BAR is part of the CalComp Graphic Functional Software Package. The CalComp Basic Software Package also contains AXIS which plots an axis and writes the scale values in a real format with two decimal places. However, AXIS does not meet the needs of this program, so NEWAX was written to plot an axis with scale values in integer format.

The first major section of PLOTTR sets up and writes the line graph commands to the plotter tape. Using the scaling range determined in RXPROC, Subroutine NEWAX is called to draw the X and Y axes, with their associated scale values in integer format. Through calls to Subroutine SYMBOL, each axis label and the report titles are printed. The Y axis may have up to six titles, one for each line graph to be drawn on the same pair of axes. For each line graph, the data points are loaded into the arrays XPLT and YPLT and Subroutine LINE is called to draw the line graph connecting the data points. Finally the plotter pen is repositioned for the next plot and a new origin is established.

The second section of PLOTTR sets up and writes the histogram commands to the plotter tape. The Y coordinate data values are loaded into the Y array. As in the first section, NEWAX and SYMBOL are called to plot the axes and label the plot. The width of the bars is determined from the length of the X-axis and the number of data points. Then, for each X

value, the height of the bar is determined and Subroutine BAR is called to draw the bar.

The third section calls Subroutine PLOT to close out the plotter tape. PLOT is also part of the CalComp Basic Software Package.

2.3.2.3 NEWAX Logic. Subroutine NEWAX (XPAGE, YPAGE, AXLEN, ANGLE, FIRSTV, DELTAV) plots an axis with a tick mark every inch and writes the scale values in integer format. The arguments of NEWAX are as follows:

XPAGE, YPAGE	are the coordinates, in inches, of the axis line's starting point.
AXLEN	is the length of the axis line, in whole inches.
ANGLE	is the angle, in degrees (either 0. or 90.) at which the axis is drawn. The value is 0° for the X axis and 90° for the Y axis.
FIRSTV	is the starting value, in real whole numbers, which appears at the first tick mark on the axis. This number and each scale value along the axis is drawn in integer format. Since the digit size is .1 inches (10 characters per inch) and since a scale value appears every inch, the scale values must be no more than 8 digits plus a sign.
DELTAV	represents the number of data units per inch of the axis. This value, which must be a real whole number, is added to FIRSTV for each succeeding 1-inch division along the scale.

2.3.2.4 ERRMSG Logic. Subroutine ERRMSG outputs error messages for RXPROC and performs relevant housekeeping. The error messages and their meanings are presented in table 4. Depending on the nature of the error that was encountered, ERRMSG will (1) return control to RXPROC for continued processing, (2) position the Commands file to the next report

Table 4. Exception Messages Produced by Subroutine ERRMSG (Part 1 of 2)

Error Message Number	Message	Program Action	User Action
<1 or >15			
1	WARNING - ERRMSG ARGUMENT IS OUT OF RANGE - CHECK RXPROC.	Continues processing.	Check RXPROC for call to ERRMSG with a value for IERR outside of range.
2	WARNING - ILLEGAL COMMAND ON COMMANDS FILE.	Reads next command on Commands file. Program terminates.	Check out Commands file to determine reason for illegal command. Rerun RXPROC. If error still present, recreate Commands file.
3	IRRECOVERABLE I/O ERROR ON COMMANDS FILE. PROGRAM TERMINATED. NUMBER OF DICTIONARY ITEMS EXCEEDS STORAGE SPACE. PROGRAM TERMINATED.	Program terminates.	Increase parameter NDP in interactive program and in RXPROC to accommodate all items of the Dictionary.
4	WARNING-SCALE RANGE TOO SMALL. DEFAULT RANGE USED.	Sets scale range to default range and continues processing.	Review requirements of scale ranges.
5	WARNING-TWO OR MORE DATA POINTS OCCUPY THE SAME POSITION.	Continues processing.	Provided to alert the user that one or more data points will be suppressed on the online printer point graph.
6	YVAR, XVAR, EQUAL COMBINATION IS NOT LISTED IN THE DICTIONARY. REPORT NUMBER n HAS BEEN OMITTED. (Printed by RXPROC.)	Advances the Commands file to the next report.	Check this combination against the Dictionary entries.
7	UNABLE TO ATTACH SPECIFIED FERM FILE. PROGRAM TERMINATED.	Program terminates.	Determine the reason that the file was not available. The file may be nonexistent or busy.
8	WARNING-THE NUMBER OF Y VARIABLE NAMES EXCEEDS THE NUMBER OF Y TITLES. EXTRA Y VARIABLE NAMES HAVE BEEN IGNORED.	Reads next command on Commands file.	Check Commands file for extra Y variable names.
9	WORD DAY NOT ENCOUNTERED ON FILE 20. PROGRAM TERMINATED.	Program terminates.	Check the Summary Data file to ensure that it has the appropriate format.

Table 4. (Part 2 of 2)

<u>Error Message Number</u>	<u>Message</u>	<u>Program Action</u>	<u>User Action</u>
10	IRRECOVERABLE I/O ERROR ON DATA FILE 20. PROGRAM TERMINATED.	Program terminates.	Return program. If error still present, recreate Summary Data file.
11	REPORT NUMBER n HAS BEEN OMITTED. APPROPRIATE VALUES FOR DAY NOT FOUND ON DATA FILE 20.	Advances the Commands file to the next report.	Compare value for DAY index against Summary Data file. This message usually occurs when the reported day is greater than the number of days covered by the summary data.
12	WARNING-ILLEGAL COMMAND ON FILE POINTERS FILE.	Returns to read next record on File Pointers file.	Check the File Pointers file for invalid commands.
13	IRRECOVERABLE I/O ERROR ON DICTIONARY FILE. PROGRAM TERMINATED.	Program terminates.	Return program. If error still exists, recreate Dictionary file.
14	ABNORMAL ENDING FOR ITEMS IN THE DICTIONARY. PROGRAM TERMINATED.	Program terminates.	Check Dictionary file for "ZZZZZ" card terminating Dictionary items.
15	INCORRECT ENTRY OF DATA DESCRIBING FORMATS ASSOCIATED WITH THE DICTIONARY ITEMS. PROGRAM TERMINATED.	Program terminates.	Check Dictionary file for validity of data describing the formats.

request and return to RXPROC for processing of the next report request, or (3) return to RXPROC for immediate termination.

2.3.2.5 MORE Logic. Subroutine MORE is a GMAP routine that uses the MME GEMORE instruction to attach a magnetic tape. MORE is called once by RXPROC if the Summary Data file is on tape. The first argument points to a descriptive array, the second argument gives the number of entries in the array (always 1 for RXPROC), and the third argument is a status word. An entry in the descriptive array (ITAPE for RXPROC) consists of a word pair. The second word contains the reel number in five characters, left justified. The reel number is obtained from the File Pointers file. The first word is BCD and contains media type (3 = 9-track magnetic tape), left justified, and the file code (20 for RXPROC), right justified. Element ITAPE(1) is always preset to 300020 in RXPROC.

2.3.3 REMTEX File Maintenance. The REMTEX file formats consist of TSS files and BCD files.

2.3.3.1 TSS Files. In addition to the RXEDIT source code file, which was discussed in subsection 2.3.1, a JCL file exists in TSS format that allows a user to initiate a run from his remote terminal. Table 5 lists typical REMTEX JCL. The contents of a TSS JCL file can be altered and saved through the same technique that is employed for the RXEDIT source code file (see subsection 2.3.1).

2.3.3.2 BCD Files. REMTEX employs the following BCD files:

<u>File Code</u>	<u>File Name</u>	<u>Loaded By</u>
10	Dictionary	User
15	Commands	RXEDIT
20	Summary Data	User's program
25	Plotter Tape	RXPROC
30	File Pointers	RXEDIT

The Plotter Tape is an intermediate file produced on the H-6000 and used as input to the CalComp 7000. Since it is a work file, no further discussion of it will be presented. All of the other BCD files may be generated by the following code:

Table 5. Example of a JCL File for Executing
RXPROC From the Terminal

```
10$:IDENT:4513,RXPRO,315,JOHN ROLEY,674,10
20$:OPTION:FORTRAN
30$:LOWLOAD
40$:USE:.GTLIT
50$:SELECT:674IDP00/CSC/RXPROCOB
60$:SELECT:674IDP00/CSC/ERRMSG0B
70$:SELECT:674IDP00/CSC/PLOTTROB
80$:SELECT:674IDP00/CSC/NEWAXOB
90$:SELECT:674IDP00/CSC/MOREOB
100$:LIBRARY:PL
110$:EXECUTE:DUMP
120$:LIMITS:25,38K,-4K,10K
130$:PRMFL:PL,R,R,LIBRARY/PLOTLIBN
140$:PRMFL:30,R,S,674IDP00/CSC/RXFPT
150$:TAPE:25,X3D,,99999,,PLOTTER-TAPE,,DEN8
160$:FFILE:25,FXLNG/80,BUFSIZ/81
170$:ENDJOB
```

```

$      SNUMB  installation-dependent
$      IDENT  installation-dependent
$      USERID installation-dependent
$      CONVERT NSPIN
$      DATA  IN,,COPY
.
.
.      punched card deck containing data to be loaded
.
.
$      ENDCOPY
$      PRMFL  OT,W,S,catalog-file string for file to be loaded
$      ENDJOB

```

Normally, only the Dictionary file will be loaded as described above. However, in the following discussion, the data fields will be specified for all of the BCD files, enabling a user to set up a punched card deck representing each file.

2.3.3.2.1 Dictionary File and Summary Data File. Since these two files are interrelated, they will be discussed together.

The REMTEX package is independent of the type of summary data used as input. However, this data must be in the appropriate format and a Dictionary must be developed to guide REMTEX in reading the data. Table 6 partially lists the contents of a Summary Data file.

The data must meet the following requirements:

- a. Groups of data are divided into sections by a leading card with DAY in the columns 1 through 3, blank in column 4, day index in columns 5 and 6, and a 2-digit program code in columns 7 and 8. The day index in I2 format indicates the day of battle with which the following data are associated. The program code in I2 format is any value the user wishes to specify to identify his program. (See ① of table 6.)
- b. The data cards which follow the DAY card are formatted so that columns 1 through 4 contain a key, columns 5 and 6 contain an index value, and columns 7 through 132 contain the actual

Table 6. Format for the Summary Data File

①	DAY	1	1				
	.						
②	1101	1	0.	315.41	309.22	0.	
	1101	2	0.	18.13	17.77	0.	
	1101	3	0.	9.06	8.89	0.	
	1101	4	0.	82.96	93.33	0.	
	1101	5	270.56	0.	0.	0.	
	1101	6	0.	155.97	175.47	132.35	
	1101	7	0.	5.06	5.69	3.91	
	1101	8	0.	10.33	11.62	8.25	
	1101	9	0.	57.12	64.26	0.	
	110110		811.68	0.	0.	0.	
	110111		0.	24.64	27.72	0.	
	110200		1082.24	678.68	713.97	144.52	
	1103	1	0.	3.99	2.99	0.	
	1103	2	0.	0.25	0.18	0.	
	1103	3	0.	0.09	0.07	0.	
	1103	4	0.	0.28	0.21	0.	
	1103	5	7.43	0.	0.	0.	
	1103	6	0.	6.51	4.88	4.07	
	1103	7	0.	0.22	0.17	0.13	
	1103	8	0.	0.43	0.32	0.26	
	1103	9	0.	1.35	1.01	0.	
	110310		4.63	0.	0.	0.	
	110311		0.	1.06	0.79	0.	
	110400		12.06	14.17	10.63	4.46	
	.						
	.						
	DAY	2	1				
	.						
	.						
	1101	1	0.	718.55	754.96	0.	
	1101	2	0.	43.06	45.35	0.	
	1101	3	0.	21.53	22.67	0.	
	1101	4	0.	189.58	228.61	0.	
	1101	5	507.93	0.	0.	0.	
	1101	6	0.	355.13	428.16	745.27	
	1101	7	0.	11.87	14.30	24.95	
	1101	8	0.	24.63	29.79	50.95	
	1101	9	0.	130.54	157.42	0.	
	110110		1590.17	0.	0.	0.	
	.						
	.						

data to be processed. (See ② of table 6.)
No two lines of data may be identical in
columns 1 through 6 within the same day.

Table 7 shows both a sample Summary table, which is produced by IDAGAM II for each day of battle, and the format of this data on the Summary Data file. In creating the Summary Data file by IDAGAM II, the first two digits of the key were used to denote the Summary table number from which the data was taken, in this case number 11. The table was then divided into four groups as indicated in table 7. The second two digits of the key were used to denote the group within the table from which the data were taken. For the key 1101, the index values of 1 through 11 in columns 5 and 6 denote the various missions. An index value of 0 is used for the key 1102 to denote that this data record is a set of totals; no other data records within this day will have the key 1102. The data which follow the index may be of any format but they must be contained in columns 7 through 132. Table 8 lists the data for the first day of battle as contained on a Summary Data file produced by IDAGAM II.

The Summary Data file format has been set up in this manner to allow the user flexibility in his choice of variables to be plotted. The user can elect to plot the days of battle versus any piece of data on the file. For example, he might plot day versus the number of CAS missions flown by aircraft type F-104. He might also choose to plot, for a specified day of battle, the 11 types of missions versus the number of aircraft of type F-104 which flew that mission. Both of these selections are possible when the data file is set up as illustrated. The user is also allowed to plot up to six point and line graphs on the same set of axes so he can observe the relationship between the various types of aircraft. For example, the user may plot mission versus aircraft types 1 through 6 on the same plot.

The REMTEX program, however, does restrict the user in that he must select either day or an index from columns 5 and 6 as his X-axis variable. The index for Summary table 11 denotes mission type, but for another table it might denote region, sector, weapon type, or any other value selected by the user.

