NASA TECHNICAL MEMORANDUM

NASA TM X-64862

GSE DATA MANAGEMENT SYSTEM PROGRAMMERS'/USERS' MANUAL

By Ronald A. Schlagheck, Bert D. Dolerhie, Jr., and Fred J. Ghiglieri
Systems Analysis and Integration Laboratory

June 1974

NASA

George C. Marshall Space Flight Center
Marshall Space Flight Center, Alabama

TECHNICAL REPORT STANDARD TITLE PAGE

1. REPORT NO.	2. GOVERNMENT ACCESSION NO.	3. RECIPIENT'S CATALOG NO.
NASA TM X-64862		

4. TITLE AND SUBTITLE	5. REPORT DATE
GSE Data Management System Programmers'/Users' Manual	June 1974
	6. PERFORMING ORGANIZATION CODE

7. AUTHOR(S)	8. PERFORMING ORGANIZATION REPORT #
Ronald A. Schlagheck, Bert D. Dolerhie, Jr., Fred J. Ghiglieri*	

9. PERFORMING ORGANIZATION NAME AND ADDRESS	10. WORK UNIT NO.
George C. Marshall Space Flight Center Marshall Space Flight Center, Alabama 35812	11. CONTRACT OR GRANT NO.
	13. TYPE OF REPORT & PERIOD COVERED
12. SPONSORING AGENCY NAME AND ADDRESS	Technical Memorandum
National Aeronautics and Space Administration Washington, D.C. 20546	14. SPONSORING AGENCY CODE

15. SUPPLEMENTARY NOTES

Prepared by Systems Analysis and Integration Laboratory, Science and Engineering
*Teledyne Brown Engineering Employee

16. ABSTRACT

The GSE Data Management System is a computerized program which provides for a central storage source for key data associated with the mechanical ground support equipment (MGSE). Eight major sort modes can be requested by the user. Attributes that are printed automatically with each sort include the GSE End Item number, description, class code, functional code, fluid media, use location, design responsibility, weight, cost, quantity, dimensions, and applicable documents. Multiple subsorts are available for the class code, functional code, fluid media, use location, design responsibility, and applicable document categories. This manual includes a description of these sorts and how to use them. The program and GSE data bank may be easily updated and expanded.

17. KEY WORDS	18. DISTRIBUTION STATEMENT
Mechanical Ground Support Equipment (MGSE) Computerized Data Management Systems Computer Inventory Programs Computerized Sorting Systems Launch Facilities	Unclassified-unlimited O. M. Hardage O. M. HARDAGE Chief, Operations Planning and Analysis Branch

19. SECURITY CLASSIF. (of this report)	20. SECURITY CLASSIF. (of this page)	21. NO. OF PAGES	22. PRICE
Unclassified	Unclassified	108	NTIS

For sale by National Technical Information Service, Springfield, Virginia 22151

TABLE OF CONTENTS

Section		Page
I.	INTRODUCTION	1
II.	PROGRAM OPERATION INSTRUCTIONS	3
	Control Card Description	3
	GSE Data Card Input	6
	Program Instruction Data Input	6
III.	SAMPLE OUTPUT	11
	Description of Sort Types	11
	User Messages	17
	Error Messages	22
IV.	PROGRAM ASPECTS	27
	Program Execution	27
	Program Modification Procedures	29
	Changes to Program Procedures Tables	30
	Changing and Adding Fortran Routines	33
	Deck Setup, Control Cards, and Options	34
	Program Characteristics	36
	Description of the Program Listing	36
APPENDIX A.	DEFINITION OF INPUT FORMAT FOR GSE ATTRIBUTE DATA CARDS	37
	Card Type 1	39
	Card Type 2	42
	Card Type 3	43
	Card Type 4	47
APPENDIX B.	GSE DATA MANAGEMENT SYSTEM PROGRAM LISTING	49
APPENDIX C.	SUPPLEMENT RUN PROCEDURES	99

LIST OF ILLUSTRATIONS

Figure	Title	Page
1.	Control card deck setup	4
2.	Pictorial control card deck setup	5
3.	Control card deck with GSE data input and all program instruction data types	7
4.	Control card deck with multiple sorts of the same mnemonic code	9
5.	Example page of Master Summary Table	12
6.	Example of Stage Sort One Table	13
7.	Example of Stage Sort Two Table	14
8.	Example of Media Sort Table	15
9.	Example of Functional Sort Table	16
10.	Example of Class Code Sort Table	18
11.	Example of Use Location Sort Table	19
12.	Example of Applicable Document Sort Table	20
13.	Master program execution flow chart	28
14.	Procedure Table catalogue listing	31
15.	Control card deck with program modifications	35
A-1.	GSE input format sheet	40
A-2.	Example of GSE filled-in data sheet	41
C-1.	Sample Run Card No. 1	102
C-2.	Sample Run Card No. 2	103

TECHNICAL MEMORANDUM X-64862

GSE DATA MANAGEMENT SYSTEM PROGRAMMERS'/USERS' MANUAL

SECTION I. INTRODUCTION

The Ground Support Equipment (GSE) Data Management System (DMS) is a computer program that was developed to provide a central storage source for key data associated with mechanical GSE. The system is fast, accurate, and is easy to use by noncomputer-oriented individuals. This document describes the program and how to use it.

The GSE DMS takes large amounts of GSE data and sorts it alphanumerically (by letter and number) into various categories. For example, a typical user of the program might want a presentable tabular listing of all GSE in the Skylab project associated with a particular vehicle module, such as the Orbital Workshop (OWS). Corresponding GSE attributes [End Item (EI) numbers, descriptions, quantities, fluid media, etc.] are displayed as a major part of this listing. There is a total of eight different types of computer printouts currently available to the user. Seven of these consist of GSE data that have been sorted alphanumerically under a particular catagory. The previous example, Skylab OWS, would be found under what is known as a Stage Sort. Other sort categories include groupings by fluid media, functional code, class code, applicable document, and use location. The other printout available is the master summary table. It is basically an executive inventory list of all the GSE data that are currently in the data bank. Section II, Program Operations Instructions, provides the procedures for data card input format and defines the various program options. Section III illustrates the various example outputs and describes the error and user messages that are included in the GSE DMS. Section IV, Program Aspects, is intended for those individuals who wish to know more about the computer program itself and it has a general explanation of the internal operations of the computer program. Also included in this section is a procedure on how to make temporary or permanent changes to the program.

SECTION II. PROGRAM OPERATION INSTRUCTIONS

The Ground Support Equipment (GSE) Data Management System (DMS) is designed for execution by the Marshall Space Flight Center (MSFC) Univac 1108 Exec VIII Computer System. The GSE Data Management System consists of Fortran programming and a current GSE data bank. For convenience of the user, the complete system has been stored on magnetic tape to permit accessability from remote-site operation.

The GSE Data Management System is capable of performing eight different major types of sorts using the GSE data bank. The GSE data bank may have data added, updated, and/or deleted as required by the user. If these changes are to be permanent, the user can have a new tape made containing the revised GSE data bank.

Control Card Description

Figures 1 and 2 show the control card deck setup required for execution of the GSE Data Management System. The deck indicates the location for GSE data cards when applicable and the location of the instruction data cards for sort requests. It is imperative that the control cards (cards with the "@" sign punched in column 1) be punched correctly and placed in the sequence as shown in Figures 1 and 2.

Run card no. 1, first control card, is the accounting and identification card for the computer run. Certain user information must be supplied on this card and these parameters will be explained in the remainder of this section. SAMPLE is the run I. D. code and may be any six letter combination assigned by the user and AAAAAA is a six digit number which is the accounting job number assigned to the user. NAMEXX is the programmer's name in six characters, XYZ returns the computer run to the programmer's BIN number, 3 is the maximum central processing unit (CPU) run time expected, and 200 represents the maximum number of pages of output expected. Run card no. 2, a special preprinted green card, is a duplicate of the first card. Appendix C gives the details of completing this control card. A four or five digit number, represented by BBBBB on the third control card, is the current reel number of the magnetic tape which contains the GSE Data Management System. The current tape number can be obtained from the publishing organization. The remaining control cards should be exactly as shown.

```
00000000011111111112222222223333333334444444445555555556666666667777777776
12345678901234567890123456789012345678901234567890123456789012345678901234560
@RUN,//P SAMPLE,AAAAAA,NAMEXXBINXYZ,3,200
@RUN,//P SAMPLE,AAAAAA,NAMEXXBINXYZ,3,200
@ASG,T GSE-OLD-TAPE.,T,88888
@REWIND GSE-OLD-TAPE.
@ASG,T 1,F2
@ASG,T 2,F2
@ASG,T 3,F2
@ASG,T 4,F2
@ASG,T CONTROL-FILE,F/1/TRK/10
@COPIN GSE-OLD-TAPE.,TPF$.
@COPY,G GSE-OLD-TAPE.,CONTROL-FILE.
@COPY,G GSE-OLD-TAPE.,1.
@DATA,IL 2.

        GSE DATA CARDS HERE

@END
@XQT PROVE

        INSTRUCTION DATA CARDS HERE

@FIN
```

Figure 1. Control card deck setup.