For REMTEX to extract a data item from a record, information must be provided about the format to be used to decode that record, which data field contains the item, and whether the

Table 7. Comparison of an IDAGAM II Summary Table and Summary Data Created From It

File Created From Data in Summary Table 11

IDAGAM II Summary Table 11

TABLE S-11-1

CUMULATIVE SORTIES FLOWN AND AIRCRAFT KILLED -- BLUE

CUMULATIVE SORTIES FLOWN

MISSION	F-104	AIRCRAFT A-7	F-4C/D	F-111	1101 1	1101 2	1101 3	1101 4	1101 5	1101 6	1101 7	1101 8	1101 9	1101 10	1101 11	110200	TOTAL
CAS	0.	315.41	309.22	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
CASG	0.	18.13	17.77	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
CASS	0.	9.06	8.89	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
CASE	0.	82.96	93.33	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
BD	270.56	0.	0.	0.	270.56	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
ABA	0.	155.97	175.47	132.35	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
ABAG	0.	5.06	5.69	3.91	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
ABAS	0.	10.33	11.62	8.25	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
ABAE	0.	57.12	64.26	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
ABD	811.68	0.	0.	0.	811.68	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
IDR	0.	24.64	27.72	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
TOTAL	1082.2	678.7	714.0	144.5	1082.24	678.68	713.97	144.52									

CUMULATIVE AIRCRAFT KILLED

MISSION	F-104	AIRCRAFT A-7	F-4C/D	F-111	1103 1	1103 2	1103 3	1103 4	1103 5	1103 6	1103 7	1103 8	1103 9	1103 10	1103 11	110400	TOTAL
CAS	0.	3.99	2.99	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
CASG	0.	0.25	0.18	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
CASS	0.	0.09	0.07	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
CASE	0.	0.28	0.21	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
BD	7.43	0.	0.	0.	7.43	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
ABA	0.	6.51	4.88	4.07	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
ABAG	0.	0.22	0.17	0.13	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
ABAS	0.	0.43	0.32	0.26	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
ABAE	0.	1.35	1.01	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
ABD	4.63	0.	0.79	0.	4.63	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
IDR	0.	1.06	0.79	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
TOTAL	12.1	14.2	10.6	4.5	12.06	14.17	10.63	4.46									

Table 8. (Part 4 of 7)

0601 3 1	9521.56				
0601 4 1	6526.99				
0601 5 1	8401.00				
0601 6 1	380.00				
0601 7 1	990.77				
0601 8 1	540.76				
0601 9 1	949.00				
0601 10 1	474.00				
0602 00 100.	67.	67.	67.	66.	66.
0602 04 100.	67.	67.	67.	66.	66.
0602 1 0.	64.	64.	64.	66.	66.
0602 2 0.	121.	121.	121.	126.	126.
0602 3 0.	91.	91.	91.	94.	94.
0603 1 0.	0.	0.	0.	0.	0.
0603 2 0.	0.	0.	0.	0.	0.
0603 3 0.	0.	0.	0.	0.	0.
0604 00 93986.32					
0701 01 1	69434.				
0701 02 139800.					
0801 01 1	694756.				
0801 02 159440.					
9A01 1	239121.				
9A01 2	278641.				
9A01 3	280277.				
9A01 4	239134.				
9A01 5	239200.				
9A01 6	346367.				
9A01 7	458374.				
9A02 1	121200.				
9A02 2	222562.				
9A03 1	0.				
9A03 2	0.				
9A04 1	1465067.				
9A05 01 35764.					
9A06 01 1003200.					
9B01 1	516361.				
9B01 2	511476.				
9B01 3	514105.				
9B01 4	516361.				
9B01 5	516566.				
9B01 6	511212.				
9B01 7	512731.				
9B02 1	595341.				
9B02 2	590936.				
9B02 3	590851.				
9B03 1	0.				
9B03 2	1260.				
9B03 3	7340.				
9B04 1	2955251.				
9B05 01 34104.					
9B06 01 994216.					
1001 01 150.00	114.00				
1001 02 150.00	114.00				
1001 03 0.	0.				
1002 01 69.00	776.00	158.00			

Table 8. (Part 5 of 7)

100202	0.00	275.00	116.00						
100203	0.								
100301	0.								
100401	0.			311.41	305.22	0.			
1101 1	0.			16.13	17.77	0.			
1101 2	0.			64.06	5.09	0.			
1101 3	0.			82.96	53.53	0.			
1101 4	0.			0.	0.	0.			
1101 5	270.56			158.97	171.47	112.32			
1101 6	0.			15.06	5.09	3.91			
1101 7	0.			10.33	11.62	0.25			
1101 8	0.			57.12	64.26	0.			
1101 9	0.			0.	0.	0.			
110110	0.11.68			24.64	27.72	0.			
110111	0.			678.68	713.97	144.52			
110200	1082.24			3.99	2.99	0.			
1103 1	0.			0.25	0.16	0.			
1103 2	0.			0.09	0.07	0.			
1103 3	0.			0.28	0.21	0.			
1103 4	0.			0.	0.	0.			
1103 5	7.43			6.51	4.99	4.07			
1103 6	0.			0.22	0.17	0.13			
1103 7	0.			0.43	0.32	0.26			
1103 8	0.			1.35	1.01	0.			
1103 9	0.			0.61	0.	0.			
110310	0.61			1.06	0.75	0.			
110311	0.			14.17	10.63	4.46			
110400	12.06			19.72	1127.49				
110501	0.			1319.03	81.49				
1201 1	0.			96.77	40.74				
1201 2	0.			48.39	40.74				
1201 3	0.			430.50	403.99				
1201 4	0.			0.	0.				
1201 5	804.60			762.94	1093.12				
1201 6	0.			16.46	52.23				
1201 7	0.			46.49	66.61				
1201 8	0.			436.50	403.99				
1201 9	0.			0.	0.				
120110	2413.80			130.87	127.41				
120111	0.			3322.00	3392.47				
120200	3218.40			0.	3.11				
1203 1	0.			0.37	0.25				
1203 2	0.			0.12	0.19				
1203 3	0.			0.44	0.34				
1203 4	0.			0.	0.				
1203 5	2.18			16.23	12.58				
1203 6	0.			10.84	0.69				
1203 7	0.			1.03	0.50				
1203 8	0.			2.41	1.10				
1203 9	0.			0.	0.				
120310	1.19			0.70	1.50				
120311	0.			27.24	21.04				
120400	3.37			24.20					
120501	0.			702.18					
130101	1275.47								

Table 8. (Part 7 of 7)

Account	7A.	7B.	7C.	7D.	7E.	7F.	7G.	7H.	7I.	7J.	7K.	7L.	7M.	7N.	7O.	7P.	7Q.	7R.	7S.	7T.	7U.	7V.	7W.	7X.	7Y.	7Z.
1500 2	97.	7A.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1600 3	1630.	72.	64.	61.	1.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1600 4	580.	0.	14.	59.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1600 5	234.	148.	0.	0.	2.	1.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1600 6	62.	16.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1600 7	32.	36.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1600 8	62.	36.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1600 9	3.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1600 10	9.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1600 11	451.	78.	0.	122.	11.	1.	1.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1600 12	2897.	415.	84.	125.	11.	1.	1.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1600 13	1045.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1700 1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1700 2	1.58	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1700 3	1.22	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1700 4	0.85	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1700 5	0.67	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1700 6	0.87	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1700 7	1.58	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1700 8	1.77	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1700 9	0.62	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1700 10	0.62	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1700 11	0.48	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1700 12	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1700 13	1.13	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1700 14	0.65	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1700 15	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1700 16	2.24	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1700 17	1.69	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1700 18	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
DAY 2 1	1	1	1	2	2	123.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0101 1	1	1	1	2	2	100.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0101 2	1	1	1	2	2	10.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0101 3	1	1	1	2	2	10.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0101 4	0	0	0	2	2	60.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0101 5	0	0	0	2	2	60.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0101 6	1	1	1	2	2	170.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0101 7	0	0	0	1	1	170.00	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0201 1	4.15	1.93	0.	1.27	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0201 2	4.23	1.93	0.	1.27	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0201 3	2.40	2.00	0.	1.27	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0201 4	0.	0.	0.	1.27	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0201 5	0.	0.	0.	1.27	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0201 6	0.25	0.	0.	1.27	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0201 7	452.24	432.24	0.	1.88	0.53	465.59	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0202 1	1921.54	1921.54	0.	1921.54	0.	1921.54	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0202 2	536.74	536.74	0.	536.74	0.	536.74	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0202 3	904.52	904.52	0.	904.52	0.	904.52	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0202 4	904.52	904.52	0.	904.52	0.	904.52	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0202 5	4017.50	4017.50	0.	4017.50	0.	4017.50	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0202 6	1809.08	1809.08	0.	1809.08	0.	1809.08	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0202 7	105.80	105.80	0.	105.80	0.	105.80	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0101 1	2	105.80	105.80	516	150	63	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.

item is a real or integer value. This information is supplied through the Dictionary file. A Dictionary file must be created for every uniquely formatted Summary Data file.

The Dictionary file is composed of two major parts. The first is a table which lists all legal variables that may be requested for reports from the Summary Data file and the location of the corresponding data on the file. This table will be referred to as the Dictionary items. The second part is a list of up to 20 formats which will be used to decode the data records. Table 9 is a listing of the Dictionary file created for the Summary Data file in table 8.

Table 10 describes the contents and format of the Dictionary items. The Y-axis variable name is used to designate a variable whose value is given in a record. For example, BACLS, Blue aircraft type number 1 sorties, might be used to denote the sorties flown by aircraft type F-104 in table 7. The X-axis variable name must be DAY or INDEX. The qualifier is a value to designate how the X-axis variable may be used. If the X-axis variable name is DAY, then the qualifier is set to 0 if no index value can be specified and to 1 if an index value can be specified. If the X-axis variable name is INDEX, then the qualifier must be 1, indicating that a specific day must be selected by the REMTEX user.

For example, in table 7 the total number of sorties over all missions is contained in the data record whose key is 1102 and whose index value is 0. Therefore, the Dictionary must contain the combination of BACLS, DAY, and 0 for the Y-axis variable name, X-axis variable name, and qualifier, respectively, to provide the capability for the REMTEX user to receive a plot of day versus the total number of sorties flown by aircraft type F-104 over all mission types. The Dictionary must also contain the combination of BACLS, DAY, and 1 to provide the selection of plot of day versus the number of sorties flown by aircraft type F-104 for any specified mission. The REMTEX user will specify an index value, for example, mission number, to be used for the plot but the Dictionary need only contain a 1 to indicate that an index may be specified. The Dictionary may also contain the combination of BACLS, INDEX, and 1 to allow for the plotting of mission type versus the number of sorties flown by aircraft type F-104 for any specified day. Again, even though the REMTEX user will specify a day index to be used, the Dictionary need only contain a 1 to indicate that day may be specified.

Table 9. Sample Dictionary File (Part 1 of 7)

BWS1	DAY	0	4	1	0304	1
BWS2	DAY	0	4	1	0304	2
BWS3	DAY	0	4	1	0304	3
BWS4	DAY	0	4	1	0304	4
BWS5	DAY	0	4	1	0304	5
BWS6	DAY	0	4	1	0304	6
BWS7	DAY	0	4	1	0304	7
BWS8	DAY	0	4	1	0304	8
BWS9	DAY	0	4	1	0304	9
BWS10	DAY	0	4	1	0304	10
BWS11	DAY	0	4	1	0304	11
BWS12	DAY	0	4	1	0304	12
BWS1	DAY	1	1	0	0302	1
BWS2	DAY	1	1	0	0302	2
BWS3	DAY	1	1	0	0302	3
BWS4	DAY	1	1	0	0302	4
BWS5	DAY	1	1	0	0302	5
BWS6	DAY	1	1	0	0302	6
BWS7	DAY	1	1	0	0302	7
BWS8	DAY	1	1	0	0302	8
BWS9	DAY	1	1	0	0302	9
BWS10	DAY	1	1	0	0302	10
BWS11	DAY	1	1	0	0302	11
BWS12	DAY	1	1	0	0302	12
BWS1	INDEX	1	1	0	0302	1
BWS2	INDEX	1	1	0	0302	2
BWS3	INDEX	1	1	0	0302	3
BWS4	INDEX	1	1	0	0302	4
BWS5	INDEX	1	1	0	0302	5
BWS6	INDEX	1	1	0	0302	6
BWS7	INDEX	1	1	0	0302	7
BWS8	INDEX	1	1	0	0302	8
BWS9	INDEX	1	1	0	0302	9
BWS10	INDEX	1	1	0	0302	10
BWS11	INDEX	1	1	0	0302	11
BWS12	INDEX	1	1	0	0302	12
BPS1	DAY	0	4	1	0304	13
BPS2	DAY	0	4	1	0304	14
BPS3	DAY	0	4	1	0304	15
BPS1	DAY	1	1	0	0302	13
BPS2	DAY	1	1	0	0302	14
BPS3	DAY	1	1	0	0302	15
BPS1	INDEX	1	1	0	0302	13
BPS2	INDEX	1	1	0	0302	14
BPS3	INDEX	1	1	0	0302	15
RWS1	DAY	0	4	1	0404	1
RWS2	DAY	0	4	1	0404	2
RWS3	DAY	0	4	1	0404	3
RWS4	DAY	0	4	1	0404	4
RWS5	DAY	0	4	1	0404	5
RWS6	DAY	0	4	1	0404	6

Table 9. (Part 2 of 7)

RWS7	DAY	0	4	1	0404	7
RWS8	DAY	0	4	1	0404	8
RWS9	DAY	0	4	1	0404	9
RWS10	DAY	0	4	1	0404	10
RWS11	DAY	0	4	1	0404	11
RWS12	DAY	0	4	1	0404	12
RWS1	DAY	1	1	0	0402	1
RWS2	DAY	1	1	0	0402	2
RWS3	DAY	1	1	0	0402	3
RWS4	DAY	1	1	0	0402	4
RWS5	DAY	1	1	0	0402	5
RWS6	DAY	1	1	0	0402	6
RWS7	DAY	1	1	0	0402	7
RWS8	DAY	1	1	0	0402	8
RWS9	DAY	1	1	0	0402	9
RWS10	DAY	1	1	0	0402	10
RWS11	DAY	1	1	0	0402	11
RWS12	DAY	1	1	0	0402	12
RWS1	INDEX	1	1	0	0402	1
RWS2	INDEX	1	1	0	0402	2
RWS3	INDEX	1	1	0	0402	3
RWS4	INDEX	1	1	0	0402	4
RWS5	INDEX	1	1	0	0402	5
RWS6	INDEX	1	1	0	0402	6
RWS7	INDEX	1	1	0	0402	7
RWS8	INDEX	1	1	0	0402	8
RWS9	INDEX	1	1	0	0402	9
RWS10	INDEX	1	1	0	0402	10
RWS11	INDEX	1	1	0	0402	11
RWS12	INDEX	1	1	0	0402	12
RPS1	DAY	0	4	1	0404	13
RPS2	DAY	0	4	1	0404	14
RPS3	DAY	0	4	1	0404	15
RPS1	DAY	1	1	0	0402	13
RPS2	DAY	1	1	0	0402	14
RPS3	DAY	1	1	0	0402	15
RPS1	INDEX	1	1	0	0402	13
RPS2	INDEX	1	1	0	0402	14
RPS3	INDEX	1	1	0	0402	15
BAC1S	DAY	0	2	1	1102	1
BAC2S	DAY	0	2	1	1102	2
BAC3S	DAY	0	2	1	1102	3
BAC4S	DAY	0	2	1	1102	4
BAC5S	DAY	0	2	1	1102	5
BAC6S	DAY	0	2	1	1102	6
BAC7S	DAY	0	2	1	1102	7
BAC8S	DAY	0	2	1	1102	8
BAC9S	DAY	0	2	1	1102	9
BAC10S	DAY	0	2	1	1102	10
BAC11S	DAY	0	2	1	1104	1
BAC12S	DAY	0	2	1	1104	2
BAC13S	DAY	0	2	1	1104	3
BAC14S	DAY	0	2	1	1104	4
BAC15S	DAY	0	2	1	1104	5
BAC16S	DAY	0	2	1	1104	6
BAC17S	DAY	0	2	1	1104	7
BAC18S	DAY	0	2	1	1104	8
BAC19S	DAY	0	2	1	1104	9

Table 9. (Part 3 of 7)

BAC20S	DAY	0	2	1	1104	10
BAC1S	DAY	1	2	1	1101	1
BAC2S	DAY	1	2	1	1101	2
BAC3S	DAY	1	2	1	1101	3
BAC4S	DAY	1	2	1	1101	4
BAC5S	DAY	1	2	1	1101	5
BAC6S	DAY	1	2	1	1101	6
BAC7S	DAY	1	2	1	1101	7
BAC8S	DAY	1	2	1	1101	8
BAC9S	DAY	1	2	1	1101	9
BAC10S	DAY	1	2	1	1101	10
BAC11S	DAY	1	2	1	1103	1
BAC12S	DAY	1	2	1	1103	2
BAC13S	DAY	1	2	1	1103	3
BAC14S	DAY	1	2	1	1103	4
BAC15S	DAY	1	2	1	1103	5
BAC16S	DAY	1	2	1	1103	6
BAC17S	DAY	1	2	1	1103	7
BAC18S	DAY	1	2	1	1103	8
BAC19S	DAY	1	2	1	1103	9
BAC20S	DAY	1	2	1	1103	10
BAC1S	INDEX	1	2	1	1101	1
BAC2S	INDEX	1	2	1	1101	2
BAC3S	INDEX	1	2	1	1101	3
BAC4S	INDEX	1	2	1	1101	4
BAC5S	INDEX	1	2	1	1101	5
BAC6S	INDEX	1	2	1	1101	6
BAC7S	INDEX	1	2	1	1101	7
BAC8S	INDEX	1	2	1	1101	8
BAC9S	INDEX	1	2	1	1101	9
BAC10S	INDEX	1	2	1	1101	10
BAC11S	INDEX	1	2	1	1103	1
BAC12S	INDEX	1	2	1	1103	2
BAC13S	INDEX	1	2	1	1103	3
BAC14S	INDEX	1	2	1	1103	4
BAC15S	INDEX	1	2	1	1103	5
BAC16S	INDEX	1	2	1	1103	6
BAC17S	INDEX	1	2	1	1103	7
BAC18S	INDEX	1	2	1	1103	8
BAC19S	INDEX	1	2	1	1103	9
BAC20S	INDEX	1	2	1	1103	10
BAC1K	DAY	0	2	1	1106	1
BAC2K	DAY	0	2	1	1106	2
BAC3K	DAY	0	2	1	1106	3
BAC4K	DAY	0	2	1	1106	4
BAC5K	DAY	0	2	1	1106	5
BAC6K	DAY	0	2	1	1106	6
BAC7K	DAY	0	2	1	1106	7
BAC8K	DAY	0	2	1	1106	8
BAC9K	DAY	0	2	1	1106	9
BAC10K	DAY	0	2	1	1106	10
BAC11K	DAY	0	2	1	1108	1
BAC12K	DAY	0	2	1	1108	2
BAC13K	DAY	0	2	1	1108	3
BAC14K	DAY	0	2	1	1108	4
BAC15K	DAY	0	2	1	1108	5
BAC16K	DAY	0	2	1	1108	6
BAC17K	DAY	0	2	1	1108	7

Table 9. (Part 4 of 7)

EAC18K	DAY	0	2	1	1108	8
EAC19K	DAY	0	2	1	1108	9
EAC20K	DAY	0	2	1	1108	10
BAC1K	DAY	1	2	1	1105	1
BAC2K	DAY	1	2	1	1105	2
BAC3K	DAY	1	2	1	1105	3
BAC4K	DAY	1	2	1	1105	4
BAC5K	DAY	1	2	1	1105	5
BAC6K	DAY	1	2	1	1105	6
BAC7K	DAY	1	2	1	1105	7
BAC8K	DAY	1	2	1	1105	8
BAC9K	DAY	1	2	1	1105	9
BAC10K	DAY	1	2	1	1105	10
EAC11K	DAY	1	2	1	1107	1
EAC12K	DAY	1	2	1	1107	2
EAC13K	DAY	1	2	1	1107	3
EAC14K	DAY	1	2	1	1107	4
EAC15K	DAY	1	2	1	1107	5
EAC16K	DAY	1	2	1	1107	6
EAC17K	DAY	1	2	1	1107	7
EAC18K	DAY	1	2	1	1107	8
EAC19K	DAY	1	2	1	1107	9
EAC20K	DAY	1	2	1	1107	10
BAC1K	INDEX	1	2	1	1105	1
BAC2K	INDEX	1	2	1	1105	2
BAC3K	INDEX	1	2	1	1105	3
BAC4K	INDEX	1	2	1	1105	4
BAC5K	INDEX	1	2	1	1105	5
BAC6K	INDEX	1	2	1	1105	6
BAC7K	INDEX	1	2	1	1105	7
BAC8K	INDEX	1	2	1	1105	8
BAC9K	INDEX	1	2	1	1105	9
BAC10K	INDEX	1	2	1	1105	10
EAC11K	INDEX	1	2	1	1107	1
EAC12K	INDEX	1	2	1	1107	2
EAC13K	INDEX	1	2	1	1107	3
EAC14K	INDEX	1	2	1	1107	4
EAC15K	INDEX	1	2	1	1107	5
EAC16K	INDEX	1	2	1	1107	6
EAC17K	INDEX	1	2	1	1107	7
EAC18K	INDEX	1	2	1	1107	8
EAC19K	INDEX	1	2	1	1107	9
EAC20K	INDEX	1	2	1	1107	10
RAC1S	DAY	0	2	1	1202	1
RAC2S	DAY	0	2	1	1202	2
RAC3S	DAY	0	2	1	1202	3
RAC4S	DAY	0	2	1	1202	4
RAC5S	DAY	0	2	1	1202	5
RAC6S	DAY	0	2	1	1202	6
RAC7S	DAY	0	2	1	1202	7
RAC8S	DAY	0	2	1	1202	8
RAC9S	DAY	0	2	1	1202	9
RAC10S	DAY	0	2	1	1202	10
RAC11S	DAY	0	2	1	1204	1
RAC12S	DAY	0	2	1	1204	2
RAC13S	DAY	0	2	1	1204	3
RAC14S	DAY	0	2	1	1204	4
RAC15S	DAY	0	2	1	1204	5

Table 9. (Part 5 of 7)

RAC16S	DAY	0	2	1	1204	6
RAC17S	DAY	0	2	1	1204	7
RAC18S	DAY	0	2	1	1204	8
RAC19S	DAY	0	2	1	1204	9
RAC20S	DAY	0	2	1	1204	10
RAC1S	DAY	1	2	1	1201	1
RAC2S	DAY	1	2	1	1201	2
RAC3S	DAY	1	2	1	1201	3
RAC4S	DAY	1	2	1	1201	4
RAC5S	DAY	1	2	1	1201	5
RAC6S	DAY	1	2	1	1201	6
RAC7S	DAY	1	2	1	1201	7
RAC8S	DAY	1	2	1	1201	8
RAC9S	DAY	1	2	1	1201	9
RAC10S	DAY	1	2	1	1201	10
RAC11S	DAY	1	2	1	1203	1
RAC12S	DAY	1	2	1	1203	2
RAC13S	DAY	1	2	1	1203	3
RAC14S	DAY	1	2	1	1203	4
RAC15S	DAY	1	2	1	1203	5
RAC16S	DAY	1	2	1	1203	6
RAC17S	DAY	1	2	1	1203	7
RAC18S	DAY	1	2	1	1203	8
RAC19S	DAY	1	2	1	1203	9
RAC20S	DAY	1	2	1	1203	10
RAC1S	INDEX	1	2	1	1201	1
RAC2S	INDEX	1	2	1	1201	2
RAC3S	INDEX	1	2	1	1201	3
RAC4S	INDEX	1	2	1	1201	4
RAC5S	INDEX	1	2	1	1201	5
RAC6S	INDEX	1	2	1	1201	6
RAC7S	INDEX	1	2	1	1201	7
RAC8S	INDEX	1	2	1	1201	8
RAC9S	INDEX	1	2	1	1201	9
RAC10S	INDEX	1	2	1	1201	10
RAC11S	INDEX	1	2	1	1203	1
RAC12S	INDEX	1	2	1	1203	2
RAC13S	INDEX	1	2	1	1203	3
RAC14S	INDEX	1	2	1	1203	4
RAC15S	INDEX	1	2	1	1203	5
RAC16S	INDEX	1	2	1	1203	6
RAC17S	INDEX	1	2	1	1203	7
RAC18S	INDEX	1	2	1	1203	8
RAC19S	INDEX	1	2	1	1203	9
RAC20S	INDEX	1	2	1	1203	10
RAC1K	DAY	0	2	1	1206	1
RAC2K	DAY	0	2	1	1206	2
RAC3K	DAY	0	2	1	1206	3
RAC4K	DAY	0	2	1	1206	4
RAC5K	DAY	0	2	1	1206	5
RAC6K	DAY	0	2	1	1206	6
RAC7K	DAY	0	2	1	1206	7
RAC8K	DAY	0	2	1	1206	8
RAC9K	DAY	0	2	1	1206	9
RAC10K	DAY	0	2	1	1206	10
RAC11K	DAY	0	2	1	1208	1
RAC12K	DAY	0	2	1	1208	2
RAC13K	DAY	0	2	1	1208	3

Table 9. (Part 6 of 7)

RAC14K	DAY	0	2	1	1208	4
RAC15K	DAY	0	2	1	1208	5
RAC16K	DAY	0	2	1	1208	6
RAC17K	DAY	0	2	1	1208	7
RAC18K	DAY	0	2	1	1208	8
RAC19K	DAY	0	2	1	1208	9
RAC20K	DAY	0	2	1	1208	10
RAC1K	DAY	1	2	1	1205	1
RAC2K	DAY	1	2	1	1205	2
RAC3K	DAY	1	2	1	1205	3
RAC4K	DAY	1	2	1	1205	4
RAC5K	DAY	1	2	1	1205	5
RAC6K	DAY	1	2	1	1205	6
RAC7K	DAY	1	2	1	1205	7
RAC8K	DAY	1	2	1	1205	8
RAC9K	DAY	1	2	1	1205	9
RAC10K	DAY	1	2	1	1205	10
RAC11K	DAY	1	2	1	1207	1
RAC12K	DAY	1	2	1	1207	2
RAC13K	DAY	1	2	1	1207	3
RAC14K	DAY	1	2	1	1207	4
RAC15K	DAY	1	2	1	1207	5
RAC16K	DAY	1	2	1	1207	6
RAC17K	DAY	1	2	1	1207	7
RAC18K	DAY	1	2	1	1207	8
RAC19K	DAY	1	2	1	1207	9
RAC20K	DAY	1	2	1	1207	10
RAC1K	INDEX	1	2	1	1205	1
RAC2K	INDEX	1	2	1	1205	2
RAC3K	INDEX	1	2	1	1205	3
RAC4K	INDEX	1	2	1	1205	4
RAC5K	INDEX	1	2	1	1205	5
RAC6K	INDEX	1	2	1	1205	6
RAC7K	INDEX	1	2	1	1205	7
RAC8K	INDEX	1	2	1	1205	8
RAC9K	INDEX	1	2	1	1205	9
RAC10K	INDEX	1	2	1	1205	10
RAC11K	INDEX	1	2	1	1207	1
RAC12K	INDEX	1	2	1	1207	2
RAC13K	INDEX	1	2	1	1207	3
RAC14K	INDEX	1	2	1	1207	4
RAC15K	INDEX	1	2	1	1207	5
RAC16K	INDEX	1	2	1	1207	6
RAC17K	INDEX	1	2	1	1207	7
RAC18K	INDEX	1	2	1	1207	8
RAC19K	INDEX	1	2	1	1207	9
RAC20K	INDEX	1	2	1	1207	10
BPC	DAY	0	3	1	1605	1
RPC	DAY	0	3	1	1610	1
BWL1	DAY	0	3	1	1605	2
BWL2	DAY	0	3	1	1605	3
BWL3	DAY	0	3	1	1605	4
BWL4	DAY	0	3	1	1605	5
BWL5	DAY	0	3	1	1605	6
BWL6	DAY	0	3	1	1605	7
BWL7	DAY	0	3	1	1605	8
BWL8	DAY	0	3	1	1605	9
BWL9	DAY	0	3	1	1605	10

Table 9. (Part 7 of 7)

BWL10	DAY	0	3	1	1605	11
BWL11	DAY	0	3	1	1605	12
BWL12	DAY	0	3	1	1605	13
RWL1	DAY	0	3	1	1610	2
RWL2	DAY	0	3	1	1610	3
RWL3	DAY	0	3	1	1610	4
RWL4	DAY	0	3	1	1610	5
RWL5	DAY	0	3	1	1610	6
RWL6	DAY	0	3	1	1610	7
RWL7	DAY	0	3	1	1610	8
RWL8	DAY	0	3	1	1610	9
RWL9	DAY	0	3	1	1610	10
RWL10	DAY	0	3	1	1610	11
RWL11	DAY	0	3	1	1610	12
RWL12	DAY	0	3	1	1610	13

4
(14x,1217,518)
(10F10.2)
(13F8.0)
(12x,F9.0,11F7.0,3F9.0)

Table 10. Format for Dictionary Items

<u>Columns</u>	<u>Entry</u>	<u>Range</u>	<u>Format</u>
1-6	Y-axis variable name		A6 left justified
7-8	Blank		
9-14	X-axis variable name	DAY or INDEX	A6 left justified
15-16	Blank		
17	Qualifier	0 or 1	I1
18-19	Blank		
20-21	Format index	1-20	I2 right justified
22-23	Blank		
24	Type of data	Real = 1 Integer = 0	I1
25-26	Blank		
27-30	Key		A4 left justified
31-32	Blank		
33-34	Index of item in the data record		I2 right justified

When the REMTEX user requests a report, he specifies the Y-axis variable name, X-axis variable name, and an index value. The index value will determine whether the qualifier is a 0 or 1. The Dictionary is then searched for the combination of Y-variable, X-variable, and qualifier to determine whether a request is valid. For any valid combination, the Dictionary will provide the information needed to locate the data on the Summary Data file. The format index will indicate which of the formats listed in the second part of the Dictionary will be used to decode the data record, and the type of data will indicate whether the Y-axis variable is real (type = 1) or integer (type = 0). The key will indicate the key of the data record on the Summary Data file and together with the index value will determine the exact record in the Summary Data file that will contain the pertinent data. Finally, the index of the item in the data record will specify the exact data field to be extracted from the data record.

The second part of the Dictionary file contains a maximum of 20 formats which will be used to decode the various data records on the Summary Data file. This set of formats is ordered and the indexes of the format statements are used in the list of Dictionary items as described above.

The Dictionary file is structured as shown in table 11. The list of Dictionary items is followed by a flag ZZZZZZ to designate the end of the first part of the file. The next item on the file is an integer value (in I2 format) to specify the number of formats which follow. Each format is then specified in A36 format, beginning with "(" and ending with ")". For example, if the data is in 10(F10.2,2X) format, the format specification would be (10(F10.2,2X)).

2.3.3.2.2 Commands File. The Commands file contains the groups of commands that constitute a report request. The number of report requests on one Commands file is theoretically unlimited. This file is an input to RXPROC and normally is loaded by the RXEDIT program based on interactive user responses. The command is located in columns 1 through 3 and the text is located in columns 7 through 42. The legal commands and their meaning are presented in table 12. A sample Commands file, as produced by RXEDIT, is displayed in table 13.