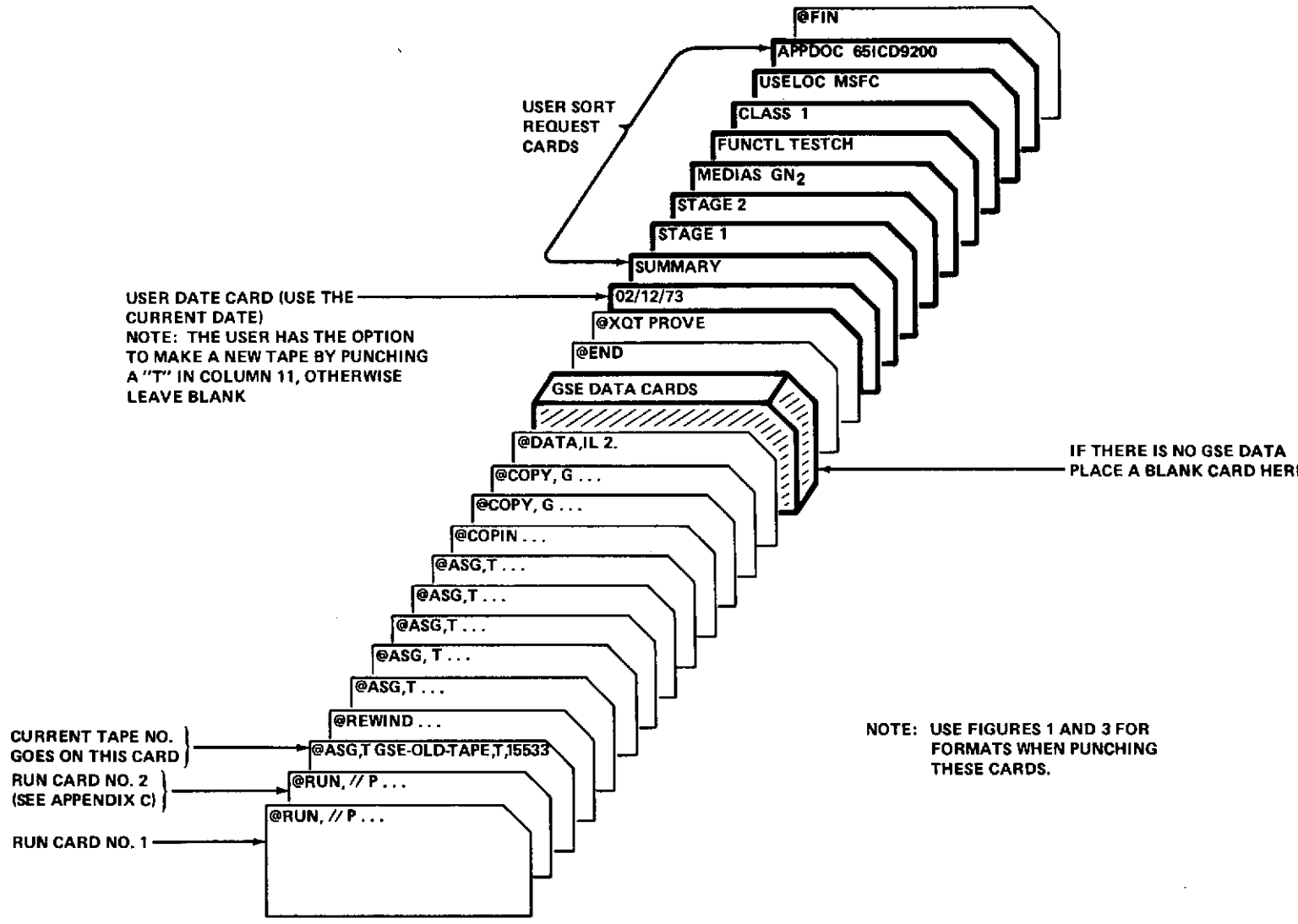

Figure 2. Pictorial control card deck setup.

GSE Data Card Input

The GSE data bank may have data added, updated, and/or deleted. If no GSE data are to be added, updated, and/or deleted from the data bank, one blank computer card must be placed between the @DATA,IL 2 and @END control cards. To add a new item of GSE to the data bank or to update an existing item of GSE in the data bank, the set of cards for each item must follow the card formats described in Appendix A. To delete a piece of GSE from the data bank, one card must be prepared. In column 1 of a computer card the number 1 must be punched and beginning in column 7 the EI number, exactly as it was originally entered into the data bank, must be punched. The remainder of the computer card must be left blank. It is suggested that any deletion cards be placed before any new and/or updated GSE data cards. Figure 3 shows a control card deck with a deletion and an addition to the GSE data bank; however, any single option or combination of the three options is allowable.

Program Instruction Data Input

The program instruction data consists of two types of data cards. Type #1 contains the date of the computer run and an option of whether or not a new GSE Data Management System tape is to be created. Type #2 is the code mnemonic for the sort to be performed.

TYPE #1 INSTRUCTION DATA CARD

In columns 1 through 8 inclusive, the user must punch the date that is desired to be printed on the output tables. The date should follow the format shown in Figure 3. This date will also be recorded with any new GSE data that have been added to the master data file. If a new GSE Data Management System tape is desired because of a GSE data revision or other change, punch a T in column 11 of this same card. If no tape is desired, punch only the date and leave the remainder of the card blank. Only one Type #1 instruction data card is allowed in a control card deck. Figure 3 shows an example of Type #1 instruction card with no tape being created.

TYPE #2 INSTRUCTION DATA CARD(S)

Each Type #2 data card specifies a sort selected for output. One Type #2 data instruction card is required for each sort, but the user may place as many as eight of these cards in each control card deck. Special codes punched

```
0000000001111111111222222222233333333334444444444555555555566666666677777777788
1234567890123456789012345678901234567890123456789012345678901234567890123456789
@RUN,//P SAMPLE,AAAAAA,NAMEXXBINXYZ,3,200
@RUN,//P SAMPLE,AAAAAA,NAMEXXBINXYZ,3,200
@ASG,T GSE-OLD-TAPE.,T,BBBBB
@REWIND GSE-OLD-TAPE.
@ASG,T 1,F2
@ASG,T 2,F2
@ASG,T 3,F2
@ASG,T 4,F2
@ASG,T CONTROL-FILE,F/1/TRK/10
@COPIN GSE-OLD-TAPE.,IPF$.
@COPY,G GSE-OLD-TAPE.,CONTROL-FILE.
@COPY,G GSE-OLD-TAPE.,1.
@DATA,IL 2.
1     DSV-4B-1874
1  1  DSV-4B-433      S-IVB PNEUMATIC CONSOLE--THIS
$                     MODEL IS SIMILAR TO THE DSV-4B-433A
$                     EXCEPT FOR IDENTIFICATION AND
$                     CHANGES IN ELECTRICAL CONNECTORS.
$                     THIS MODEL HAS SATURN IB EFFECTIVITY
2  ASIN-SOG      1  6   1900    S-IVB
3        KSC,SDF          PNEUMA  86     35     72        250K 68          HE,N2
4  651CD9792    1B62632       40M05832      40M11695      651CD9200     651CD9202
*  MAN-021
@END
@XQT PROVE
04/18/73
SUMARY
STAGE1
STAGE2
MEDIAS     GN2
FUNCTL     TESTCH
CLASSC     1
USELOC     MSFC
APPDOC     651CD9200

@FIN
```

Figure 3. Control card deck with GSE data input and all program instruction data types.

on these cards contain the necessary information to perform the desired sort. The sort code mnemonics that the user must use are listed below:

Sort Code Mnemonic	Type of Sort
SUMARY	Master Summary Table
STAGE1	Stage Sort 1
STAGE2	Stage Sort 2
MEDIAS	Fluid Media
FUNCTL	Functional Code
CLASSC	Class Code
USELOC	Use Location
APPDOC	Applicable Document

Explanations of each of the sort types and the type of output are given in Section III.

For the Master Summary Table, Stage Sort 1, or Stage Sort 2 sorts, punch their mnemonic code beginning in column 1 of a computer card and leave the remainder of the card blank. When requesting the Fluid Media, Functional Code, Class Code, or Use Location sorts, punch their mnemonic code beginning in column 1 of a computer card. Since there are multiple classifications within each of these four sorts, a second type of code is also required which specifies detailed information about the sorts. This code is punched beginning in column 11 of the same computer card and is known as the Descriptor Code. The remainder of the card is left blank. The Descriptor Codes for each of these sorts can be found in Appendix A in the discussion of their use in preparing GSE data for the GSE data bank. These codes are used to provide detailed information about the GSE and must be used to relocate this information during these sorts. For the Applicable Document Sort, punch the mnemonic code beginning in column 1 of a computer card, and beginning in column 11 punch the document number to be searched for during the sort. Figure 3 shows an example of each of the eight possible sorts.

For Fluid Media, Functional Code, Class Code, Use Location, and Applicable Document sorts, multiple sorts of each can be requested. For example, Four Fluid Media sorts could be performed using one control card deck by punching a different Descriptor Code on each of four MEDIAS sort request cards. However, a maximum of eight sort requests are allowed. Figure 4 shows an example control card deck with four Fluid Media and two Applicable Document sorts.

```
00000000011111111112222222223333333333444444444555555555566666666677777777778
12345678901234567890123456789012345678901234567890123456789012345678901234567890
@RUN,//P SAMPLE,AAAAAA,NAMEXXBINXYZ,3,200
@RUN,//P SAMPLE,AAAAAA,NAMEXXBINXYZ,3,200
@ASG,T GSE-OLD-TAPE.,T,BBBBB
@REWIND GSE-OLD-TAPE.
@ASG,T 1,F2
@ASG,T 2,F2
@ASG,T 3,F2

@ASG,T 4,F2
@ASG,T CONTROL-FILE,F/1/TRK/10
@COPIN GSE-OLD-TAPE.,TPF$.
@COPY,G GSE-OLD-TAPE.,CONTROL-FILE.
@COPY,G GSE-OLD-TAPE.,1.
@DATA,IL 2.

@END
@XQT PROVE
04/18/73
MEDIAS     GN2
MEDIAS     GO2
MEDIAS     HE
MEDIAS     LH2
APPDOC     MAN-018
APPDOC     65ICD9200
@FIN
```

Figure 4. Control card deck with multiple sorts of the same mnemonic code.

SECTION III. SAMPLE OUTPUT

Eight different types of sorts are available for selection to generate the outputs from the GSE Data Management System. The user can request sorts by stage groups, fluid media, functional operation, ground operation class, use location, and applicable document. A summary of the available records in the master data bank can also be outputed. The total system is a user oriented retrieval method that can display various types of GSE characteristics on different output formats.

The GSE Data Management System provides error messages and user messages that can be encountered as part of the program's output. These messages are associated with the GSE attribute data, sort requests, and tape option.

Description of Sort Types

The Master Summary Table provides the user with a complete listing of all records contained in the data bank. The table contains the EI number, program title, stage name, and reference date. Figure 5 is an example page of the Master Summary Table output. This table permits one to determine when a particular EI was last updated or changed.

The Stage Sort 1 and Stage Sort 2 outputs are listed by program type and stage, module, or system. The Stage Sort 1 output lists for each EI number the description, class code, functional code, use location, and associated applicable documents. The Stage Sort 2 output lists for each EI number the description, fluid media, weight, length, width, height, quantity, unit cost in purchase year, and design responsibility. Figure 6 is an example page of the Stage Sort One Table output, and Figure 7 is an example page of the Stage Sort Two Table output. A complete listing of all entities contained in the data bank is obtained when Stage Sort One and Two is requested.