Table 11. Structure of the Dictionary File

<u>Data</u>	<u>Format</u>
1. Dictionary items.	As shown in table 10.
2. Flag to designate end of Dictionary items.	"ZZZZZZ" in columns 1-6.
3. Number of formats used.	I2 in columns 1-2.
4. The formats.	A36 with "(" as the first character and ")" as the last nonblank character entered in columns 1-36.

Table 12. Command Identification Keys for the Commands File

<u>Key Name</u>	<u>Command Description</u>
RTT	Report title
XAX	X-axis title
YAX	Y-axis title
XSU	X-axis scale, upper
YSL	Y-axis scale, lower
YSU	Y-axis scale, upper
XVB	X variable name
YVB	Y variable name and its quali- fier
FMT	Output format
HAX	Histogram Y-axis title
HDD	Histogram data display
ODS	Output destination
ERP	End of report number n

Table 13. A Sample Commands File Produced by RXEDIT
 (Part 1 of 2)

```

RTT   RED WEAPONS LOSSES
RTT   -----
RTT   REPORT DATE - 19 JAN 77
XAX                                     DAY
YAX   WEAPONS TYPE 1
YAX   WEAPONS TYPE 2
YAX   WEAPONS TYPE 3
XSU           20
YSL           0
YSU       25000
XVB   DAY
YVB   RWL1           0
YVB   RWL2           0
YVB   RWL3           0
FMT   BOTH
HAX
HDD   YES
ODS   BOTH
ERP           1
  
```

Table 13. (Part 2 of 2)

RTI	RED WEAPONS LOSSES	
RTI	-----	
RTI	REPORT DATE - 19 JAN 77	
XAX		DAY
YAX	WEAPONS TYPE 1	
YAX	WEAPONS TYPE 2	
YAX	WEAPONS TYPE 3	
XSU	20	
YSL	0	
YSU	25000	
XVB	DAY	
YVB	RWL1	0
YVB	RWL2	0
YVB	RWL3	0
FMT	TABLE	
ERP	2	

2.3.3.2.3 File Pointers File. The File Pointers file contains the information needed to locate the Dictionary file, the Summary Data file, and the Commands file. This file is used by RXPROC to dynamically attach the input files and normally is loaded by the RXEDIT program based on interactive user responses. The legal records are presented below:

<u>Columns 1-3</u>	<u>Columns 7-42</u>
DIC	catalog-file string for the Dictionary file
COM	catalog-file string for the Commands file
CFS	catalog-file string for the Summary Data file
TPN	5-digit left-justified reel number for the Summary Data tape.

Table 14 shows a sample File Pointers file created by RXEDIT.

2.4 REMTEX Flexibility

Although the REMTEX Postprocessor system was developed to support the new IDAGAM II model, it was designed and implemented as a generalized system. Any Honeywell Series 600/6000 user can use REMTEX by keying his Summary Data file and loading a Dictionary file that describes the contents of the Summary Data file. Although REMTEX searches for a key of DAY, any ascending chronological value may be contained in the record, such as week number. Since the user specifies all axis and report titles, a change of the chronological value would be transparent.

Also, if the user does not have access to a remote terminal, he may bypass the RXEDIT interactive portion of REMTEX by loading his File Pointers file and his Commands file through the regular HIS batch procedure as described in subsection 2.3.3.2.

Finally, if the user does not have access to a CalComp 7000 plotter, he may produce all of the allowable output formats solely on his online printer.

Table 14. Sample File Pointers File Created by RXEDI¹⁷

DIC	674IDP00/CSC/DICT
COM	674IDP00/CSC/RXCOMM
TPN	99999

SECTION 3. FINDINGS AND CONCLUSIONS

Based on the REMTEX description presented in this TM, the logical conclusion is that the REMTEX system is a valuable tool for users who meet the following criteria:

- a. The user has access to an HIS 600 or 6000 Series computer.
- b. The user has a requirement for the display of his output data in graph, histogram, and/or tabular format.
- c. The user has one or more programs that produce or are capable of producing a keyed BCD output file in HIS system standard format.

APPENDIX A
SOURCE LISTING OF RXEDIT

```

*#RUN=574IDP00/CSC/RXFEDITHS(BCD,CORE=24)
PARAMETER NDP=1000
CHARACTER RTIT*36(3),XTIT*36,YTIT*36(6),XSU*6,YSL*6,YSU*6,
RXVAR*6,YVAR*6(6),QUAL*6(6),INPSC*6,TAPNO*6,CATL*36,OUTDS*36,
RDUJFM*36,HISTIL*36,REPLY*6,ERP*6,DICT*36,COMM*36,DISP*6
CHARACTER DNAM*6(NDP,2)
INTEGER DQUAL(NDP)
CHARACTER BLANKS/4H /
DIMENSION BUF(380)
CHARACTER CATFIL*37,SEM*1
DATA SEM,ISTAT1,ISTAT2/" ";" ,0400000000000,0403700000000/
DATA ISTAT3/0000000000000/
IERP=0
WRITE(06,100)
100 FORMAT("NOTIFY COMPUTER OPERATIONS TO OBTAIN ALL NECESSARY",
&" MAGNETIC TAPES FROM",/,,"THE TAPE LIBRARY.",/,," REMTEX IS EXECUTING -",
&" PLEASE RESPOND WHEN = APPEARS.")

C - ENTER AND WRITE TO FILE 30 THE NEEDED FILES AND/OR TAPES. -

633 WRITE(06,4000)
4000 FORMAT("ENTER CATALOG-FILE STRING FOR THE DICTIONARY, 36 ",
&"CHARACTERS MAXIMUM.")
READ(05,201) DICT
ENCODE(CATFIL,5001) DICT,SEM
5001 FORMAT(A36,A1)
CALL ATTACH(10,CATFIL,1,0,ISTAT,BUF)
IF(ISTAT.EQ.ISTAT1.OR.ISTAT.EQ.ISTAT2) GO TO 666
WRITE(06,5002)
5002 FORMAT("ILLEGAL CATALOG-FILE STRING")
GO TO 633
666 WRITE(06,4001)
4001 FORMAT("ENTER CATALOG-FILE STRING FOR THE COMMANDS FILE",
&" 36 CHARACTERS MAXIMUM.")
READ(05,201) COMM
ENCODE(CATFIL,5001) COMM,SEM
CALL ATTACH(15,CATFIL,2,0,ISTAT,BUF)
IF(ISTAT.EQ.ISTAT1.OR.ISTAT.EQ.ISTAT2) GO TO 515
WRITE(06,5002)
GO TO 666
515 WRITE(06,112)
112 FORMAT(45HENTER INPUT SOURCE "T","TAPE","D",OR "DISK")
READ(05,202) INPSC
IF(INPSC.EQ."D".OR.INPSC.EQ."DISK") GO TO 520
IF(INPSC.EQ."T".OR.INPSC.EQ."TAPE") GO TO 525
WRITE(06,875)
&75 FORMAT("ILLEGAL ENTRY")
GO TO 515
525 WRITE(06,113)
113 FORMAT("ENTER TAPE NUMBER OF INPUT SOURCE - 5 INTEGER DIGITS")
READ(05,202) TAPNO
GO TO 530
520 CONTINUE
WRITE(06,114)
114 FORMAT("ENTER CATALOG-FILE STRING OF INPUT SOURCE - CAT-1",
&"CAT-2",.../FILE, -- 36",/,," CHARACTERS MAXIMUM.")

```

```

READ(05,201) CATFL
ENCODE(CATFIL,5001) CATFL,SEM
CALL ATTACH(25,CATFIL,3,0,ISTAT,BUF)
IF(ISTAT.EQ.ISTAT1.OR.ISTAT.EQ.ISTAT2) GO TO 530
WRITE(06,5002)
GO TO 520
530 CONTINUE
WRITE(06,4002) DICT
4002 FORMAT("CATALOG-FILE STRING FOR THE DICTIONARY = ",A36)
WRITE(06,4003) COMM
4003 FORMAT("CATALOG-FILE STRING FOR THE COMMANDS FILE = ",A36)
WRITE(06,780) INPSC
780 FORMAT("INPUT SOURCE = ",A6)
IF(INPSC.EQ."D".OR.INPSC.EQ."DISK") GO TO 790
WRITE(06,785) TAPNO
785 FORMAT("TAPE NUMBER OF INPUT SOURCE = ",A6)
GO TO 786
790 WRITE(06,791) CATFL
791 FORMAT("CATALOG-FILE STRING OF INPUT SOURCE = ",A36)
786 CONTINUE
WRITE(06,5024)
5024 FORMAT("DO YOU WISH TO CHANGE ANY CATALOG FILE NAMES -",
8" 'YES' OR 'NO'")
READ(05,202) REPLY
IF(REPLY.EQ."YES") GO TO 633
IF(REPLY.EQ."NO") GO TO 973
WRITE(06,875)
GO TO 786
973 WRITE(30,5020) DICT
5020 FORMAT(6HDIC ,A36)
WRITE(30,5021) COMM
5021 FORMAT(6HCOM ,A36)
IF(INPSC.EQ."D".OR.INPSC.EQ."DISK") GO TO 970
WRITE(30,968) TAPNO
968 FORMAT(6HTPN ,A6)
GO TO 974
970 WRITE(30,972) CATFL
972 FORMAT(6HCFS ,A36)
974 CONTINUE
WRITE(06,5025)
5025 FORMAT("DO YOU WISH TO LOAD THE COMMANDS FILE - 'YES' OR 'NO'")
READ(05,202) REPLY
IF(REPLY.EQ."NO") GO TO 650
IF(REPLY.EQ."YES") GO TO 901
WRITE(06,875)
GO TO 974

```

C - READ IN DICTIONARY. -

```

901 NDI=0
900 NDI=NDI+1
IF(NDI.LE.NDP) GO TO 910
WRITE(06,920) NDP
920 FORMAT("COMPLETE DICTIONARY IS NOT IN STORAGE.",/, " THERE IS ONLY ",
8"ROOM FOR ",I5," ITEMS. PROGRAM TERMINATED.")
STOP

```

```

910 READ(10,940,END=930) (DNAM(NDI,I),I=1,2),DQUAL(NDI)
940 FORMAT(2(A6,2X),I1)
IF(DNAM(NDI,1).NE."ZZZZZ")GO TO 900
930 NDI=NDI-1

```

C - ENTER COMMANDS FOR PLOTS. -

```

1) ISTOP=0
DO 12 I=1,3
WRITE(06,101) I
101 FORMAT("ENTER LINE ",I1," OF REPORT TITLE - 36 CHARACTERS MAXIMUM.")
READ(05,201) RTIT(I)
201 FORMAT(A36)
12 CONTINUE
WRITE(06,104)
104 FORMAT(43HENTER X-AXIS TITLE - 36 CHARACTERS MAXIMUM.)
READ(05,201) XTIT
NOYVAR =0
430 DO 500 I=1,6
WRITE(06,105) I
105 FORMAT("ENTER Y-AXIS TITLE ",I1," 36 CHARACTERS MAXIMUM. IF YOU",//
&"HAVE ALREADY TRANSMITTED YOUR LAST Y-AXIS TITLE, ENTER ONE BLANK",//
&"AND EXECUTE A CARRIAGE RETURN.")
READ(05,201) YTIT(I)
IF(YTIT(I).EQ.BLANKS) GO TO 502
NOYVAR=I
500 CONTINUE
502 IF(NOYVAR.GT.0) GO TO 501
WRITE(06,405)
405 FORMAT("THE FIRST Y-AXIS TITLE CANNOT BE BLANK.")
GO TO 400
501 WRITE(06,107)
107 FORMAT("ENTER X-AXIS SCALE(UPPER END)--'D','DEFAULT', OR A ",
&"POSITIVE INTEGER",//," WHICH IS LESS THAN OR EQUAL TO 100.")
READ(05,202) XSU
IF(XSU.EQ."D".OR.XSU.EQ."DEFAULT") GO TO 450
CALL ADJUST(XSU)
202 FORMAT(A6)
DECODE(XSU,435) IXSU
435 FORMAT(I6)
IF(IXSU.GT.0.AND.IXSU.LE.100) GO TO 440
WRITE(06,875)
GO TO 501
440 WRITE(06,108)
108 FORMAT("ENTER Y-AXIS SCALE(LOWER END), AN INTEGER",
&" WHICH IS GREATER THAN OR",//," EQUAL TO ZERO AND LESS THAN ONE MILLION.")
READ(05,202) YSL
IF(YSL.EQ."D".OR.YSL.EQ."DEFAULT") GO TO 446
CALL ADJUST(YSL)
DECODE(YSL,435) IYSL
IF(IYSL.GE.0.AND.IYSL.LT.1000000) GO TO 445
446 WRITE(06,875)
GO TO 440
445 WRITE(06,109)
109 FORMAT("ENTER Y-AXIS SCALE(UPPER END), AN INTEGER",
&" GREATER THAN ZERO AND LESS THAN ONE MILLION.")