The Fluid Media sort provides an output table which contains a complete list of all GSE containing the specified type of fluid media. The Media Sort Table output for the requested fluid media contains for each item of GSE the EI number, description, function code, wieght, length, width, height, quantity, unit cost in purchase year, and design responsibility. Figure 8 shows an example page from a Media Sort Table output. The functional code sort provides an output that contains a complete list of GSE that services a particular operational requirement. When a specific operational requirement

MASTER SUMMARY TABLE

ENTITY #	EI NUMBER	PROGRAM	STAGE	REFERENCE DATE
1	CD106R0001A A	SHUTTLE	SHUTTLE MAIN ENGN.	02/01/73
2	CD106R0002A A	SHUTTLE	SHUTTLE MAIN ENGN.	02/01/73
3	CD106R0003A A	SHUTTLE	SHUTTLE MAIN ENGN.	02/01/73
4	CD187R0001A A	SHUTTLE	SHUTTLE MAIN ENGN.	02/01/73
5	CD187R0002A A	SHUTTLE	SHUTTLE MAIN ENGN.	02/01/73
6	CD183R0001A A	SHUTTLE	SHUTTLE MAIN ENGN.	02/01/73
7	CD209R0001A A	SHUTTLE	SHUTTLE MAIN ENGN.	02/01/73
8	CD209R0002A A	SHUTTLE	SHUTTLE MAIN ENGN.	02/01/73
9	CD267R0001A A	SHUTTLE	SHUTTLE MAIN ENGN.	02/01/73
10	CD267R0002A A	SHUTTLE	SHUTTLE MAIN ENGN.	02/01/73
11	CD394R0001A A	SHUTTLE	SHUTTLE MAIN ENGN.	02/01/73
12	CD394R0002A A	SHUTTLE	SHUTTLE MAIN ENGN.	02/01/73
13	CD394R0003A A	SHUTTLE	SHUTTLE MAIN ENGN.	02/01/73
14	CD394R0004A A	SHUTTLE	SHUTTLE MAIN ENGN.	02/01/73
15	CD395R0002A A	SHUTTLE	SHUTTLE MAIN ENGN.	02/01/73
16	CD395R0003A A	SHUTTLE	SHUTTLE MAIN ENGN.	02/01/73
17	CD396R0002A A	SHUTTLE	SHUTTLE MAIN ENGN.	02/01/73
18	CD396R0003A A	SHUTTLE	SHUTTLE MAIN ENGN.	02/01/73
19	CD396R0004A A	SHUTTLE	SHUTTLE MAIN ENGN.	02/01/73
20	CD409R0001A A	SHUTTLE	SHUTTLE MAIN ENGN.	02/01/73
21	CD409R0002A A	SHUTTLE	SHUTTLE MAIN ENGN.	02/01/73
22	CP02N010819	SATURN	S-IC STAGE	01/30/73
23	DC90M04563	SHUTTLE	EXTERNAL TANK	01/25/73
24	DC90M04564	SHUTTLE	EXTERNAL TANK	01/25/73
25	DC90M04565	SHUTTLE	EXTERNAL TANK	01/25/73
26	DC90M04567	SHUTTLE	EXTERNAL TANK	01/25/73
27	DC90M04568	SHUTTLE	EXTERNAL TANK	01/25/73
28	DSV-4B-1874	SATURN	S-IVB STAGE	8/25/72
29	DSV-4B-1875	SATURN	S-IVB STAGE	8/25/72
30	DSV-4B-286	SATURN	S-IVB STAGE	8/25/72
31	DSV-4B-286A	SATURN	S-IVB STAGE	8/25/72
32	DSV-4B-432	SATURN	S-IVB STAGE	8/25/72
33	DSV-4B-432A	SATURN	S-IVB STAGE	8/25/72
34	DSV-4B-433	SATURN	S-IVB STAGE	8/25/72
35	DSV-4B-433A	SATURN	S-IVB STAGE	8/25/72
36	DSV-4B-436	SATURN	S-IVB STAGE	8/25/72
37	DSV-4B-436A	SATURN	S-IVB STAGE	8/25/72
38	DSV-4B-472	SATURN	S-IVB STAGE	8/25/72
39	DSV-4B-473	SATURN	S-IVB STAGE	8/25/72
40	DSV-4B-477	SATURN	S-IVB STAGE	8/25/72
41	DSV-4B-478	SATURN	S-IVB STAGE	01/30/73
42	DSV-4B-479	SATURN	S-IVB STAGE	8/25/72
43	DSV-4B-493	SATURN	S-IVB STAGE	8/25/72
44	DSV-4B-493A	SATURN	S-IVB STAGE	8/25/72
45	DSV-4-187	SATURN	S-IVB STAGE	8/25/72

Figure 5. Example page of Master Summary Table.

is specified, the output table contains the EI number, description, fluid media, weight, length, width, height, quantity, unit cost in purchase year, and design responsibility for each item of GSE that performs the function. Figure 9 shows an example page from a Functional Sort Table output.

STAGE SORT ONE TABLE

04/18/73 SATURN PROJECT PAGE 17

S-II STAGE

EI NUMBER	DESCRIPTION	CLASS CODE	FUNCTIONAL CODE	USE LOCATION	APPLICABLE DOCUMENTS
G7-854600-41B	S7-41B PNEUMATIC CONSOLE--ENCLOSED STATIONARY,PURGED UNIT CONTAINING AMBIENT TEMPERATURE PNEUMATIC SYSTEM SUPPLIED FROM THE A CONSOLE. THE B CONSOLE PROVIDES REMOTE / COMPUTER CONTROLLED PRESSURIZATION OF THE S-II STAGE SYSTEMS.	1	PNEUMA	KSC,SDF,MTF	CEI181M002A 651CD9791 G7-984731 MSFC-MAN-058 G7-985141 40M11703 40M11725 651CD9200 651CD9202
G7-854600-41C	S7-41C PNEUMATIC CONSOLE--ENCLOSED STATIONARY,PURGED UNIT CONTAINING AMBIENT TEMPERATURE PNEUMATIC SYSTEM SUPPLIED FROM THE A CONSOLE. THE C CONSOLE PROVIDES S-II STAGE PURGING AND CHECKOUT FUNCTIONS AND CONTROL PRESSURE FOR STAGE COMPONENT ACTUATION.	1	PNEUMA	KSC,SDF,MTF	CEI181M002A 651CD9791 G7-984731 MSFC-MAN-058 G7-985143 40M11703 40M11725 651CD9200 651CD9202
G7-854600-41D	S7-41D PNEUMATIC CONSOLE--OPEN-SYSTEMS SUPPLIED FROM THE A CONSOLE AND FACILITY. THE D CONSOLE FILTERS, REGULATES,AND DISTRIBUTES GASES, THROUGH THE A7-71 HEAT EXCHANGER, TO THE S-II STAGE ENGINE START BOTTLES.	1	PNEUMA	KSC,SDF,MTF	CEI181M002A 651CD9791 G7-984731 MSFC-MAN-058 G7-985141 40M11703 40M11725 651CD9200 651CD9202
G7-855071	A7-71 HEAT EXCHANGER--STATIONARY LH2 BATH TYPE S.S. DEWAR VESSEL, CONTAINS 3 CHILLING COIL CIRCUITS. CHILLED GASES USED FOR J-2 ENGINE H2 START BOTTLE AND STAGE HE BOTTLE CONDITIONING. A7-71 USED WITH S7-41 A AND D CONSOLES.	1	PNEUMA	KSC,MTF	CP362M00001A 651CD9791 G7-984071 D-611864 SM-S-2-18 40M11703 40M11725 651CD9200 651CD9202

N/A OR NA - NOT APPLICABLE

CLASS CODES: 1.STAGE SYSTEMS 2.PAYLOAD MODULE 3.EXPERIMENT

Figure 6. Example of Stage Sort One Table.

STAGE SORT TWO TABLE

04/16/73 SHUTTLE PROJECT PAGE 44

SHUTTLE MAIN ENGN.

EI NUMBER	DESCRIPTION	MEDIA	WT	ENVELOPE DIM L	W	H	QTY	UNIT COST/YR	DESIGN RESPONSIBILITY
CD106R0001A A	FLOW TESTER, PNEUMATIC - ATMOSPHERIC FOR SPACE SHUTTLE MAIN ENGINE (SSME) NOTE: LAST LETTER IN EI NUMBER INDICATES THE EFFECTIVE PERIOD OF CONTRACT (A IS FOR GSE USED IN ENGINE DEVELOPEMENT B IS FOR GSE USED IN FLIGHT OPERATIONS)	GN2,HE	TBD LB TBD KG	TBD IN TBD CM	TBD IN TBD CM	TBD IN TBD CM	6	TBD	ROCKETDYNE
CD106R0002A A	FLOW TESTER, PNEUMATIC - HIGH PRESS. FOR SPACE SHUTTLE MAIN ENGINE (SSME) NOTE: LAST LETTER IN EI NUMBER INDICATES THE EFFECTIVE PERIOD OF CONTRACT (A IS FOR GSE USED IN ENGINE DEVELOPEMENT B IS FOR GSE USED IN FLIGHT OPERATIONS)	HE	TBD LB TBD KG	TBD IN TBD CM	TBD IN TBD CM	TBD IN TBD CM	6	TBD	ROCKETDYNE
CD106R0003A A	LEAK DETECTOR, HELIUM MASS SPEC. FOR SPACE SHUTTLE MAIN ENGINE (SSME) NOTE: LAST LETTER IN EI NUMBER INDICATES THE EFFECTIVE PERIOD OF CONTRACT (A IS FOR GSE USED IN ENGINE DEVELOPEMENT B IS FOR GSE USED IN FLIGHT OPERATIONS)	HE	TBD LB TBD KG	TBD IN TBD CM	TBD IN TBD CM	TBD IN TBD CM	2	TBD	ROCKETDYNE
CD182R0001A A	TOOL SET,SPECIAL FOR SPACE SHUTTLE MAIN ENGINE NOTE: LAST LETTER IN EI NUMBER INDICATES THE EFFECTIVE PERIOD OF CONTRACT (A IS FOR GSE USED IN ENGINE DEVELOPEMENT B IS FOR GSE USED IN FLIGHT OPERATIONS)	N/A	TBD LB TBD KG	TBD IN TBD CM	TBD IN TBD CM	TBD IN TBD CM	4	TBD	ROCKETDYNE

N/A OR NA = NOT APPLICABLE

Figure 7. Example of Stage Sort Two Table.