```

```

READ(05,202) YSU
IF(YSU.EQ."D".OR.YSU.EQ."DEFAULT") GO TO 447
CALL ADJUST(YSU)
DECODE(YSU,435) IYSU
IF(IYSU.GT.0.AND.IYSU.LT.1000000) GO TO 450
447 WRITE(06,875)
GO TO 445
450 ICOUNT=0
1060 ICOUNT=ICOUNT+1
1002 WRITE(06,110)
110 FORMAT("ENTER X-AXIS VARIABLE NAME, 'DAY' OR 'INDEX'.")
READ(05,202) XVAR
IF(XVAR.EQ."DAY".OR.XVAR.EQ."INDEX") GO TO 1001
WRITE(06,800)
800 FORMAT("ILLEGAL X-AXIS VARIABLE USED.")
GO TO 1002
1001 CONTINUE
DO 510 I=1,N0YVAR
1012 WRITE(06,111) I
111 FORMAT(28HENTER Y-AXIS VARIABLE NAME ,I1)
READ(05,202) YVAR(I)
420 WRITE(06,125) I
125 FORMAT("ENTER QUALIFIER VALUE ",I1,". IF X-AXIS VARIABLE IS 'DAY',",
&,"ENTER '0' FOR TOTALS OR A POSITIVE INTEGER TO SET AN INDEX VALUE;",
&" IF X-AXIS VARIABLE IS 'INDEX', ENTER A POSITIVE INTEGER TO SPECIFY",
&," THE DAY NUMBER. THE VALUE MUST BE LESS THAN OR EQUAL TO 100.")
READ(05,202) QUAL(I)
CALL ADJUST(QUAL(I))
DECODE(QUAL(I),203) IQUAL
203 FORMAT(I6)
IF(IQUAL.LE.100.AND.IQUAL.GE.0) GO TO 410
WRITE(06,875)
GO TO 420
410 IF(IQUAL.NE.0) IQUAL=1
IFLAG=0
DO 1010 J=1,NDI
IF(YVAR(I).NE.DNAM(J,1)) GO TO 1010
IFLAG=1
IF(XVAR.EQ.DNAM(J,2).AND.IQUAL.EQ.DQUAL(J)) GO TO 510
1010 CONTINUE
IF(IFLAG.NE.0) GO TO 1020
WRITE(06,801)
801 FORMAT("ILLEGAL Y-AXIS VARIABLE USED.")
GO TO 1012
1020 IF(ICOUNT.GE.3) GO TO 855
IC=3-ICOUNT
WRITE(06,810) IC
810 FORMAT("ILLEGAL COMBINATION OF Y-AXIS VARIABLE, X-AXIS ",
&"VARIABLE, AND QUALIFIER VALUE.\"", " PROGRAM WILL ALLOW YOU ",I2,
&" MORE TRIES AT A LEGAL COMBINATION.")
GO TO 1060
855 WRITE(06,860)
860 FORMAT("YOU HAVE FAILED TO REQUEST A VALID REPORT.",
&"THE ENTRIES",
&" MADE FOR THIS REPORT WILL BE IGNORED.",
&"THE ENTRIES YOU HAVE MADE ",
&"ARE AS FOLLOWS -")
ISTOP=1

```

```

GO TO 630
510 CONTINUE
537 WRITE(06,116)
116 FORMAT(68HENTER OUTPUT FORMAT - "H","HISTOGRAM","G","GRAPH","B",
8"90TH","TABLE")
READ(05,201) OUTFM
IF(OUTFM.NE."TABLE") GO TO 541
WRITE(06,6002)
6002 FORMAT("YOUR SELECTION WILL CAUSE A TABLE TO BE PRODUCED BY THE",
8" PRINTER. NO",//," PLOTS WILL BE PRODUCED.")
GO TO 605
541 IF(OUTFM.EQ."H".OR.OUTFM.EQ."HISTOGRAM".OR.OUTFM.EQ."B".OR.OUTFM.EQ.
8"90TH") GO TO 540
IF(OUTFM.EQ."G".OR.OUTFM.EQ."GRAPH") GO TO 535
WRITE(06,875)
GO TO 537
540 WRITE(06,117)
117 FORMAT("ENTER HISTOGRAM Y-AXIS TITLE - 36 CHARACTERS MAXIMUM.",//
8"BLANK DEFAULTS TO Y-AXIS TITLE 1")
READ(05,201) HISTIL
533 WRITE(06,119)
119 FORMAT("DO YOU WISH TO DISPLAY DATA VALUES ABOVE",//
8" EACH HISTOGRAM BAR --'YES' OR 'NO'")
READ(05,202) DISP
IF(DISP.EQ.'YES'.OR.DISP.EQ.'NO') GO TO 535
WRITE(06,875)
GO TO 533
535 WRITE(06,115)
115 FORMAT(65HENTER OUTPUT DESTINATION - "C","CALCOMP","P","PRINTER","B",
8"90TH")
READ(05,201) OUTDS
IF(OUTDS.EQ."C".OR.OUTDS.EQ."CALCOMP".OR.OUTDS.EQ."P".OR.OUTDS.EQ.
8"PRINTER".OR.OUTDS.EQ."B".OR.OUTDS.EQ."90TH") GO TO 605
WRITE(06,875)
GO TO 535

```

C - DISPLAY REPORT OF COMMANDS. -

```

605 WRITE(06,600)
600 FORMAT("DISPLAY OF REPORT DESIRED - 'YES' OR 'NO' ")
READ(05,202) REPLY
IF(REPLY.EQ."NO") GO TO 610
IF(REPLY.EQ."YES") GO TO 630
WRITE(06,875)
GO TO 605
630 CONTINUE
DD 710 I=1,3
WRITE(06,700) I,RTIT(I)
700 FORMAT("LINE ",I1," OF REPORT TITLE = ",A36)
710 CONTINUE
WRITE(06,720) XTIT
720 FORMAT("X-AXIS TITLE = ",A36)
DD 730 I=1,N0YVAR
WRITE(06,740) I,YTIT(I)
740 FORMAT("Y-AXIS TITLE ",I1," = ",A36)
730 CONTINUE

```

```

WRITE(06,750) XSU
750 FORMAT("X-AXIS SCALE ( UPPER END ) = ",A6)
IF(XSU.EQ."D".OR.XSU.EQ."DEFAULT") GO TO 756
WRITE(06,755) YSL
755 FORMAT("Y-AXIS SCALE (LOWER END) = ",A6)
WRITE(06,760) YSU
760 FORMAT("Y-AXIS SCALE (UPPER END) = ",A6)
756 WRITE(06,765) XVAR
765 FORMAT("X-AXIS VARIABLE NAME = ",A6)
DO 770 I=1,NOYVAR
WRITE(06,775) I,YVAR(I),QUAL(I)
775 FORMAT("Y-AXIS VARIABLE NAME ",I1," AND ITS QUALIFIER = ",A6,2X,A6)
770 CONTINUE
IF(ISTOP.EQ.1) GO TO 870
WRITE(06,793) OUTFM
793 FORMAT("OUTPUT FORMAT = ",A36)
IF(OUTFM.EQ."TABLE") GO TO 610
IF(OUTFM.EQ."G".OR.OUTFM.EQ."GRAPH") GO TO 615
WRITE(06,794) HISTIL
794 FORMAT("HISTOGRAM Y-AXIS TITLE = ",A36)
WRITE(06,795) DISP
795 FORMAT("DISPLAY OF HISTOGRAM DATA = ",A6)
615 WRITE(06,792) OUTDS
792 FORMAT("OUTPUT DESTINATION = ",A36)
610 WRITE(06,620)
620 FORMAT("DO YOU WISH TO KEEP THIS REPORT? --'YES' OR 'NO' ")
READ(05,202) REPLY
IF (REPLY.EQ."NO") GO TO 870
IF(REPLY.EQ."YES" ) GO TO 640
WRITE(06,875)
GO TO 610

```

C - WRITE ON COMMANDS FILE. -

```

640 IERP=IERP+1
DO 944 I=1,3
WRITE(15,942) RTIT(I)
942 FORMAT(6HRTI ,A36)
944 CONTINUE
WRITE(15,946) XTIT
946 FORMAT(6HXXA ,A36)
DO 948 I=1,NOYVAR
WRITE(15,950) YTIT(I)
950 FORMAT(6HYAX ,A36)
948 CONTINUE
WRITE(15,952) XSU
952 FORMAT(6HXSU ,A6)
IF(XSU.EQ."D".OR.XSU.EQ."DEFAULT") GO TO 967
WRITE(15,954) YSL
954 FORMAT(6HYSL ,A6)
WRITE(15,956) YSU
956 FORMAT(6HYSU ,A6)
957 WRITE(15,958) XVAR
958 FORMAT(6HXVB ,A6)
DO 966 I=1,NOYVAR
WRITE(15,960) YVAR(I),QUAL(I)

```

```

950 FORMAT(6HYVR ,A6,A6)
956 CONTINUE
WRITE(15,978) OUTFM
978 FORMAT(6HFMT ,A36)
IF(OUTFM.EQ."TABLE") GO TO 3000
IF(OUTFM.EQ."G".OR.OUTFM.EQ."GRAPH") GO TO 969
WRITE(15,980)HISTIL
980 FORMAT(6HHAX ,A36)
WRITE(15,981) DISP
981 FORMAT(6HHDD ,A6)
959 WRITE(15,976) OUTDS
976 FORMAT(6HODS ,A36)
3000 ENCODE(ERP,982) IERP
982 FORMAT(16)
CALL ADJUST(ERP)
WRITE(15,984) ERP
984 FORMAT(6HERP ,A6)
870 WRITE(06,118)
118 FORMAT(43HADDITIONAL REPORTS DESIRED? - "YES" OR "NO")
READ(05,202) REPLY
IF(REPLY.EQ."YES") GO TO 10
IF(REPLY.EQ."NO") GO TO 645
WRITE(06,875)
GO TO 870

```

C - DETACH FILES. -

```

545 ENDFILE 15
650 ISWITCH=ISTAT3
CALL DETACH(10,ISTAT,BUF)
IF(ISTAT.NE.ISTAT3) ISWITCH=ISTAT
CALL DETACH(15,ISTAT,BUF)
IF(ISTAT.NE.ISTAT3) ISWITCH=ISTAT
CALL DETACH(30,ISTAT,BUF)
IF(ISTAT.NE.ISTAT3) ISWITCH=ISTAT
CALL DETACH(25,ISTAT,BUF)
IF(ISTAT.NE.ISTAT3) ISWITCH=ISTAT
IF(ISWITCH.EQ.ISTAT3) GO TO 651
WRITE(06,5005)
5005 FORMAT("UNABLE TO DETACH FILES. EXECUTE A 'REMO CLEARFILES'",
&/," AFTER THIS PROGRAM TERMINATES.")
551 CONTINUE
WRITE(06,199)
199 FORMAT(33HREMTX EDIT PROGRAM HAS COMPLETED)
WRITE(06,198)
198 FORMAT(///15HHAVE A NICE DAY)
STOP
END

```

```
SUBROUTINE ADJUST(BUFF)
CHARACTER BUFF*6,REC*1(6),BLANK*1
DATA BLANK/1H /
DECODE(BUFF,300) (REC(I),I=1,6)
300 FORMAT(6A1)
DO 400 I=1,6
IF(REC(6-I+1).EQ.BLANK) GO TO 400
JJ=I-1
IF(JJ.EQ.0) RETURN
GO TO 410
400 CONTINUE
RETURN
410 DO 420 I=1,JJ
DO 450 J=1,5
REC(6-J+1)=REC(6-J)
450 CONTINUE
REC(1)=BLANK
420 CONTINUE
ENCODE(BUFF,310) (REC(I),I=1,6)
310 FORMAT(6A1)
RETURN
END
```

APPENDIX B
SOURCE LISTING OF RXPROC

```

1      COMMON/PLOTTER/ XDAT(11,100),YDAT(11,100),NBR(11),
2      *      YLABLS(6,10),XLABL(6),LINES,TYPE,TITLE(6),TITLE1(6),
3      *      TITLE2(6),TITLE3(6),XSCALE(11),YSCALE(11)
4      PARAMETER NDP=1000
5      DIMENSION IGRAPH(22,56)
6      DIMENSION HISLAB(6)
7      DIMENSION ICHAR(36,7)
8      INTEGER CATFIL(7),ONE,THREE,FIVE,ONETWO,TREFOR,FIVSIX,UNDER1,
9      1 UNDER2
10     DIMENSION IQUAL(6),IFORM(6),IENT(6),ITYP(6)
11     DIMENSION IX(20),X(20)
12     DIMENSION ITAPE(2)
13     INTEGER BUFFER(400)
14     CHARACTER CGRAPH*6(22,56),CNBUFF*6(7),CYLABL*6(6,10),CHISLB*6(6),
15     1 LINSIZ*6(14)
16     EQUIVALENCE (IGRAPH(1,1),CGRAPH(1,1)),(INBUFF(1),CNBUFF(1)),
17     2 (HISLAB(1),CHISLB(1)),(YLABLS(1,1),CYLABL(1,1))
18     EQUIVALENCE (X,IX)
19     CHARACTER DATV*126,ITABT*6,ITAB*6(6),FORMT*36
20     CHARACTER FORM*36(20)
21     CHARACTER DNAM(NDP,2),DTAB(NDP)
22     INTEGER DQUAL(NDP),DFORM(NDP),DTYP(NDP),DENT(NDP)
23     DIMENSION INBUFF(7)
24     INTEGER TYPE
25     INTEGER UNIT(6)
26     INTEGER TICK2,TICK4,TICK6
27     INTEGER YLABLS,XLABL,TITLE,TITLE1,TITLE2,TITLE3,HISLAB
28     INTEGER XSCALE,YSCALE,XINC,YINC
29     INTEGER APOSTR,BLANK,YDASH
30     INTEGER APOS1,APOS2,APOS3,APOS4,APOS5,APOS6,APOS16,DAP
31     INTEGER QUAL(6),PTDATA(40,100)
32     CHARACTER ICCMM*6,XVAR*6,YVAR*6(6)
33     DATA DAP/6H      -'/
34     DATA APOS1  /6H'   /
35     DATA APOS2  /6H'   /
36     DATA APOS3  /6H'   /
37     DATA APOS4  /6H'   /
38     DATA APOS5  /6H'   /
39     DATA APOS6  /6H'   /
40     DATA APOS16 /6H'   /
41     DATA YDASH/6H-----/
42     DATA TICK2/6H--+---/
43     DATA TICK4/6H----+--/
44     DATA TICK6/6H-----+/
45     DATA UNIT/1000000,100000,10000,1000,100,10/
46     DATA BLANK/6H      /
47     DATA ONETWO   /' 1 2'/
48     DATA TREFOR  /' 3 4'/
49     DATA FIVSIX  /' 5 6'/
50     DATA ONE     /' 1  '/
51     DATA THREE   /' 3  '/
52     DATA FIVE    /' 5  '/
53     DATA UNDER1 /' -  '/
54     DATA UNDER2 /' - -'/
55     DATA CATFIL(7) /';';'/
56     DATA ITAPE(1) /6H50002U/

```

```

57 C ** SECTION A PERFORM INITIAL HOUSEKEEPING.
58 C READ FILE POINTERS FILE (FC=30).
59 C ATTACH DICTIONARY FILE (FC=10),
60 C RXEDIT COMMANDS FILE (FC=15),
61 C AND INPUT DATA FILE (FC=20, TAPE OR DISC).
62 C READ AND STORE DICTIONARY DATA.
63 CALL PLUTS(0,0,25)
64 NUM=1
65 ITAPNO=0
66 MOT=10
67 50 READ(30,100,END=60,ERR=800) LINSIZ
68 WRITE(06,100) LINSIZ
69 ICOMM = LINSIZ(1)
70 DO 51 I=1,6
71 51 INBUFF(I) = LINSIZ(I+1)
72 IF(ICOMM.EQ.'DIC') GO TO 52
73 IF(ICOMM.EQ.'COM') GO TO 56
74 IF(ICOMM.EQ.'TPN') GO TO 550
75 IF(ICOMM.EQ.'CFS') GO TO 555
76 CALL ERRMSG(12,$901,$903)
77 GO TO 50
78 DO 53 I=1,6
79 53 CATFIL(I) = INBUFF(I)
80 CALL ATTACH(MOT,CATFIL,I,0,ISTAT,BUFFER)
81 IF(ISTAT.EQ.0) GO TO 50
82 WRITE(06,606) CATFIL,ISTAT
83 606 FORMAT(1X,7A6,' ISTAT=',A6)
84 CALL ERRMSG(7,$901,$903)
85 GO TO 50
86 DO 57 I=1,6
87 57 CATFIL(I) = INBUFF(I)
88 CALL ATTACH(15,CATFIL,I,0,ISTAT,BUFFER)
89 IF(ISTAT.EQ.0) GO TO 50
90 WRITE(06,606) CATFIL,ISTAT
91 CALL ERRMSG(7,$901,$903)
92 GO TO 50
93 550 CONTINUE
94 ITAPE(2) = INBUFF(1)
95 CALL MORE(ITAPE,1,ISTAT)
96 IF(ISTAT.EQ.0) GO TO 551
97 WRITE(06,552) ISTAT
98 552 FORMAT(' ISTAT FROM MORE = ',A6)
99 WRITE(06,553)
100 553 FORMAT('NO TAPE DRIVE - WILL REQUEST JOB PUT INTO HOLD QUEUE')
101 PAUSE 'PLEASE PLACE THIS JOB IN HOLD UNTIL TAPE DRIVE AVAILABLE'
102 GO TO 550
103 551 CONTINUE
104 REWIND 20
105 GO TO 50
106 555 CONTINUE
107 DO 556 I=1,6
108 556 CATFIL(I) = INBUFF(I)
109 CONTINUE
110 CALL ATTACH(20,CATFIL,I,0,ISTAT, )
111 IF(ISTAT.EQ.0) GO TO 50
112 WRITE(06,606) CATFIL,ISTAT

```

```

113         CALL ERRMSG(7,5901,5903)
114         GO TO 50
115     60 CONTINUE
116         NDI=0
117     301 NDI=NDI+1
118         IF(NDI.LE.NDP) GO TO 308
119         CALL ERRMSG(5,5901,5903)
120     308 READ(MOT,305,END=304,ERR=320) DNAM(NDI,1),DJAM(NDI,2),DQUAL(NDI),
121         X DFORM(NDI),DTYP(NDI),DTAB(NDI),DENT(NDI)
122     305 FORMAT(2(A6,2X),11,2X,12,2X,11,2X,A4,2X,12)
123         IF(DNAM(NDI,1).NE.'ZZZZZ') GO TO 301
124         READ(MOT,310,END=325,ERR=320) NFS
125     310 FORMAT(I2)
126         DO 315 I=1,NFS
127         READ(MOT,321,END=325,ERR=320) FORM(I)
128     321 FORMAT(A36)
129     315 CONTINUE
130         NDI=NDI-1
131         DO 309 I=1,NDI
132         WRITE(06,305)(DNAM(I,J),J=1,2),DQUAL(I),DFORM(I),DTYP(I),
133         X DTAB(I),DENT(I)
134     309 CONTINUE
135         WRITE(06,310) NFS
136         WRITE(06,321) (FORM(I),I=1,NFS)
137         GO TO 5
138     304 CALL ERRMSG(14,5901,5903)
139     320 CALL ERRMSG(13,5901,5903)
140     325 CALL ERRMSG(15,5901,5903)
141     5 CONTINUE
142 C ** SECTION B PERFORM RECURRING HOUSEKEEPING. OUTPUT DATA POINTS
143 C WILL BE ACCUMULATED IN ARRAY YDAT, INDEXED BY LINE NUMBER,
144 C THEN BY INTEGER X VALUE. PRINTER PAGE IMAGES WILL BE COLLECTED IN
145 C ARRAY IGRAPH (CGRAPH) INDEXED BY COLUMN NO. (6 COLUMNS PER
146 C COMPUTER WORD), THEN BY ROW NO.
147         WRITE(06,7)
148     7 FORMAT(1H1)
149         DO 3 J=1,6
150         HISLAB(J) = BLANK
151     3 CATFIL(J) = BLANK
152         DO 4 I=1,11
153         DO 4 J=1,100
154         XDAT(I,J)=0
155     4 YDAT(I,J)=0
156         DO 2 I=1,40
157         DO 2 J=1,100
158     2 PTDATA(I,J) = 0
159         XSCALE(1)=0
160         YSCALE(1)=0
161         YSCALE(11)=0
162         XSCALE(11)=0
163         IHDD=0
164         ISDFLT=0
165         IFEMP=0
166         NOTITL=0
167         NOYVAL=0
168         DO 10 I=1,22

```

```

169         DO 10 J=1,56
170         IGRAPH(I,J)=BLANK
171     10 CONTINUE
172         DO 15 I=1,6
173         DO 15 J=1,6
174     15 YLABLS(I,J)=BLANK
175         DO 40 I=6,22
176     40 IGRAPH(I,47)=YDASH
177 C ** SECTION C READ, INTERPRET, AND PROCESS THE COMMANDS IN THE
178 C COMMANDS FILE. AT EOF, TERMINATE ACTIVITY. AT FINAL COMMAND FOR
179 C A REPORT ('ERP'), BRANCH TO SECTION D
180     80 CONTINUE
181     READ(15,100,END=901,ERR=800) LINSIZ
182 100 FORMAT(22A6)
183     WRITE(06,100) LINSIZ
184     ICOMM = LINSIZ(1)
185     DO 90 I=1,6
186     90 CNBUFF(I) = LINSIZ(I+1)
187         IF(ICOMM.EQ.'RTI') GO TO 500
188         IF(ICOMM.EQ.'XAX') GO TO 505
189         IF(ICOMM.EQ.'YAX') GO TO 510
190         IF(ICOMM.EQ.'XSU') GO TO 520
191         IF(ICOMM.EQ.'YSL') GO TO 525
192         IF(ICOMM.EQ.'YSU') GO TO 530
193         IF(ICOMM.EQ.'XVB') GO TO 535
194         IF(ICOMM.EQ.'YVB') GO TO 540
195         IF(ICOMM.EQ.'ODS') GO TO 560
196         IF(ICOMM.EQ.'FMT') GO TO 565
197         IF(ICOMM.EQ.'HAX') GO TO 570
198         IF(ICOMM.EQ.'HDD') GO TO 575
199         IF(ICOMM.EQ.'ERP') GO TO 902
200         CALL ERRMSG(1,3901,3903)
201         GO TO 80
202     500 CONTINUE
203         NOTITL=NOTITL+1
204         DO 501 I=1,6
205         TITLE1(NOTITL*6-6+I)=INBUFF(I)
206         IGRAPH(10+I,52+NOTITL)=INBUFF(I)
207     501 CONTINUE
208         GO TO 80
209     505 CONTINUE
210         DO 506 I=1,6
211         XLAHL(I)=INBUFF(I)
212         IGRAPH(8+I,50)=INBUFF(I)
213     506 CONTINUE
214         GO TO 80
215     510 CONTINUE
216         NOYVAL=NOYVAL+1
217         DO 511 I=1,6
218         YLABLS(I,NOYVAL)=INBUFF(I)
219     511 CONTINUE
220         GO TO 80
221     520 CONTINUE
222         IF(CNBUFF(1).EQ.'D'.OR.CNBUFF(1).EQ.'DEFAULT') ISDFLT=1
223         IF(ISDFLT.EQ.1) GO TO 80
224     601 FORMAT(16)

```

```

225      DECODE(CNBUFF(1),601) XSCALE(11)
226      GO TO 80
227      525 CONTINUE
228      IF(ISDFLT.EQ.1) GO TO 80
229      DECODE(CNBUFF(1),601) YSCALE(1)
230      GO TO 80
231      530 CONTINUE
232      IF(ISDFLT.EQ.1) GO TO 80
233      DECODE(CNBUFF(1),601) YSCALE(11)
234      GO TO 80
235      535 CONTINUE
236      XVAR = CNBUFF(1)
237      N=0
238      GO TO 80
239      540 CONTINUE
240      N = N+1
241      IF(N.LE.NOYVAL) GO TO 541
242      CALL ERRMSG(8,3901,3903)
243      GO TO 80
244      541 YVAR(N) = CNBUFF(1)
245      DECODE(CNBUFF(2),542) QUAL(N)
246      542 FORMAT(16)
247      GO TO 80
248      560 CONTINUE
249      IPRINT=1
250      IPLOT=1
251      IF(CNBUFF(1).EQ.'CALCOM') IPRINT=0
252      IF(CNBUFF(1).EQ.'C') IPLOT=0
253      IF(CNBUFF(1).EQ.'PRINTE') IPLOT=0
254      IF(CNBUFF(1).EQ.'P') IPLOT=0
255      GO TO 80
256      565 CONTINUE
257      ITABLE=0
258      IHIST=1
259      IGRAF=1
260      IF(CNBUFF(1).EQ.'GRAPH') IHIST=0
261      IF(CNBUFF(1).EQ.'G') IHIST=0
262      IF(CNBUFF(1).EQ.'HISTOG') IGRAF=0
263      IF(CNBUFF(1).EQ.'H') IGRAF=0
264      IF(CNBUFF(1).NE.'TABLE') GO TO 80
265      ITABLE=1
266      IGRAF=0
267      IHIST=0
268      IPRINT=1
269      IPLOT=0
270      GO TO 80
271      570 CONTINUE
272      DO 571 I=1,6
273      571 HISLAB(I)=INBUFF(I)
274      GO TO 80
275      575 IF(CNBUFF(1).EQ.'YES') IHDD=1
276      GO TO 80
277      C IRRECOVERABLE I/O - TERMINATE PROGRAM
278      810 CALL ERRMSG(10,3901,3903)
279      C IRRECOVERABLE I/O - TERMINATE PROGRAM
280      CONTINUE

```

```

281          CALL ERRMSG(2,3901,3903)
282 C          TERMINATE PROGRAM
283 901 CONTINUE
284          CALL PLOTTR(0)
285          CALL TDUMP(25)
286          STOP
287 903 REWIND 20
288          GO TO 5
289 C ** SECTION D   COMPARE COMBINATION OF Y VARIABLE NAME, X VARIABLE
290 C   NAME AND QUALIFIER AGAINST DICTIONARY FOR VALIDITY OF REPORT
291 C   REQUEST.
292 902 CONTINUE
293 300 DO 350 J=1,NOYVAL
294          IQUAL(J)=0
295          IF(IQUAL(J).NE.0) IQUAL(J)=1
296          DO 340 I=1,NDI
297          IF(YVAR(J).NE.DNAM(I,1).OR.XVAR.NE.DNAM(I,2).OR.IQUAL(J).NE.
298 X IQUAL(I)) GO TO 340
299          IFORM(J)=DFORM(I)
300          ITYP(J)=DTYP(I)
301          ITAB(J)=DTAB(I)
302          IENT(J)=DENT(I)
303          GO TO 350
304 340 CONTINUE
305          WRITE(06,360) YVAR(J),XVAR,IQUAL(J),INBUFF(1)
306 360 FORMAT(2(A6,2X),I2,2X,'COMBINATION IS NOT LISTED IN THE',
307 1 'DICTIONARY. REPORT NUMBER',A6,' HAS BEEN OMITTED.')
308          CALL ERRMSG(6,3901,3903)
309 C ** SECTION E   READ Y VARIABLE DATA FROM INPUT SOURCE.
310 350 CONTINUE
311          MIX=1
312          IF(XVAR.EQ.'INDEX') GO TO 452
313 400 READ(20,410,END=499,ERR=810) ITABT,INDX,DATV
314 410 FORMAT(A4,I2,A126)
315          IF(ITABT.NE.'DAY ') GO TO 400
316          IDAY=INDX
317          MIX=MAX0(MIX,IDAY)
318 415 READ(20,410,END=489,ERR=810) ITABT,INDX,DATV
319          IF(ITABT.NE.'DAY ') GO TO 405
320          IDAY=INDX
321          MIX=MAX0(MIX,IDAY)
322          GO TO 415
323 405 DO 420 J=1,NOYVAL
324          IF(ITABT.NE.ITAB(J)) GO TO 420
325          IF(IQUAL(J).EQ.0) GO TO 418
326          IF(QUAL(J).NE.INDX) GO TO 420
327 418 NF=IFORM(J)
328          FORMT=FORM(NF)
329          IF(ITYP(J).EQ.0) GO TO 450
330          DECODE(DATV,FORMT) (X(I),I=1,IENT(J))
331          GO TO 455
332 450 DECODE(DATV,FORMT) (IX(I),I=1,IENT(J))
333          X(IENT(J))=IX(IENT(J))
334 435 YDAT(J,IDAY)=X(IENT(J))
335 420 CONTINUE
336          GO TO 415

```

```

337      452 IDAY=0
338      IFLAG=0
339      450 READ(20,410)END=493,ERR=810) ITABT,INDX,DATV
340      IF(ITABT.NE.'DAY ') GO TO 450
341      455 I=1
342      DO 460 J=1,NQYVAL
343      IF(DUAL(J).EQ.1) GO TO 462
344      460 CONTINUE
345      GO TO 450
346      462 IFLAG=1
347      465 READ(20,410)END=489,ERR=810) ITABT,INDX,DATV
348      IF(ITABT.NE.'DAY ') GO TO 455
349      DO 470 J=1,NQYVAL
350      IF((ITABT.NE.ITAB(J).OR.DUAL(J).NE.IDAY) GO TO 470
351      MIX=MIX/(MIX+INDX)
352      NF=1+FORM(J)
353      FORM=FORM*NF
354      IF(I.IE(J).EQ.0) GO TO 480
355      DECODE(DATV,FORM) IX(I)=I+IENT(J)
356      GO TO 465
357      480 DECODE(DATV,FORM) IX(I)=I+IENT(J)
358      XIENT(J)=IX(IENT(J))
359      485 YDAT(I,INDX)=X(IENT(J))
360      470 CONTINUE
361      GO TO 465
362      490 WRITE(06,475) INBUFF(1)
363      475 FORMAT('EMERGENT NUMBER',I5,' HAS BEEN OMITTED')
364      CALL ERRMSG(0,4901,4905)
365      495 IF(IDAY.EQ.0) CALL ERRMSG( 9,4901,4905)
366      IF (IFLAG.NE.0) GO TO 489
367      WRITE(06,484) INBUFF(1)
368      484 FORMAT('EMERGENT NUMBER',I6,' HAS BEEN OMITTED SINCE')
369      CALL ERRMSG(11,4901,4905)
370 C ** SECTION 4 ** CALCULATE MINIMUM AND MAXIMUM VALUES OF THE Y VARIABLE
371 C DATA. USE MINIMUM AND MAXIMUM VALUES TO SET THE SCALES. IF THE
372 C USER HAS DEFAULTED OR IF THE USER CHOSE SCALE VALUES TOO SMALL FOR
373 C THE DATA.
374      489 CONTINUE
375      YMIN=YDAT(1,1)
376      YMAX=YDAT(1,1)
377      DO 490 J=1,NQYVAL
378      DO 490 I=1,MIX
379      YMIN=MIN(YMIN,YDAT(J,I))
380      YMAX=MAX(YMAX,YDAT(J,I))
381      490 CONTINUE
382      IYMIN=YMIN
383      IYMAX=YMAX
384      IYMIN=IYMIN
385      IYMAX=IYMAX
386      IF(IYMAX.NE.YMAX) IYMAX=IYMAX*1
387      IF(IYMIN.NE.YMIN) IYMIN=IYMIN*1
388      X YSCALE(11) GO TO 497
389      CALL ERRMSG(6,4901,4905)
390      495 XSCALE(11)=MIX
391      XSCALE(11)= (XSCALE(11)+9)/10)*10
392      XINC=XSCALE(11)/10