MEDIA SORT TABLE

04/18/73 ALL PROGRAMS PAGE 1

HELIUM

EI NUMBER	DESCRIPTION	FUNCTIONAL CODE	WT	ENVELOPE DIM L	W	H	QTY	UNIT COST/YR	DESIGN RESPONSIBILITY
CD106R0001A A	FLOW TESTER, PNEUMATIC - ATMOSPHERIC FOR SPACE SHUTTLE MAIN ENGINE (SSME) NOTE: LAST LETTER IN EI NUMBER INDICATES THE EFFECTIVE PERIOD OF CONTRACT (A IS FOR GSE USED IN ENGINE DEVELOPEMENT B IS FOR GSE USED IN FLIGHT OPERATIONS)	TESTCH	TBD LB TBD KG	TBD IN TBD CM	TBD IN TBD CM	TBD IN TBD CM	6	TBD	ROCKETDYNE
CD106R0002A A	FLOW TESTER, PNEUMATIC - HIGH PRESS. FOR SPACE SHUTTLE MAIN ENGINE (SSME) NOTE: LAST LETTER IN EI NUMBER INDICATES THE EFFECTIVE PERIOD OF CONTRACT (A IS FOR GSE USED IN ENGINE DEVELOPEMENT B IS FOR GSE USED IN FLIGHT OPERATIONS)	TESTCH	TBD LB TBD KG	TBD IN TBD CM	TBD IN TBD CM	TBD IN TBD CM	6	TBD	ROCKETDYNE
CD106R0003A A	LEAK DETECTOR, HELIUM MASS SPEC. FOR SPACE SHUTTLE MAIN ENGINE (SSME) NOTE: LAST LETTER IN EI NUMBER INDICATES THE EFFECTIVE PERIOD OF CONTRACT (A IS FOR GSE USED IN ENGINE DEVELOPEMENT B IS FOR GSE USED IN FLIGHT OPERATIONS)	TESTCH	TBD LB TBD KG	TBD IN TBD CM	TBD IN TBD CM	TBD IN TBD CM	2	TBD	ROCKETDYNE
CD209R0001A A	CONSOLE, ROCKET ENGINE ELECTRICAL PNEUMATIC-CHECKOUT FOR SPACE SHUTTLE MAIN ENGINE (SSME) NOTE: LAST LETTER IN EI NUMBER INDICATES THE EFFECTIVE PERIOD OF CONTRACT (A IS FOR GSE USED IN ENGINE DEVELOPEMENT B IS FOR GSE USED IN FLIGHT OPERATIONS)	TESTCH	TBD LB TBD KG	TBD IN TBD CM	TBD IN TBD CM	TBD IN TBD CM	2	TBD	ROCKETDYNE

N/A OR NA = NOT APPLICABLE

Figure 8. Example of Media Sort Table.

FUNCTIONAL SORT TABLE

04/18/73 ALL PROGRAMS PAGE 3

TEST-CHECKOUT EQUIPMENT

EI NUMBER	DESCRIPTION	MEDIA	AT	ENVELOPE DIM L	W	H	QTY	UNIT COST/YR	DESIGN RESPONSIBILITY
CD267R0001A A	TEST SET, SINGLE ROCKET ENGINE FOR SPACE SHUTTLE MAIN ENGINE (SSME) NOTE: LAST LETTER IN EI NUMBER INDICATES THE EFFECTIVE PERIOD OF CONTRACT (A IS FOR GSE USED IN ENGINE DEVELOPEMENT B IS FOR GSE USED IN FLIGHT OPERATIONS)	N/A	TBD	LB TBD IN TBD IN TBD IN TBD KG TBD CM TBD CM TBD CM			2	TBD	ROCKETDYNE
CD267R0002A A	TEST SET, INSTALLED ROCKET ENGINE FOR SPACE SHUTTLE MAIN ENGINE (SSME) NOTE: LAST LETTER IN EI NUMBER INDICATES THE EFFECTIVE PERIOD OF CONTRACT (A IS FOR GSE USED IN ENGINE DEVELOPEMENT B IS FOR GSE USED IN FLIGHT OPERATIONS)	N/A	TBD	LB TBD IN TBD IN TBD IN TBD KG TBD CM TBD CM TBD CM			4	TBD	ROCKETDYNE
CD395R0002A A	COVERS - PROTECTIVE, ENGINE THRUST CHAMBER FOR SPACE SHUTTLE MAIN ENGINE (SSME) NOTE: LAST LETTER IN EI NUMBER INDICATES THE EFFECTIVE PERIOD OF CONTRACT (A IS FOR GSE USED IN ENGINE DEVELOPEMENT B IS FOR GSE USED IN FLIGHT OPERATIONS)	N/A	TBD	LB TBD IN TBD IN TBD IN TBD KG TBD CM TBD CM TBD CM			11	TBD	ROCKETDYNE
CD395R0003A A	PAD, PROTECTIVE-THRUST CHAMBER INTERIOR FOR SPACE SHUTTLE MAIN ENGINE (SSME) NOTE: LAST LETTER IN EI NUMBER INDICATES THE EFFECTIVE PERIOD OF CONTRACT (A IS FOR GSE USED IN ENGINE DEVELOPEMENT B IS FOR GSE USED IN FLIGHT OPERATIONS)	N/A	TBD	LB TBD IN TBD IN TBD IN TBD KG TBD CM TBD CM TBD CM			4	TBD	ROCKETDYNE

N/A OR NA - NOT APPLICABLE

Figure 9. Example of Functional Sort Table.

The class code sort option provides an output which contains a complete list of the GSE that supports the major hardware category specified. The EI number, description, function code, stage, fluid media, quantity, and design responsibility for each item of GSE associated with the requested operation is contained in the output table. Figure 10 shows an example page from a Class Code Sort Table output. The use location sort option generates a complete list of all GSE that is used at a specific geographical location or facility. The output contains the EI number, description, function code, weight, length, width, height, fluid media, and design responsibility for each item of GSE used in the specified location. Figure 11 shows an example page from a Use Location Sort Table output. The applicable document sort provides an output which contains all GSE that is referenced in the requested document. The output contains the EI number, description, class code, function code, and associated applicable documents. Figure 12 shows an example page from an Applicable Document Sort Table output.

User Messages

The GSE Data Management System can have data added, updated, and/or deleted from the data bank. When no GSE attribute data are to be changed, the message shown in Example 1 is printed showing that no changes occurred. The total number of GSE items in the data bank is also shown.

```
***GSE ATTRIBUTE DATA FILE HAS BEEN UPDATED***
    0 NEW ENTITIES
    0 UPDATED ENTITIES
    0 DELETED ENTITIES
  171 TOTAL ENTITIES USED DURING THIS RUN
```

Example 1

If any changes to the GSE attribute data have been included in the control card deck, the number of each specific type of change will be printed out along with the new total number of GSE items now in the data bank. Example 2 shows a total of five changes to the GSE data bank and the new total.

CLASS CODE SORT TABLE

12/26/72 ALL PROGRAMS PAGE 1

GSE TEST SUPPORT

EI NUMBER	DESCRIPTION	FUNCTION CODE	STAGE	MEDIA	QTY	DESIGN RESPONSIBILITY
CP02N01C819	S-IC PNEUMATIC PORTABLE TESTERS--SIX PORTABLE SUITCASES CONTAINING SWITCHES,INDICATOR LIGHTS,AND METERS USED IN CHECKING CONTINUITY OF ELECTRICAL COMPONENTS OF THE S-IC PNEUMATIC CONSOLE AND THE S-IC PNEUMATIC CHECKOUT RACKS.	TESTCH	S-IC	N/A	16	GEHUNTSVILLE
05V-48-478	POWER CABLE REEL CART--THIS MOBILE UNIT CONSISTS OF STORAGE REEL,CRANK, FRAME,SEMI-PNEUMATIC WHEELS,TWO WATER HOSES,ELECTRICAL POWER CABLE, AND QUICK DISCONNECTS. THE UNIT IS USED TO SUPPLY COOLING WATER AND ELECTRICAL POWER TO THE 05V-48-479.	SERVIC	S-IVB	N/A	8	ASTN-SOG
22A12006	TESTER ASSY, HYDRAULIC PERFORMANCE--A MOBILE UNIT CONTAINING SWITCHES,INDICATOR LIGHTS, CABLES,AND A MANIFOLD ASSEMBLY. THE TESTER SIMULATES THE STEP FUNCTION FLOW REQUIREMENTS OF THE S-IC F-1 ENGINES START SEQUENCE TO CHECKOUT THE S-IC HYDRAULIC SUPPLY AND CHECKOUT UNIT.	TESTCH	S-IC	N/A	01	GEHUNTSVILLE
65B36693	TESTER ASSY,FORWARD UMBILICAL SERVICER CONSOLE--A PORTABLE SUITCASE CONTAINING SWITCHES,INDICATOR LIGHTS,METERS,AND CABLES TO TEST THE S-IC FORWARD UMBILICAL SERVICE CONSOLE.	TESTCH	S-IC	N/A	03	GEHUNTSVILLE

Figure 10. Example of Class Code Sort Table.

USE LOCATION SORT TABLE

04/18/73 ALL PROGRAMS PAGE 2

MARSHALL CENTER

ET NUMBER	DESCRIPTION	FUNCTIONAL CODE	WT	ENVELOPE DIM L	W	H	MEDIA	DESIGN RESPONSIBILITY
DC9DM04567	EXTERNAL TANK(ET)LH2 TEST TANK STORAGE CRADLE-USED TO SUPPORT THE LH2 TEST TANK IN THE HORIZONTAL POSITION DURING PERIODS OF LOW ACTIVITY OR STORAGE-CONSTRUCTED OF COMMERCIAL GRADE STEEL SECTIONS,PLATES,AND HARDWARE,NO ROTATIONAL CAPABILITY,CAN BE DISASSEMBLED FOR SHIPMENT	TRANSP	TBD LB TBD KG	TBD IN TBD CM	TBD IN TBD CM	TBD IN TBD CM	N/A	S&E-ASTN-ETT
DC9DM04568	EXTERNAL TANK(ET)LOX TEST TANK STORAGE CRADLE-USED TO SUPPORT THE LOX TEST TANK IN THE HORIZONTAL POSITION DURING PERIODS OF LOW ACTIVITY OR STORAGE-CONSTRUCTED OF COMMERCIAL GRADE STEEL SECTIONS,PLATES,AND HARDWARE,NO ROTATIONAL CAPABILITY,CAN BE DISASSEMBLED FOR SHIPMENT	TRANSP	TBD LB TBD KG	TBD IN TBD CM	TBD IN TBD CM	TBD IN TBD CM	N/A	S&E-ASTN-ETT
DSV-4B-286	PNEUMATIC CONSOLE PORTABLE TEST--THIS MODEL SIMILAR TO DSV-4B-286A EXCEPT FOR IDENTIFICATION AND CHANGES IN ELECTRICAL CONNECTORS. THIS MODEL HAS SATURN IB EFFECTIVITY	TESTCH	140 LB 63KG	28 IN 71CM	28 IN 71CM	12 IN 30CM	N/A	MDAC-WD
DSV-4B-286A	PNEUMATIC CONSOLES PORTABLE TEST SET THIS MODEL CONSISTS OF THREE TEST AND MONITOR SETS AND FOUR CABLE CASES, ALL OF THE SUITCASE TYPE. THE TEST AND MONITOR SETS CONTAIN ELECTRICAL CONNECTORS, SWITCHES, CIRCUIT BREAKERS,FUSES,METERS,AND INDICATOR LIGHTS. THE CABLE CASES CONTAIN THE CABLES TO POWER,CONTROL AND MONITOR ALL FUNCTIONS WITHIN THE DSV-4B-432A,DSV-4B-433A AND DSV-4B-438A FOR TEST AND CHECKOUT AND TO PROVIDE LOCAL CONTROL. THE MODEL HAS SATURN V EFFECTIVITY.	TESTCH	140 LB 63KG	28 IN 71CM	28 IN 71CM	12 IN 30CM	N/A	MDAC-WD

N/A OR NA - NOT APPLICABLE

Figure 11. Example of Use Location Sort Table.