```

```

395      DO 486 I=1,6
396      J=1
397      486 IF(IYMAX.GT.UNIT(I)) GO TO 487
398      487 YSCALE(11)=YSCALE(11)+UNIT(J)
399      IF(YSCALE(11).LT.IYMAX) GO TO 487
400      YINC=YSCALE(11)/10
401      GO TO 496
402      497 XSCALE(11)=((XSCALE(11)+9)/10)*10
403      XINC=XSCALE(11)/10
404      YSCALE(11)=(YSCALE(11)/10)*10
405      YSCALE(11)=((YSCALE(11)+9)/10)*10
406      YINC=(YSCALE(11)-YSCALE(11))/10
407      496 CONTINUE
408      DO 498 I=2,10
409      XSCALE(I)=XSCALE(I-1)+XINC
410      498 YSCALE(I)=YSCALE(I-1)+YINC
411      WRITE THE DATA TO BE PLOTTED.
412      WRITE(06,482)
413      482 FORMAT('Y-DATA')
414      DO 481 J=1,NOYVAL
415      DO 481 I=1,MIX
416      WRITE(06,483) J,I,YDAT(J,I)
417      483 FORMAT(2(2X,I2),2X,F10.2)
418      481 CONTINUE
419      910 REWIND 20
420      C ** SECTION G ALL DATA VALUES HAVE BEEN ACCUMULATED IN ARRAY YDAT.
421      C EACH DATA VALUE IS CONVERTED FOR THE DATA POINT MATRIX PTDATA.
422      C PTDATA IS DIMENSIONED BY 40 FOR THE ROWS OF POINTS ON A PRINTER
423      C PAGE , THEN BY 100 FOR THE COLUMNS OF POINTS ON A PRINTER PAGE.
424      DO 200 N=1,NOYVAL
425      I = NOYVAL +1 -N
426      DO 200 J=1,100
427      IF(YDAT(I,J).EQ.0.0) GO TO 200
428      RATIO=(YDAT(I,J)-YSCALE(11))/(YSCALE(11)-YSCALE(1))
429      IF(RATIO.LT.0.0) CALL ERRMSG(4,5901,5903)
430      IF(RATIO.GT.1.0) CALL ERRMSG(4,5901,5903)
431      Z=RATIO*40.0
432      K=Z+0.5
433      IF(K.LE.0) K=1
434      IF(K.GT.40) K=40
435      IF(PTDATA(K,J).NE.0) CALL ERRMSG(5,5901,5903)
436      PTDATA(K,J)=1
437      200 CONTINUE
438      IF(CIPLT.EQ.0) GO TO 920
439      C ** SECTION H SET UP AND CALL SUBROUTINE PLOTTER FOR LINE GRAPHS AND
440      C HISTOGRAMS.
441      DO 905 I=1,6
442      905 NBR(I) = MIX
443      IF(IGRAF.EQ.0) GO TO 915
444      C LINE GRAPH REQUIRED
445      TYPE = 1
446      LINES = NOYVAL
447      CALL PLOTTR(NUM)
448      NUM = NUM+1
449      915 IF(IHIST.EQ.0) GO TO 920
450      C HISTOGRAM REQUIRED

```

AD-A040 634

COMPUTER SCIENCES CORP ARLINGTON VA
DESCRIPTION OF THE REMOTE TERMINAL EXECUTIVE (REMTEX) POSTPROCE--ETC(U)
JAN 77 W H POLLITT, M C FLYTHE

F/G 9/2

DCA100-74-C-0002

NL

UNCLASSIFIED

CCTC-TM-144-77

2 OF 2
ADA
040634



END

DATE
FILMED
7-77



```

449         TYPE = 2
450         CALL PLOTTR(NUM)
451         NUM = NUM+1
452     920 IF(IPRINT,EQ,0) GO TO 5
453 C ** SECTION 1 SET UP PRINT PAGE IMAGES AND WRITE POINT GRAPHS,
454 C HISTOGRAMS AND TABLES TO THE ONLINE PRINTER.
455 C
456 C SET UP Y SCALE INCREMENTS AND AXIS
457 DO 120 I=1,11
458 ENCODE(CGGRAPH(5,3+I*4),700) YSCALE(12-I)
459 700 FORMAT(I3)
460 K=YSCALE(12-I)/1000
461 IF(K,EQ,0) GO TO 120
462 ENCODE(CGGRAPH(4,3+I*4),711) K
463 120 CONTINUE
464 DO 122 I=7,47
465 122 FLD(10,6,1)IGRAPH(5,I) = FLD(30,6,DAP)
466 DO 124 I=7,47,4
467 124 FLD(24,6,1)IGRAPH(5,I) = FLD(24,6,DAP)
468 IF(IHIST,EQ,0) GO TO 225
469 C SET UP HISTOGRAM
470 IF(HISLAB(1),NE,BLANK) GO TO 204
471 DO 203 I=1,6
472 203 HISLAB(I) = YLABLS(J,1)
473 204 CONTINUE
474 DO 205 I=1,6
475 205 DECODE(CHISLB(I),610) (ICHAR(J,1),J=I+6-5,36)
476 DO 206 I=1,6
477 206 ENCODE(CGGRAPH(3,I),619) ICHAR(I-10,1)
478 619 FORMAT(SX,A1)
479 DO 210 I=1,XSCALE(11),16
480 DO 211 J=6,22
481 DO 211 K=6,46
482 IGRAPH(J,K) = BLANK
483 211 IGRAPH(J,K) = BLANK
484 DO 212 J=6,21
485 K= I+J-6
486 IF(K,GT,XSCALE(11)) GO TO 220
487 IF(K,GT,MIX) GO TO 220
488 ENCODE(CGGRAPH(J,48),250) K
489 250 FORMAT(1X,I3,2X)
490 DO 213 L=7,46
491 IF(PIDATA(47-L,K),NE,1) GO TO 213
492 IGRAPH(J,L) = YDASH
493 IF(IH00,EQ,0) GO TO 215
494 ITEMP = YDAT(1,K)
495 ENCODE(CGGRAPH(J,L-1),711) ITEMP
496 215 CONTINUE
497 IF(L,EQ,46) GO TO 212
498 DO 214 M=L+1,46
499 IF(J,NE,6) IGRAPH(J,M) = APOS1
500 214 IGRAPH(J+1,M) = APOS1
501 GO TO 212
502 213 CONTINUE
503 212 CONTINUE
504 C OUTPUT HISTOGRAM

```

```

505          WRITE(06,101)
506      101 FORMAT(1H1)
507          WRITE(06,100)   IGRAPH
508      210 CONTINUE
509      C   OUTPUT FINAL PAGE OF HISTOGRAM
510      220 WRITE(06,101)
511          WRITE(06,100)   IGRAPH
512      225 IF(IGRAF.EQ.0)  GO TO 950
513      C   SET UP POINT GRAPH.
514      C
515      C   SET UP Y AXIS LABELS.
516          DO 546  I=1,6
517          DO 546  J=1,6
518      546 DECODE(CYLABL(J,I),610)  ICHAR(J*6-5,I),ICHR(J*6-4,I),
519      *      ICHAR(J*6-3,I),ICHR(J*6-2,I),ICHR(J*6-1,I),ICHR(J*6,1)
520      610 FORMAT(6A1)
521          DO 547  I=11,46
522          ENCODE(CGGRAPH(1,I),620)  ICHAR(I-10,1),ICHR(I-10,2)
523          ENCODE(CGGRAPH(2,I),620)  ICHAR(I-10,3),ICHR(I-10,4)
524      547 ENCODE(CGGRAPH(3,I),620)  ICHAR(I-10,5),ICHR(I-10,6)
525      620 FORMAT(2X,A1,2X,A1)
526          IGRAPH(1,8)=ONE
527          IGRAPH(1,9)=UNDER1
528          IF(NOYVAL.EQ.1) GO TO 590
529          IGRAPH(1,8)=ONETWO
530          IGRAPH(1,9)=UNDER2
531          IF(NOYVAL.EQ.2) GO TO 590
532          IGRAPH(2,8)=THREE
533          IGRAPH(2,9)=UNDER1
534          IF(NOYVAL.EQ.3) GO TO 590
535          IGRAPH(2,8)=TREFOR
536          IGRAPH(2,9)=UNDER2
537          IF(NOYVAL.EQ.4) GO TO 590
538          IGRAPH(3,8)=FIVE
539          IGRAPH(3,9)=UNDER1
540          IF(NOYVAL.EQ.5) GO TO 590
541          IGRAPH(3,8)=FIVSIX
542          IGRAPH(3,9)=UNDER2
543      C   SET UP X SCALE INCREMENTS
544      590 CONTINUE
545          DO 548  I=6,22,5
546          IGRAPH(I+1,47)=TICK4
547          IF(I.GE.21) GO TO 548
548          IGRAPH(I+3,47)=TICK2
549          IGRAPH(I+4,47)=TICK6
550      548 CONTINUE
551          DO 549  I=6,22
552      549 IGRAPH(I,48)=BLANK
553          J=1
554          DO 130  I=5,22,5
555          ENCODE(CGGRAPH(I,48),711) XSCALE(J)
556      711 FORMAT(16)
557      130 J=J+3
558          J=2
559          DO 140  I=7,22,5
560          ENCODE(CGGRAPH(I,48),712) XSCALE(J)

```

```

561       712 FORMAT(I4)
562       140 J=J+3
563         J=5
564         DO 150 I=9,22,5
565           ENCODE(CGRAPH(I,48),713) XSCALE(J)
566       713 FORMAT(I2)
567       150 J=J+5
568     C     SET UP DATA POINTS
569         DO 230 I=6,22
570         DO 230 J=6,46
571       230 IGRAPH(I,J) = BLANK
572         DO 235 I=1,40
573         DO 235 J=1,100
574         IF(PTDATA(I,J).EQ.0) GO TO 235
575         K = (J+10)/XINC
576         IF(K.EQ.0) K=1
577         Z = 6.0+ (K-1)/6.0
578         L = Z
579         A = 36.0*(Z-L)
580         M=A+0.1
581         FLD(M,6,IGRAPH(L,47-1)) = FLD(30,6,PTDATA(I,J))
582       235 CONTINUE
583     C     OUTPUT POINT GRAPH
584         WRITE(06,101)
585         WRITE(06,100) IGRAPH
586       950 IF(ITABLE.EQ.0) GO TO 5
587     C     SET UP TABLE
588         IPAGE = 0
589         LINECT=1
590     C     SET UP AND OUTPUT HEADERS
591       951 CONTINUE
592         IPAGE = IPAGE + 1
593         WRITE(06,101)
594         WRITE(06,650) (TITLE1(I),I=1,6),IPAGE
595       650 FORMAT(48X,6A6,25X,5HPAGE,I2)
596         WRITE(06,651) (TITLE2(I),I=1,6)
597         WRITE(06,651) (TITLE3(I),I=1,6)
598       651 FORMAT(48X,6A6)
599         WRITE(06,655) (XLABL(I),I=1,3),((YLABLS(J,K),J=1,3),K=1,5),
600         1 YLABLS(1,6),YLABLS(2,6)
601       655 FORMAT(/6X,3A6,6X,5(3A6),2A6)
602     C     OUTPUT DATA LINES
603       955 WRITE(06,660) LINECT,(YDAT(I,LINECT),I=1,NOYVAL)
604       660 FORMAT(/6X,16,12X,6(6X,F12.1))
605         IF(LINECT.EQ.MIX) GO TO 5
606         LINECT = LINECT + 1
607         I=(LINECT/25)*25
608         J=(LINECT*25)/25
609         IF(I.EQ.J) GO TO 951
610         GO TO 955
611       END

```

```

1      SUBROUTINE PLOTTR(NUM)
2      COMMON/PLOTTR/ XDAT(11,100),YDAT(11,100),NBR(11),
3      *      YLABLS(10),XLABL,LINES,TYPE,TITLE,TITLE1,TITLE2,TITLES,
4      *      XSCALE(11),YSCALE(11)
5      INTEGER XSCALE,YSCALE
6      DIMENSION X(1002),XPLT(102),YPLT(102),Y(1100)
7      EQUIVALENCE (X(1),XDAT(1,1))
8      DIMENSION YPOS(4)
9      CHARACTER YLABLS*36,XLABL*36,TITLE*36,TITLE1*36,TITLE2*36,
10     *      TITLES*36
11     INTEGER TYPE
12     DATA BLANK/' ',YPOS/-30.0,10.0,10.0,10.0/,XPOS/0.0/
13     DATA XLEN/5.0/,YLEN/5.0/
14     C      IF TYPE = 1 LINE GRAPH
15     C      IF TYPE = 2 HISTOGRAM
16     C      IF TYPE = 1
17     C      XDAT(N,M) CONTAINS X-COORDINATE ,M DATA POINTS,N LINES
18     C      YDAT(N,M) CONTAINS Y-COORDINATE ,M DATA POINTS,N LINES
19     C      NBR(N) NUMBER OF DATA POINTS FOR EACH LINE
20     C      YLABLS(N) Y-AXIS LABELS FOR EACH LINE
21     C      XLABL X-AXIS LABEL
22     C      TITLE TITLE OF PLOT
23     C      XLEN LENGTH OF X-AXIS
24     C      YLEN LENGTH OF Y-AXIS
25     C      LINES NUMBER OF LINES TO BE PLOTTED
26     C
27     C      IF TYPE = 2
28     C      YDAT(1,100) CONTAINS VALUE OF EACH BAR IN HISTOGRAM
29     C      NBR(1) CONTAINS NUMBER OF BARS TO BE PLOTTED
30     C      XLEN LENGTH OF X-AXIS
31     C      YLEN LENGTH OF Y-AXIS
32     C      YLABLS(1) Y-AXIS LABEL
33     C      TITLE TITLE OF PLOT
34     C
35     C
36     C      *** NOTE ***
37     C
38     C      XDAT AND YDAT ARRAYS MUST BE INITIALIZED TO ZERO
39     C
40     C      BEFORE DATA IS PLACED IN THE ARRAYS FOR EACH PLOT
41     C      IF(NUM .EQ. 0) GO TO 1000
42     C      DO 1 I=1,NBR(1)
43     C      DO 1 J=1,6
44     C      1 XDAT(J,I) = I
45     C      IF(TYPE .EQ. 2) GO TO 100
46     C      *** SET UP AND WRITE LINE GRAPH TO PLOTTER TAPE
47     C      DO 10 I=1,100
48     C      DO 10 J=1,11
49     C      10 Y(J+(I-1)*11) = YDAT(J,I)
50     C      X(1001) = XSCALE(1)
51     C      X(1002) = (XSCALE(11) - XSCALE(1))/5
52     C      XX = (XLEN-3.5)*0.5
53     C      CALL NEWAX(0.0,0.0,XLEN,0.0,X(1001),X(1002))
54     C      CALL SYMBOL(XX,-0.475,0.10,XLABL,0.0,36)
55     C      CALL SYMBOL(XX,YLEN+0.50,0.10,TITLE1,0.0,36)
56     C      CALL SYMBOL(XX,YLEN+0.35,0.10,TITLE2,0.0,36)