APPLICABLE DOCUMENT SORT TABLE

04/18/73 DOCUMENT 65ICD9200 PAGE 8

ALL PROGRAMS

EI NUMBER	DESCRIPTION	CLASS CODE	FUNCTIONAL CODE	APPLICABLE DOCUMENTS
650804000	HYDRAULIC POWER UNIT--STATIONARY SUB-UNIT OF S-IC HYDRAULIC SUPPLY AND CHECKOUT UNIT. CONTAINS PUMPS, MOTORS, VALVES, GAGES, FILTERS, AND AN 800 GALLON RESERVIOR FOR THE PURPOSE OF STORING, CONDITIONING, AND DELIVERING RJ-1 FLUID UNDER PRESSURE TO THE FIVE F-1 ENGINES OF THE S-IC STAGE DURING ALL GROUND PHASES OF ENGINE AND FLIGHT -ACTUATOR- CHECKOUT.	1	SERVIC	65ICD9790 65ICD9200 650803000 65ICD9202 40M05834 MSFC-MAN-006
8800-0000-0	GROUND SUPPORT COOLING UNIT-- ENCLOSED, PURGED, STATIONARY UNIT CONTAINING A COOLANT RESERVOIR, COOLANT CIRCULATION SUBSYSTEM, REFRIGERANT SUBSYSTEM, PUMPS, MOTORS, COMPRESSOR, NUMEROUS GAGES, VALVES, METERS, AND CONTROLS. IT IS USED TO CONDITION COOLANT MEDIA FOR IU AND S-IVB OR BOARD COLD PLATES COOLING AND HEAT DISAPATION.	1	SERVIC	65ICD9793 MAN-008 8800-001 65ICD9202 65ICD9200 40M11847
674500	INERT PREFILL UNIT--CONTAINS TWO SEPARATE STATIONARY SUB-UNITS, 892996-HYDRAULIC PUMPING UNIT AND 893100-CHECKOUT CONSOLE. USED FOR DELIVERING INERT FLUID TO THE S-IC STAGE F-1 ENGINE JACKETS FOR PRE-IGNITION FILLING.	1	SERVIC	65ICD9790 159155 159152 65ICD9200 65ICD9202 40M00213 MSFC-MAN-007
892996	HYDRAULIC PUMPING UNIT--STATIONARY SUB-UNIT OF INERT PREFILL UNIT. CONTAINS VALVES, PUMP, MOTOR, GAGES, AND A 770 GALLON FLUID RESERVIOR. USED FOR STORING AND PUMPING INERT FLUID.	1	SERVIC	65ICD9790 159155 159152 65ICD9200 65ICD9202 40M00213 MSFC-MAN-007

N/A OR NA = NOT APPLICABLE
CLASS CODES: 1.STAGE SYSTEMS 2.PAYLOAD MODULE 3.EXPERIMENT

Figure 12. Example of Applicable Document Sort Table.

```
***GSE ATTRIBUTE DATA FILE HAS BEEN UPDATED***
   2 NEW ENTITIES
   1 UPDATED ENTITIES
   2 DELETED ENTITIES
 133 TOTAL ENTITIES USED DURING THIS RUN
```

Example 2

At the end of each sort requested in the control card deck, a statement is printed stating to the user that the sort was completed. Example 3 is a simple statement from the completion of a Functional Code sort request. If a valid sort is requested for a GSE category that is not in the data bank, a message is printed stating the master file contained no data for the specified sort. If a sort was requested for the fluid media Freon and no Freon was used with any of the GSE items in the data bank, the message shown in Example 4 would be printed out.

```
*** COMPLETION OF FUNCTIONAL CODE SORT ***
```

Example 3

```
***MASTER FILE CONTAINED NO DATA WITH FREON  FLUID MEDIA SORT***
```

Example 4

When all the sorts associated with the control card deck (maximum of eight) have been performed, a message is printed showing the number of sorts actually performed. Example 5 shows this statement when six sorts were requested.

```
*** THIS RUN COMPLETED  6 SORTS ***
```

Example 5

If the user has changed the GSE data bank, made programming changes, or has other reasons, a new tape can be made. When a new tape is correctly requested using the Type #1 instruction data card, a message is printed out stating a new tape was created. Example 6 shows this statement. Each tape has a number which must be placed on the third control card of a sort request deck. When a new tape is created, a four or five digit number of the tape is found at the end of the computer run which made the tape. Example 7 shows the location of a five digit number which was generated during this run.

```
<<<<< USER MESSAGE... AN OUTPUT TAPE HAS BEEN CREATED ON THIS RUN >>>>>
```

Example 6

```
SORT: T/R=842.5MS, IC=   134, OC=   134, DIA=  1.2
   4- SAVE  BLANK 16/14 TAPE        -1 50720
      SERVICE     17/14  50720
   4 10709
   TIME: 00:01:12.324    IN: 268    OUT: 0        PAGES:   152
   INITIATION  TIME:   21:49:20-JAN 19,1973
   TERMINATION TIME:   22:04:17-JAN 19,1973
```

Example 7

Error Messages

Several types of errors can be made when making changes to the GSE data bank. When adding or updating the data bank, a set of four card types must be prepared. If one of the cards in these sets is placed out of order or punched incorrectly, a statement will be printed giving the number of the card in error. These card numbers are tabulated with an input listing of the data set printed out at the beginning of each run. Example 8 shows the result of submitting five changes to the data bank with a card out of order or punched incorrectly in one set. The correct data sets will be incorporated. The message shown in Example 8 will also be printed when a deletion card is out of

order by being placed within the set of four card types for adding or updating of of the GSE data bank. When punching the project number on card Type #1 or the class code on card Type #2 of the four card types of the GSE attribute data, a character other than a number could possibly be used. If this should occur, the error message shown in Example 8 will be printed. An incorrect dollar sign ($) for a Type #1 continuation card, when adding or updating the GSE data bank, will also result in the error message shown in Example 8.

```
***ERROR WHILE READING INPUT FILE 2***
***CARD #    17 IS OUT OF ORDER OR IS PUNCHED INCORRECTLY***


***GSE ATTRIBUTE DATA FILE HAS BEEN UPDATED***
   1 NEW ENTITIES
   1 UPDATED ENTITIES
   2 DELETED ENTITIES
 132 TOTAL ENTITIES USED DURING THIS RUN
```

Example 8

When adding or updating the GSE data bank, it is allowable to have up to 20 lines of description which consist of one Type #1 card and 19 Type #1 continuation cards (see Appendix A). If more than 20 lines of description are used, the error message shown in Example 9 will be printed out. The data associated with this EI number will not be placed in the data bank, but all correct data sets will be used.

```
***MORE THAN 20 LINES OF DESCRIPTION ASSOCIATED WITH  EI NO. OSV-48-433


                    * EXECUTION CONTINUING***
```

Example 9

The Type #1 instruction data card contains the date of the computer run and an option of creating a new tape. If the date is of an incorrect format, the error in Example 10 will be printed. This error message will also be printed out if the Type #1 instruction data card is not placed in the computer card deck. This error will cause the run to be terminated. If a new tape is to be created, a T is punched in column 11 of the Type #1 instruction data

card. When an incorrect character is used, the error message shown in Example 11 is printed. The requested sorts will be performed, but a new tape will not be made.

```
*** ERROR IN PROGRAM INSTRUCTIONS
          ...THE DATE IS MISSING OR IS PUNCHED INCORRECTLY***
```

Example 10

```
*** ERROR IN TAPE OPTION INSTRUCTION ...AN INCORRECT CHARACTER WAS USED ***

*** NO TAPE CREATED ON THIS RUN ***
```

Example 11

Each of the Type #2 instruction data cards specifies a specific sort to be output. Of the eight possible sort types, the Master Summary Table, Stage Sort 1, and Stage Sort 2 data cards require a code mnemonic only. The Applicable Document Sort may have any document number as its descriptor code. The remaining four sorts require a code mnemonic and a descriptor code. Section II gives the details of preparing these cards. If a mnemonic code is incorrect, error messages stating that the mnemonic code is misspelled and that the computer is continuing to process the remaining sorts is printed out. Example 12 shows a sample of an incorrect mnemonic code.

```
***  MNEMONIC NAME SUMMAR IS MISSPELLED***

*** CONTINUING TO PROCESS OTHER SORTS ***
```

Example 12

The descriptor code of a sort request can also be in error. When a descriptor code is used that is not valid, see Appendix A for correct codes, the error message in Example 13 is outputed. In this example the fluid media LH_3 was requested. The correct fluid should have been LH_2, therefore the error message was printed.

```
***DESCRIPTER NAME:  LH3      IN ERROR***
```

Example 13

When requesting an Applicable Document Sort, a code mnemonic and a document number must be specified. If the document number used is not associated with any GSE items in the data bank, the message in Example 14 will be printed.

```
***ERROR***DOCUMENT 555555555555 COULD NOT BE FOUND***
```

Example 14

SECTION IV. PROGRAM ASPECTS

Program Execution

The GSE Data Management program consists of control cards, a mainline program, subroutines, and procedure tables used for execution by the Univac 1108 system. The following briefly explains each of these items and discusses their function in the program. It will be convenient to refer to Figure 13 showing the general overview of the complete program.

The mainline program takes control over all other routines. The mainline reads in the program instruction data provided by the user on data cards and makes available such information to other routines. The first program instruction data card contains the date of the run and an optional variable that is used to determine whether to generate an output tape with new or updated data. If an invalid date is encountered by the subroutine DATECK an error message will be written and the run terminated. If GSE attribute information is to be added, updated, or deleted from the data base, subroutine UPDATE will be called from the MAINLINE. UPDATE uses four drum files. All information from the current tape is put on FILE 1. If there are any GSE attribute cards in the card deck run-stream, the data from these cards are put on FILE 2. UPDATE then uses FILE 3 as a working file, i.e., all data from FILE 1 and FILE 2, whether it be duplicate or not, are all put on FILE 3. UPDATE then searches FILE 3 for the most current data and puts them on FILE 4. Updated data on FILE 4 are then used for the current run and can be stored on a new user requested output tape.