```

```

57      CALL SYMBOL (XX, YLEN+0.20, 0.10, TITLE 3, 0.0, 36)
58      Y(1001) = YSCALE(1)
59      Y(1002) = (YSCALE(11)-YSCALE(1))/5
60      CALL NEWAX(0.0, 0.0, YLEN, 90., Y(1001), Y(1002))
61      XX = -0.475
62      YY = (YLEN-3.5)*0.5
63      DO 3 I=1, LINES
64      J=1
65      CALL SYMBOL (XX, YY, 0.10, YLABLS(I), 90., 36)
66      CALL SYMBOL (XX, YY+3.7, 0.1, J, 90., 0., -1)
67      XX = XX-0.15
68      3 CONTINUE
69      DO 4 I=1, LINES
70      DO 5 J=1, NBR(I)
71      XPLT(J) = XDAT(I, J)
72      YPLT(J) = YDAT(I, J)
73      5 CONTINUE
74      XPLT(NBR(I)+1) = X(1001)
75      XPLT(NBR(I)+2) = X(1002)
76      YPLT(NBR(I)+1) = Y(1001)
77      YPLT(NBR(I)+2) = Y(1002)
78      K=I
79      CALL LINE(XPLT, YPLT, NBR(I), 1, 1, K)
80      4 CONTINUE
81      7 I = MOD(NUM, 4)
82      XPOS = 0.0
83      IF (I.EQ.0) XPOS = 10.0
84      CALL PLOT(XPOS, YPOS(I+1), -3)
85      RETURN
86      C*** SET UP AND WRITE HISTOGRAM TO PLOTTER TAPE
87      100 CONTINUE
88      DO 11 I=1, 1100
89      11 Y(I)=0.0
90      DO 12 J=1, 100
91      12 Y(J) = YDAT(I, J)
92      Y(101) = YSCALE(1)
93      Y(102) = (YSCALE(11)-YSCALE(1))/5
94      CALL NEWAX(0.0, 0.0, YLEN, 90., Y(101), Y(102))
95      XX = (XLEN - 3.5) * 0.5
96      CALL SYMBOL (XX, -0.475, 0.10, XLABL, 0.0, 36)
97      CALL SYMBOL (XX, YLEN+0.50, 0.10, TITLE 1, 0.0, 36)
98      CALL SYMBOL (XX, YLEN+0.35, 0.10, TITLE 2, 0.0, 36)
99      CALL SYMBOL (XX, YLEN+0.20, 0.10, TITLE 3, 0.0, 36)
100     XX = 0.
101     YY = 0.
102     WIDTH = XLEN/NBR(1)
103     DO 6 I = 1, NBR(1)
104     HGT = ( Y(I) - Y(101) ) / Y(102)
105     CALL BAR (XX, YY, 0., HGT, WIDTH, 0., 1, 0)
106     XXX = XX + 0.3 * WIDTH
107     A = I
108     CALL NUMBER (XXX, -0.1, 0.07, A, 0., -1)
109     XX = XX + WIDTH
110     6 CONTINUE
111     YY = (YLEN - 3.5) * 0.5
112     CALL SYMBOL (-0.475, YY, 0.10, YLABLS(1), 90.0, 36)

```

```
113          GO TO 7
114 C*** CLOSE OUT PLOTTER TAPE
115 1000 CALL PLOT(0.,0.,999)
116 RETURN
117 END
```

```

1      SUBROUTINE NEWAX(XPAGE,YPAGE,AXLEN,ANGLE,FIRSTV,DELTAV)
2      ANGL=ANGLE
3      IF(ANGL.EQ.90.) ANGL=1.57080
4      IS=FIRSTV
5      IDEL=DELTAV
6      IPT=IS
7      X2=XPAGE+AXLEN*COS(ANGL)
8      Y2=YPAGE+AXLEN*SIN(ANGL)
9      CALL PLOT(XPAGE,YPAGE,3)
10     CALL PLOT(X2,Y2,2)
11     N=AXLEN
12     DELX=.05*SIN(ANGL)
13     DELY=.05*COS(ANGL)
14     X2=XPAGE;Y2=YPAGE
15     CALL PLOT(X2,Y2,3)
16     DO 10 I=1,N+1
17     CALL PLOT(X2-DELX,Y2-DELY,2)
18     ITEST1=0
19     ITEST2=10
20     DO 20 M=1,9
21     XN=M
22     IF(IPT.GE.ITEST1.AND.IPT.LT.ITEST2) GO TO 30
23     ITEST1=ITEST2
24     ITEST2=ITEST2*10
25     XN=0
26     CONTINUE
27     IF(XN.NE.0.) GO TO 30
28     WRITE(06,12)
29     12 FORMAT('SCALE VALUES TOO LARGE TO PRINT')
30     CALL PLOT(0.,0.,999)
31     CALL TDUMP(25)
32     STOP
33     30 XN=XN/2.
34     X1=X2-.18*SIN(ANGL)-XN*.1*COS(ANGL)
35     Y1=Y2-.18*COS(ANGL)-XN*.1*SIN(ANGL)
36     FPN=IS+(I-1)*IDEL
37     CALL NUMBER(X1,Y1,.1,FPN,ANGLE,-1)
38     IF(I.EQ.N+1) GO TO 10
39     IPT=IPT+IDEL
40     X2=X2+1.*COS(ANGL)
41     Y2=Y2+1.*SIN(ANGL)
42     CALL PLOT(X2,Y2,3)
43     10 CONTINUE
44     RETURN
45     END

```

```

1      SUBROUTINE ERRMSG(IERR,B,B)
2      CHARACTER INBUFF*6(15)
3      IF(IERR.GE.1.AND.IERR.LE.16) GO TO 10
4      WRITE(06,1000)
5      1000 FORMAT(1X,'WARNING--ERRMSG ARGUMENT IS OUT OF RANGE-CHECK ',
6      * 'RXPROC.')
7      RETURN
8      10 GO TO (11,12,13,14,15,16,17,18,19,20,21,22,23,24,25),IERR
9      11 WRITE(06,1001)
10     1001 FORMAT(1X,'WARNING--ILLEGAL COMMAND ON COMMANDS FILE')
11     RETURN
12     12 WRITE(06,1002)
13     1002 FORMAT(1X,'IRRECOVERABLE I/O ERROR ON COMMANDS FILE. PROGRAM ',
14     1 'TERMINATED.')
15     RETURN 1
16     13 WRITE(06,1003)
17     1003 FORMAT(1X,'NUMBER OF DICTIONARY ITEMS EXCEEDS STORAGE SPACE.',
18     2 'PROGRAM TERMINATED.')
19     RETURN 1
20     14 WRITE(06,1004)
21     1004 FORMAT(1X,'WARNING--SCALE RANGE TOO SMALL. DEFAULT RANGE USED')
22     RETURN
23     15 WRITE(06,1005)
24     1005 FORMAT(1X,'WARNING--TWO OR MORE DATA POINTS OCCUPY SAME POSITION')
25     RETURN
26     16 RETURN 2
27     17 WRITE(06,1007)
28     1007 FORMAT(1X,'UNABLE TO ATTACH SPECIFIED PERM FILE. ',
29     1 'PROGRAM TERMINATED.')
30     RETURN 1
31     18 WRITE(06,1008)
32     1008 FORMAT(1X,'WARNING--THE NUMBER OF Y VARIABLE NAMES EXCEEDS THE ',
33     5 'NUMBER OF Y TITLES. EXTRA Y VARIABLE NAMES HAVE BEEN IGNORED.')
34     RETURN
35     19 WRITE(06,1009)
36     1009 FORMAT(1X,'WORD DAY NOT ENCOUNTERED ON FILE 20. PROGRAM ',
37     4 'TERMINATED.')
38     RETURN 1
39     20 WRITE(06,1010)
40     1010 FORMAT(1X,'IRRECOVERABLE I/O ERROR ON DATA FILE 20. PROGRAM ',
41     5 'TERMINATED.')
42     RETURN 1
43     21 WRITE(06,1020)
44     1020 FORMAT(1X,'APPROPRIATE VALUES FOR DAY NOT FOUND ON DATA FILE 20.')
45     RETURN 2
46     22 WRITE(06,1030)
47     1030 FORMAT(1X,'WARNING--ILLEGAL COMMAND ON FILE POINTERS FILE ')
48     RETURN
49     23 WRITE(06,1040)
50     1040 FORMAT('IRRECOVERABLE I/O ERROR ON DICTIONARY. PROGRAM TERMINATED'
51     * )
52     RETURN 1
53     24 WRITE(06,1050)
54     1050 FORMAT('ABNORMAL ENDING FOR ITEMS IN THE DICTIONARY. PROGRAM TERMI
55     *NATED')
56     RETURN 1

```

```

57      25 WRITE(06,1060)
58 1060 FORMAT('INCORRECT ENTRY OF DATA DESCRIBING FORMATS ASSOCIATED WITH
59 * THE DICTIONARY. PROGRAM TERMINATED')
60      RETURN 1
61      50 READ(15,2000,END=60,ERR=65) INBUFF
62 2000 FORMAT(1A6,12A6)
63      GO TO 70
64      60 WRITE(06,2040)
65 2040 FORMAT('NO ERP COMMAND FOUND. PROGRAM TERMINATED')
66      RETURN 1
67      65 WRITE(06,1100)
68 1100 FORMAT(1X, '****ERROR CONDITION ENCOUNTERED WHILE READING ',
69 6 'COMMANDS FILE. PROGRAM TERMINATED.')
70      RETURN 1
71      70 IF(INBUFF(1).NE.'ERP') GO TO 50
72      DECODE(INBUFF(2),2020) IERP
73 2020 FORMAT(I6)
74      WRITE(06,2030) IERP
75 2030 FORMAT('REPORT NUMBER',I3,' HAS BEEN OMITTED')
76      RETURN 2
77      END

```

```
1      SYMDEF  MORE
3      PCC     ON
4      REFMA   ON
5 MORE  SAVE   0,1,2,3,4,5,6,7
```

```
6 .....
7 *
8 *      CALL      MORE USING TYPE,NOLIST,ERROR
9 *
10 *      NOLIST =  NUMBER OF INTERIES IN THE LIST
11 *
12 *
13 *      TYPE     =  TORUFC
14 *
15 *              T = 1 - TAPE7
16 *              T = 3 - TAPE9
17 *              T = 2 - DISK
18 *
19 *              R = 1 - RANDOM DISK FILE
20 *
21 *              FC = TWO (2) CHARACTER FILE CODE
22 *
23 *      NEXT WORD - RIGHT ADJUSTED
24 *                = FIVE (5) CHARACTER REEL NUMBER
25 *      OR       = NUMBER OF LINKS REQUESTED
26 *
27 *
28 *
29 *
30 .....
```

000024	000003	2270	31	000	31	EJECT	
000025	000002	2260	11	000	32	LDX7	3,1*
		000026			33	LDX6	2,1
000026	000000	2360	16	000	34	NEXT	NULL
000027	000072	2350	00	010	35	LDQ	0,6
000030	000006	7370	00	000	36	LDA	=6H000000
000031	000022	7350	00	000	37	LLS	6
000032	000045	7510	70	010	38	ALS	18
000033	000002	1150	03	000	39	STCA	A,70
000034	000056	6000	00	010	40	CMPA	2,00
000035	000001	2250	03	000	41	TZE	DISK
000036	000045	4450	00	010	42	LDX5	1,00
000037	000001	6350	16	000	43	SXLS	B
000040	000022	7710	00	000	44	EAA	1,6
000041	000014	7360	00	000	45	ARK	18
000042	000022	7730	00	000	46	QLS	12
000043	000004	7560	31	000	47	LRL	18
		000044			48	STQ	4,1*
000044	000011	0010	00	000	49	SAME	NULL
		000045			50	MME	GEMURE
000045	000000	000000	000	000	51	A	NULL
		000045			52	ZERO	0,0
		000045			53	ORG	A
000045	000000	000000	000	000	54	B	NULL
000046	000066	7100	00	010	55	ZERO	0,0
000047	000002	6260	16	000	56	TRA	ERROR
000050	777777	6270	17	000	57	EAX6	2,6
000051	000054	6000	00	010	58	EAX7	-1,7
000052	000054	6040	00	010	59	TZE	DONE
000053	000026	7100	00	010	60	TMI	DONE
		000054			61	TRA	NEXT
000054	000004	4500	31	000	62	DONE	NULL
000055	000001710000	010			63	STZ	4,1*
		000056			64	RETURN	MORE
000056	000001	2350	16	000	65	DISK	NULL
000057	000045	7510	07	010	66	LDA	1,6
000060	000006	7720	00	000	67	STCA	B,07
000061	000001	2350	16	000	68	QRL	6
000062	000045	7510	07	010	69	LDA	1,6
000063	000044	7100	00	010	70	STCA	B,07
000064	000004	4500	31	000	71	TRA	SAME
000065	000001710000	010			72	STZ	4,1*
		000066			73	RETURN	MORE
000066	000001710000	010			74	ERROR	NULL
					75	RETURN	MORE

ERROR LINKAGE

000067	000000000000	000
000070	444651252020	000

LITERALS

000072	000000000000	000
--------	--------------	-----

76 END

DISTRIBUTION

ADDRESSEE	COPIES
CCTC Codes	
C124 (Reference and Record Set).....	3
C124 (Stock).....	6
C126 (ATTN: Valerie Palmer).....	2
C315	15
DCA Codes	
205.....	1
Defense Documentation Center, Cameron Station Alexandria, Virginia, 22314.....	12
Studies, Analysis, and Gaming Agency, SSG The Pentagon, Washington, D.C. 20301.....	5
Computer Sciences Corporation, ATTN: Mr. Harry Pollitt, 400 Army-Navy Drive Arlington, Virginia 22202.....	4
	48

REPORT DOCUMENTATION PAGE		HEADLINE CATEGORIES 1. GOVT ACQUISITION 2. REPORTS CATALOG
19 TM-144-77	18 CCTC	
4. TITLE (Include Subtitle) Description of the Remote Terminal Executive (REMTEX) Postprocessor.		9 Technical memo.
7. AUTHOR (Last, First, Middle Initial) Pollitt, William Harold / Pollitt Flythe, Mary Catherine / Flythe		15 DCA 100-74-C-0002
9. PERFORMING ORGANIZATION NAME & ADDRESS Computer Sciences Corporation 400 Army Navy Drive Arlington, VA 22202		10. PROGRAM ELEMENT, REPORT NUMBER, AREA & WORK UNIT (If Applicable)
11. CONTROLLING OFFICE NAME & ADDRESS Command and Control Technical Center (CCTC) The Pentagon Washington, DC 20301		12. REPORT DATE 11 10 January 1977
14. MONITORING AGENCY NAME & ADDRESS (If different from Controlling Office) 12 109p.		13. NUMBER OF PAGES 112
		15. SECURITY CLASS. for this report Unclassified
		15a. DECLASS/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this report) Copies of this document may be obtained from the Defense Documentation Center, Cameron Station, Alexandria, VA 22314. Approved for public release; distribution unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in block 20, if different from report)		
18. SUPPLEMENTARY NOTES Generalized postprocessor, search and retrieval, graphic output formatter, CalComp 7000, graph, histogram, tabular, interactive report request.		
19. KEY WORDS (continue on reverse side if necessary and identify by block number)		
20. ABSTRACT (continue on reverse side if necessary and identify by block number) This Technical Memorandum (TM) describes the newly developed REMote Terminal EXECutive (REMTEX) Postprocessor software system. User interaction and input requirements are described and output formats are presented. Basic program and file maintenance instructions are included.		