Up to eight different sets of sort mnemonic codes can appear on individual data cards following the initial data card. The MAINLINE determines if such are valid names and converts then to an integer code. Subroutine MERGE is then called from the MAINLINE and uses this code to determine which sort operation to perform. There are seven different types of sorts and a master summary table available. Each sort thus has a corresponding sort subroutine as contained in Table 1.

Using the Univac Sort/Merge Package feature (see Program Characteristics), each subroutine does the internal sorting logic. Furthermore, each sort subroutine has a peculiar group of routines that handle the printout logic. For example, subroutine PRINT2 handles the pure mechanics of printing the newly sorted GSE data for Stage Sort 2 including use of subroutines DIMMET, WGTMET, and RELALP for calculation of metric units; subroutine

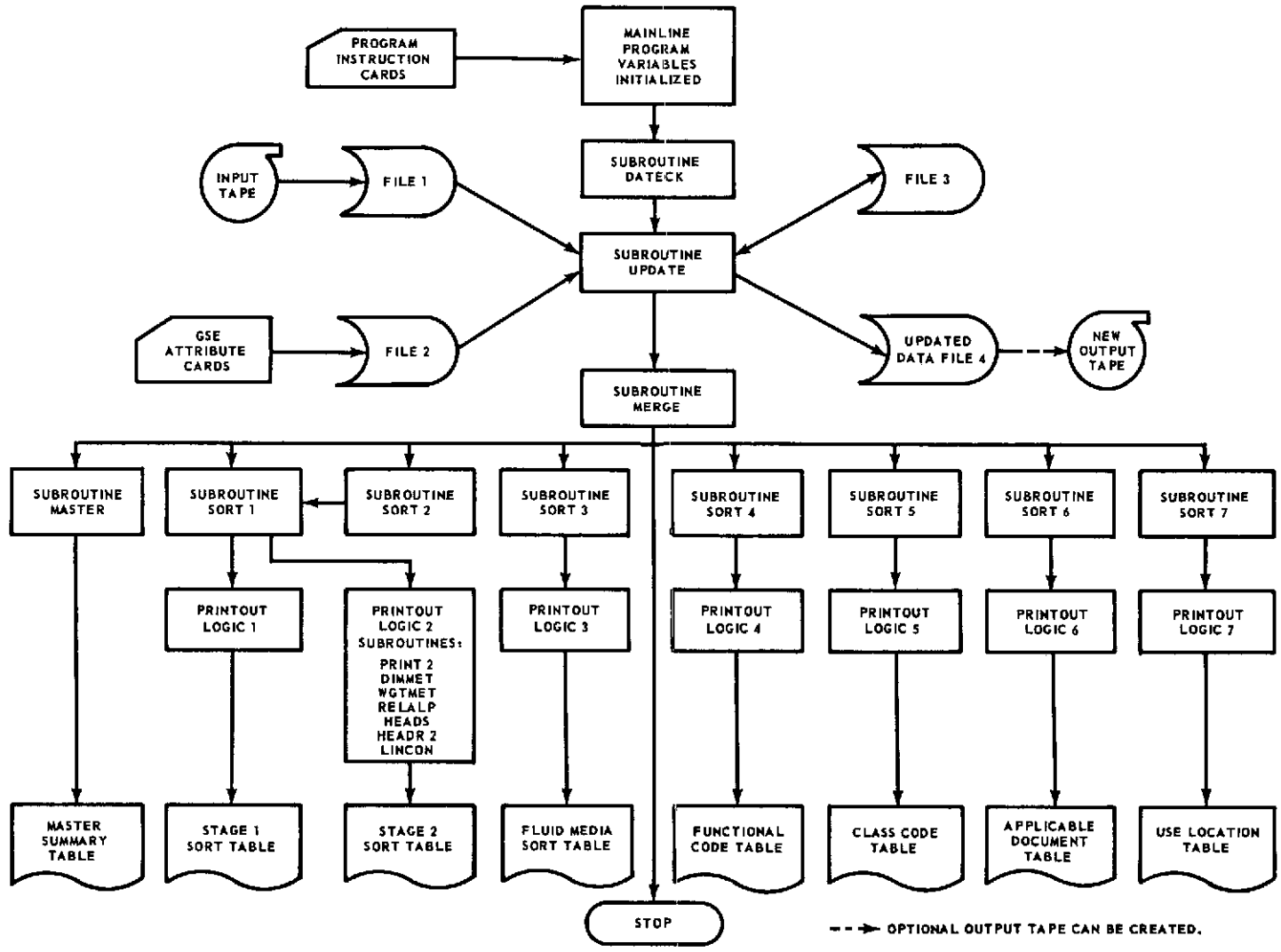

Figure 13. Master program execution flow chart.

TABLE 1. SORT SUBROUTINE NAME FOR TYPE OF SORT

Sort Subroutine Name	Type of Sort
MASTER	Master Summary Table
SORT1	Stage Sort 1
SORT2	Stage Sort 2
SORT3	Fluid Media Sort
SORT4	Functional Code Sort
SORT5	Class Code Sort
SORT6	Applicable Document Sort
SORT7	Use Location Sort

HEADS calls the appropriate heading subroutine for use by the respective print routine; subroutine HEADR2 prints the column headings and footnotes on every page of the sort. Subroutine LINCON determines whether to start writing on a new page or continue writing on the current page. This printout would be in presentable tabular listings displaying all GSE items found under Stage Sort 2. In similar fashion, the sequence of calls to PRINT1, HEADS, HEADR1, and LINCON will accomplish the printout logic for Stage Sort 1.

Program Modification Procedures

Each card of the current version of the GSE Data Management Program is catalogued by line entry number. This simplifies the procedure by which any temporary modifications can be made to the program if it is not entirely satisfactory to the program analyst. When making temporary changes, the line entry of each statement can be obtained from the left-hand column number of the current program listing (see Appendix B).

Permanent changes can also be made to the program. Being modular in form, the program is flexible enough to incorporate major additions with

minimum effort. For example, assume that a type of sorting function is desired that is not currently available from the program. Through the use of several control cards, the program analyst can add those subroutines (subprograms) that were designed to provide such extended capability. These additions would then be represented by several new block operations to be fitted on Figure 13. Refer to the discussion in the section on how to change or add Fortran subroutines.

Changes to Program Procedures Tables

Figure 14 is an example sheet from Appendix B that illustrates a typical catalogued listing from one of the program's procedure tables. Notice that each statement has the catalogued computer number positioned to the extreme left of it. These catalogued listing numbers, as shown in Appendix B, will change if any Fortran routines or procedure tables are altered when permanent changes to the program have been made by the user. To temporarily modify any of these statements, the insert card method is used. This requires an insert card that may contain any of the following information:

1. The location number of the statement to be replaced or modified (see Example 15).

2. The location number of the statement to be deleted entirely (see Example 16).

3. The sequential location numbers of the statements to be modified (see Example 17).

4. The location numbers of the statement which modifications are to follow (see Example 18).

All insert cards are punched beginning with a minus sign in card column one (1) and must follow the procedure table control card in the runstream. Example 15 shows the procedure table control card and insert card setup.

```
00000000011111111112222222223333333333444444444455555555556666666667777777778
12345678901234567890123456789012345678901234567890123456789012345678901234567890
$PDP,ULF TABLE,TABLE
-98,98
     *          ,'SRB   ',' SOLID ','ROCKET',' BST, '
-100,100
     *          ,'SPSH  ','   SPA','CE SHU',' TTLE  '
```

Example 15

```
      XSTAGE PROC                                                                TBLE  74
C ***********************************************************************TBLE  75
C *   STAGENAME PROCEDURE TABLE                                             *TBLE  76
C *                                                                         *TBLE  77
C *   THE ARRAY ''GSETLE'' CONTAINS THE VARIOUS GSE STAGENAME TITLES        *TBLE  78
C *                                                                         *TBLE  79
C *              VARIABLE                 DEFINITION                        *TBLE  80
C *              ********                 **********                        *TBLE  81
C *                                                                         *TBLE  82
C *              NSTAGE        THE NUMBER OF STAGENAMES APPEARING IN        *TBLE  83
C *                            THIS TABLE + 1                               *TBLE  84
C *                                                                         *TBLE  85
C *                    ROUTINES THAT USE XSTAGE PROC                        *TBLE  86
C *                    *****************************                        *TBLE  87
C *                                 MASTER                                  *TBLE  88
C *                                 HEADR1                                  *TBLE  89
C *                                 HEADR2                                  *TBLE  90
C *                                 SORT1                                   *TBLE  91
C *                                 PRINT1                                  *TBLE  92
C *                                                                         *TBLE  93
C ***********************************************************************TBLE  94
      INTEGER GSETLE(100)                                                        TBLE  95
      NSTAGE=12                                                                  TBLE  96
      DATA GSETLE/'           ','**UNKN','OWN** ','           '                   TBLE  97
     *           ,'SRB        ','S. R. ','BOOSTE','R          '                   TBLE  98
     *           ,'ET         ','EXTERN','AL TAN','K          '                   TBLE  99
     *           ,'SSME       ','SHUTTL','E MAIN',' ENGN.'                        TBLE 100
     *           ,'AM         ',' AIRLO','CK MOD','ULE        '                   TBLE 101
     *           ,'OWS        ',' ORBIT','AL WOR','KSHOP '                        TBLE 102
     *           ,'ATM        ','APOLLO',' TELE.',' MOUNT'                        TBLE 103
     *           ,'MDA        ','      ',' MDA  ','           '                   TBLE 104
     *           ,'IU         ',' INSTR','UMENT ','UNIT       '                   TBLE 105
     *           ,'S-IC       ',' S-IC ','STAGE ','           '                   TBLE 106
     *           ,'S-IVB      ','S-IVB ','STAGE ','           '                   TBLE 107
     *           ,'S-2        ',' S-II ','STAGE ','           '/                  TBLE 108
C    *           ,THIS SPACE IS FOR THE CONTINUATION OF GSE DESCRIPTIVE TBLE 109
C    *           ,                                                               TBLE 110
C    *           ,                                                               TBLE 111
C    *           ,                                                               TBLE 112
C    *           ,                                                               TBLE 113
C    *           ,                                                               TBLE 114
C    *           ,                                                               TBLE 115
C    *           ,                                                               TBLE 116
C    *           ,THIS SPACE IS FOR THE CONTINUATION OF GSE DESCRIPTIVE TBLE 117
      END                                                                        TBLE 118
```

Figure 14. Procedure table catalogue listing.

The statements that were in some way modified are identified as location Nos. 98 and 100. Note that on each insert card the location number is listed twice and there is a one to one replacement on the statement that was 98. It should also be noted that all statements that are placed between the insert card and the succeeding insert card, if any, will replace the statement at location No. 100. Thus it would be possible to insert a finite number of statements at location No. 100 or, as Example 16 illustrates, delete a statement completely if no statement should follow the insert card. However, due to the logic and nature of the particular procedure table, location No. 96 would have to be modified also, so as to reflect the correct number of items that would remain in that procedure table after the deletion.

```
00000000011111111112222222222333333333344444444445555555555666666666777777777
12345678901234567890123456789012345678901234567890123456789012345678901234567890
$PDP,ULF TABLE,TABLE
-96,96
      NSTAGE=11
-100,103
```

Example 16

Example 17 shows how the user can modify all statements between and including 99 through 102. If no statements appear after the insert card, the program segment identified on that insert card would be deleted entirely.

```
00000000011111111112222222222333333333344444444445555555555666666666777777777
12345678901234567890123456789012345678901234567890123456789012345678901234567890
$PDP,ULF TABLE,TABLE
-99,102
     *           ,'SPSH  ','   SPA','CE SHU','TTLE  '
```

Example 17

Example 18 illustrates how any number of statements can be inserted to follow a certain statement in the program. Note that only one location number is required on the insert card following the minus sign.

```
00000000011111111112222222222333333333344444444445555555555666666666777777777
12345678901234567890123456789012345678901234567890123456789012345678901234567890
$PDP,ULF TABLE,TABLE
-96,96
      NSTAGE=13
-97
     *           ,'ALL   ','USED I','N ALL ','STAGES'
```

Example 18

Due to the nature of the procedure tables and the Univac 1108 system, it is necessary to recompile those Fortran subroutines that use a procedure table that has been modified. Note that within the comment statement portion of each procedure table there is a list of subroutine names that use that procedure table. Recompilation requires that these subroutine names must then appear on control cards that follow the end of all insert cards to the procedure tables. Example 19 illustrates the format for such control cards. It is recommended that the "S" option be used and, as the example shows, the subroutine name should appear twice and should be separated by a comma.

```
00000000011111111112222222222333333333344444444445555555555666666666677777777778
12345678901234567890123456789012345678901234567890123456789012345678901234567890
@PDP,ULF TABLE,TABLE
-98,98
     *           ,'SRB   ','SOLID ','ROCKET',' BST. '
-100,100
     *           ,'SPSH  ',' SPA','CE SHU','TTLE  '
@FOR,S MASTER,MASTER
@FOR,S HEADR1,HEADR1
@FOR,S HEADR2,HEADR2
@FOR,S SORT1,SORT1
@FOR,S PRINT1,PRINT1
```

Example 19

Changing and Adding Fortran Routines

If it were necessary to modify the actual Fortran statements in the program, the same insert card method is illustrated in the previous example is used. Example 20 illustrates the appearance of the control card needed and an example insert card setup. It is required that the "US" option be used on these control cards, noting also that the subroutine name again appears twice and is separated by a comma. A very important reminder is that these Fortran control cards (@FOR,US) and their associated insert cards must follow the procedure table control card and its associated insert cards, if any, in the card deck runstream.

Example 21 shows that only one control card is needed to add a new subroutine to the program. Simply put the control card before the subroutine deck. Note that the "IS" option is used, followed by a blank space and then the name of the routine. Addition of a new subroutine must follow the procedure table changes, if any, that are being made.

```
0000000001111111111222222222233333333334444444444555555555566666666677777777778
1234567890123456789012345678901234567890123456789012345678901234567890123456789
@FOR,US MAIN,MAIN
-59,59
   60 FORMAT(1H1,5X,19H*** MNEMONIC NAME ,A6,17H IS MISSPELLED.***)
-62,62
   67 CDESRT(I)=10
-63,63
@FOR,US MERGE,MERGE
-22,22
      IF(CDESRT(I).LE.0.OR.CDESRT(I).GT.NUMSR1)GO TO 68
-24,24
      GO TO (8,16,24,32,40,48,56,59,74,60),NARGU
@FOR,US UPDATE,UPDATE
-119
      IF(NCARD.EQ.1.AND.NCR.EQ.0.AND.LINUM(2'HEQ*BLANK)GO TO 30
```

Example 20

```
0000000001111111111222222222233333333334444444444555555555566666666677777777778
1234567890123456789012345678901234567890123456789012345678901234567890123456789
@FOR,IS NEW,NEW
      SUBROUTINE NEW                                                    NEW 0000
C*****************************************************************     NEW 0001
C**** THIS IS A NEW FORTRAN SUBROUTINE THAT IS BEING ADDED TO GSEDMS     NEW 0002
C*****************************************************************     NEW 0003
      DIMENSION I(10)                                                   NEW 0005
    1 READ(5,100)LX                                                     NEW 0006
    5 DO 20,J=1,10                                                      NEW 0007
   15 KPRINT=I(J)*LX                                                    NEW 0008
   16 WRITE(6,17)KPRINT,LX                                              NEW 0009
   17 FORMAT(1H1,I2)                                                    NEW 0010
   20 CONTINUE                                                          NEW 0011
  100 FORMAT(1H1,I2)                                                    NEW 0012
      RETURN                                                            NEW 0013
      END                                                               NEW 0014
```

Example 21

Deck Setup, Control Cards, and Options

Following the procedure table and Fortran subroutine control cards and their associated insert cards, a final control card is required to allow for these program modifications. Example 22 shows what is called the @MAP control card. It is used to collect all routines from the current tape and make a new temporary program with the user's specified program modifications.

```
0000000001111111111222222222233333333334444444444555555555566666666677777777778
1234567890123456789012345678901234567890123456789012345678901234567890123456789
@MAP,X PART,PROVE
```

Example 22

This card must appear immediately after all subroutines and before the @XQT PROVE card. The program analyst also has the option to make a new tape that would contain any program modifications, as was discussed earlier. Simply punch "T" for tape at card column 11 of the Type 1 card. It is recommended that a trial run be made first without the "T" option so as to validate any program modifications. When such modifications are working correctly, utilize the "T" option. Figure 15 illustrates how the entire deck setup would look with sample program modifications in the runstream.

```
0000000021111111111222222222333333333344444444445555555556666666667777777778
123456789012345678901234567890123456789012345678901234567890123456789012345678901...
@RUN,//T GSEDMS,431020,XXXXXXBIN207,3,200/1500      *GSE DATA MANAGEMENT SYSTEM
@RUN,//T GSEDMS,431020,XXXXXXBIN207,3,200/1500
@ASG,T GSE-OLD-TAPE.,T,YYYYY
@REWIND GSE-OLD-TAPE.
@ASG,T 1,F2
@ASG,T 2,F2
@ASG,T 3,F2
@ASG,T 4,F2
@ASG,T CONTROL-FILE,F/1/TRK/10
@COPIN GSE-OLD-TAPE.,TPFS.
@COPY,G GSE-OLD-TAPE.,CONTROL-FILE.
@COPY,G GSE-OLD-TAPE.,1.
@DATA,IL 2.

@END
@PDP,ULF TABLE,TABLE
-98,98
     .            ,'SRB   ','SOLID ','ROCKET',' BST. '
-100,100
     .            ,'SPSH  ','   SPA','CE SHU','TTLE  '
@FOR,S MASTER,MASTER
@FOR,S HEADR1,HEADR1
@FOR,S HEADR2,HEADR2
@FOR,S SORT1,SORT1
@FOR,S PRINT1,PRINT1
@FOR,US MAIN,MAIN
-59,59
   50 FORMAT(1H1,5X,19H***  MNEMONIC NAME ,A6,17H IS MISSPELLED***)
-62,62
   57 CDESRT(I)=10
-63,63
@FOR,US MERGE,MERGE
-22,22
      NUMSR1=MAXSRT+2
      NUMSR1=MAXSRT+2
      IF(CDESRT(I).LE.0.OR.CDESRT(I).GT.NUMSR1)GO TO 68
-24,24
      GO TO (8,16,24,32,40,48,56,59,74,60),NARGU
@FOR,US UPDATE,UPDATE
-119
      IF(NCARD.EQ.1.AND.NCR.EQ.0.AND.EINUM(2).EQ.BLANK)GO TO 30
@MAP,X PART,PROVE
@XQT PROVE
04/18/73
SUMARY
STAGE1
STAGE2
MEDIAS    HE
FUNCTL    SERVIC
CLASSC    1
USELOC    MSFC
APPDOC    651CD9200
@FIN
```

Figure 15. Control card deck with program modifications.

Program Characteristics

The GSE Data Management program has several special features, one of which is the use of the Univac Sort/Merge Package. It consists of a collection of Fortran-callable subroutines which allow the program to interface with the 1108 System Sort Program. Subroutines UPDATE, SORT1 through SORT7 use this Sort Package. For more details about this package a complete write-up is available from the publishing organization.

Another feature of this program is the use of the Fortran V procedure table. It allows storage of any array information that is used as common reference among the 28 routines. Each routine can therefore use whatever procedure table it needs in order to gain access to desired specifications. Most modifications the program analyst may wish to make to the procedure table do not necessitate changes to those routines that use that table. Remember that the program is also designed in such a way that it can handle programming additions. The program presently uses 23 000 words of core with a maximum of 32 000 words allowable for remote operations. The data bank currently allows for a total of 540 GSE-EI entries but can be enlarged to fit the need for more, merely by changing the specifications on the four @ASG, TN, F2 control cards. Computer execution time per sort ranges from roughly 13 seconds for a relatively smaller output such as the Master Summary Table, to 26 seconds for one of the more typical sorts (Stage Sort1).

Description of the Program Listing

Each routine of the GSE Data Management Program is commented in a standard format so that changes or additions can be easily made by experienced programmers. As can be seen in Appendix B, the current program listing is given for each routine. Appearing first within each routine are comment statements that briefly describe its function to the program. Following this initial description are comment statements which identify and define the important variables in the subroutine. Also note that among the actual Fortran program statements are added comments that give better detail to the significant operations of the subprogram. Special attention should be given to the information provided in the comment portion of each procedure table. Conveniently located there is a description of its function, any defintions of important variables, and a cross reference to those routines that use the table.

APPENDIX A. DEFINITION OF INPUT FORMAT FOR GSE ATTRIBUTE DATA CARDS

APPENDIX A. DEFINITION OF INPUT FORMAT FOR GSE ATTRIBUTE DATA CARDS

Input to the GSE information bank is supplied by data punched in sets of four or more 80-column computer cards. The format for these cards is shown in Figure A-1. Figure A-2 shows an example of all four card types. Care should be taken when using the various Descriptor Codes explained in the following sections. These descriptor Codes are the variables used to determine the specific information which will be placed in four of the eight possible sorts. The Descriptor Codes used must contain no blank spaces within each code and must contain no zeros. (NOTE: All information should be punched left justified unless otherwise stated.)

Card Type 1

1. Card Number — This number identifies that this is the first of a set of four or more data cards (card column 1 punch a 1).

2. Project Number — This digit code number represents the general project in which the GSE item is associated. Presently the five codes used are as follows (card column 4):

Code	Project
1	Saturn
2	Skylab
3	Shuttle
4	HEAO[1]
5	Spacelab

3. EI Number — This 13-maximum-digit number should be left justified (card columns 7-19).

1. High Energy Astronomy Observatory.

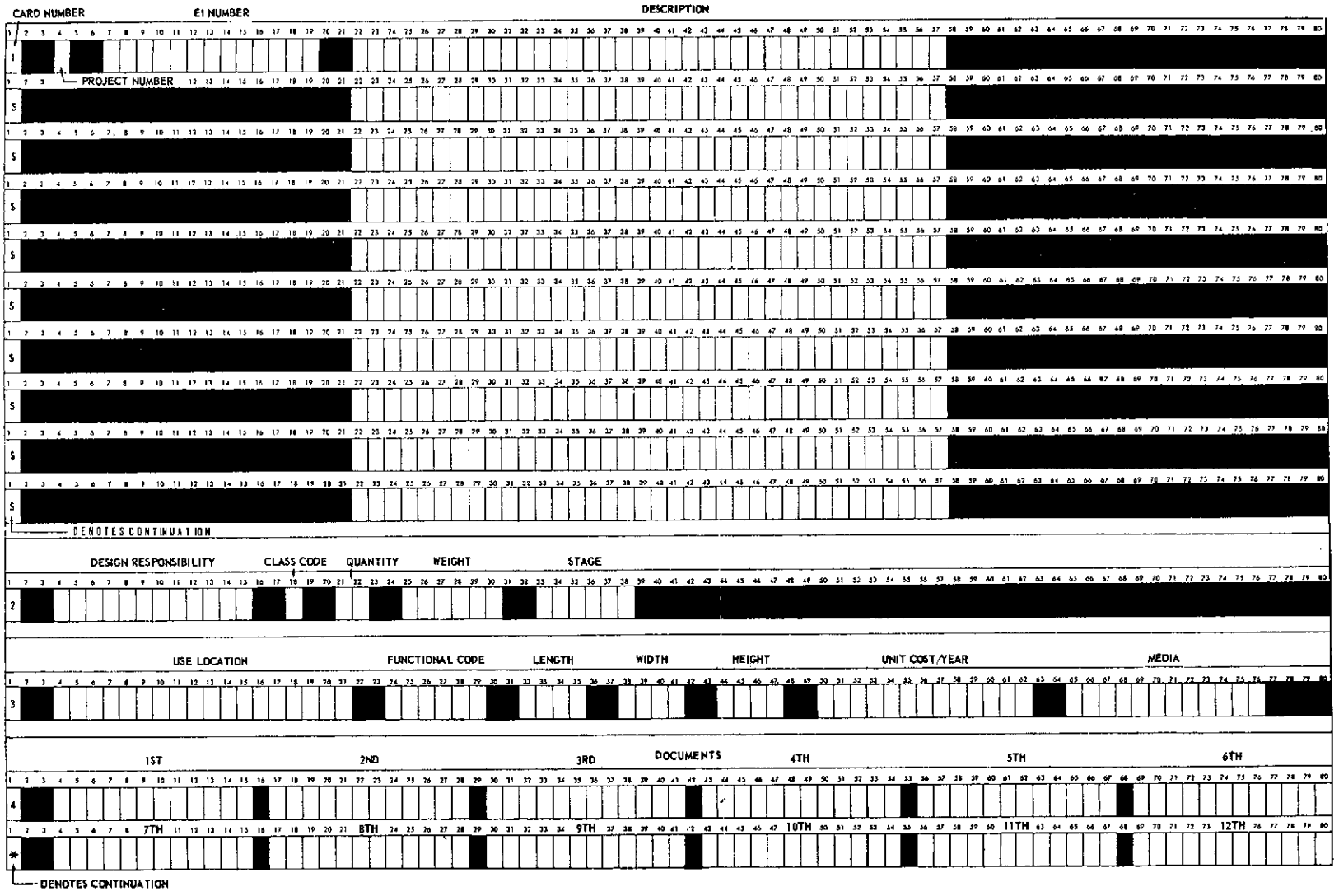

Figure A-1. GSE input format sheet.

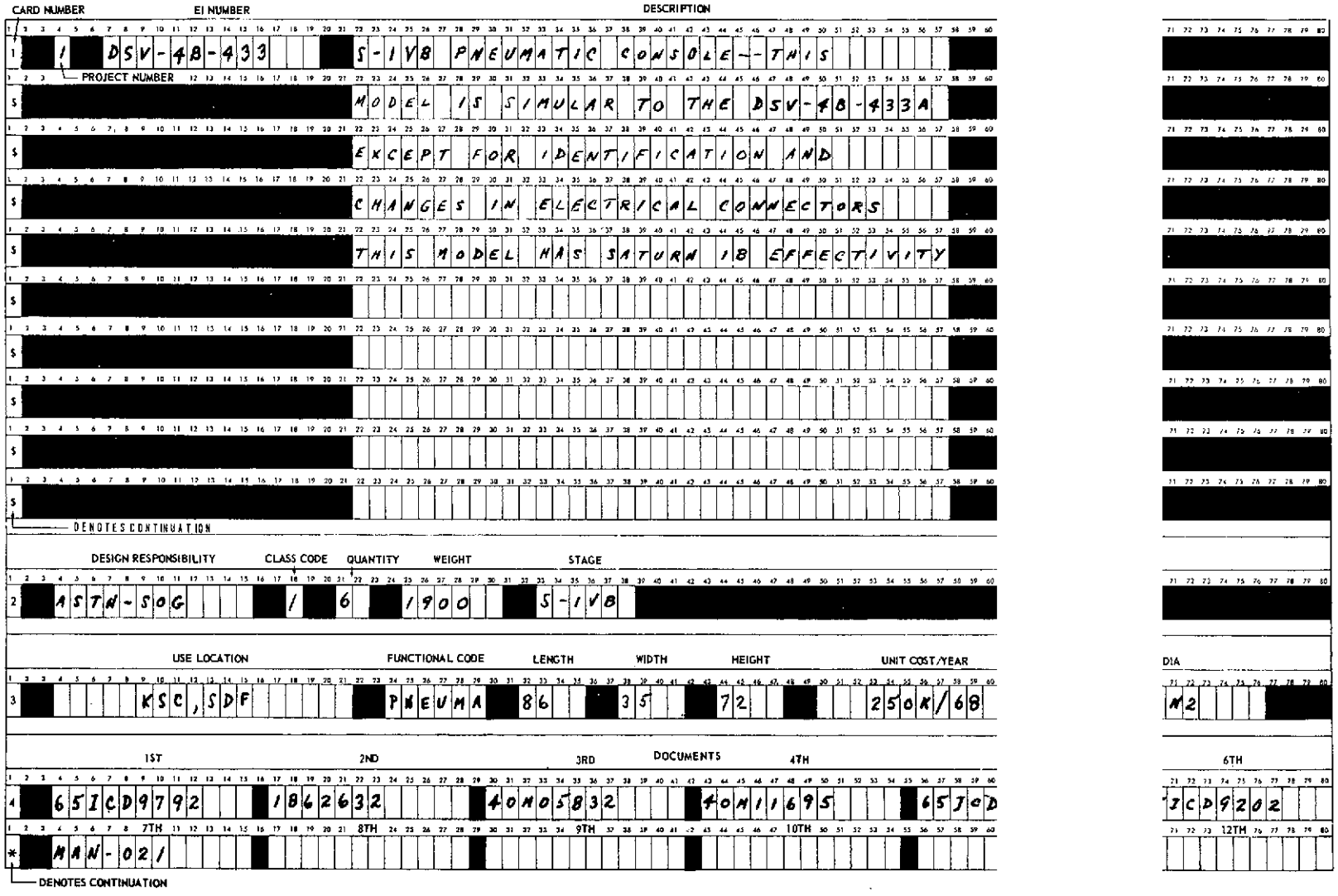

Figure A-2. Example of GSE filled-in data sheet.

4. Title — This information space provides the first available line for the writing of a brief description for GSE associated with the above EI number. Care should be taken in order that this line of description does not utilize more than 36 spaces. Characters should be punched in columns 22 through 57 inclusive, since the computer program will not accept characters outside this limit.

5. Continuation of Card Type 1 — In order to continue the description or remarks, a new card must be punched with a dollar sign in column 1 and, on that same card, the continuation of the title description should be punched in columns 22 through 57 inclusive. Take note not to exceed the designed card space.

6. Each continuation card is identical. The maximum number of continuation cards is 19 (i.e., a total of 20 description cards is allowable including the continuations).

Card Type 2

1. Card Number — Punch a number 2 in column 1.

2. Design Responsibility — This is the organization which is directly responsible for the actual design of an item of GSE. This organization may or may not be the manufacturer. The code is left justified in columns 4 through 15 inclusive (examples: Rocketdyne, ASTN-EPG, MDAC-WD).

3. Class Code — This integer, punched in column 18, represents the major hardware category that is supported by a particular piece of GSE. The code must be one of the following:

Descriptor Code	Name
1	Stage Systems
2	Payload Module
3	Experiment

4. Quantity — This is the number of pieces of GSE that fit this particular description. This integer should be placed left justified in columns 21 and 22.

5. Weight In Pounds — The weight is a left justified, six-digit-maximum integer in columns 25 through 30. It weight is not known, use TBD; right justify entry.

6. Stage — This attribute defines the stage or vehicle assembly that the GSE supports. The present program will key off an alphanumeric or mnemonic code. All mnemonics for these stages are left justified, six-or-less-character words that must conform exactly to one of the following:

Code	Module or Stage
AM	Airlock Module
ATM	Apollo Telescope Mount
ET	Shuttle External Tank
IU	Instrument Unit
MDA	Multiple Docking Adapter
OWS	Orbital Workshop
SRB	Shuttle Solid Rocket Booster
SSME	Space Shuttle Main Engine
S-1C	S-IC Stage
S-2	S-II Stage
S-IVB	S-IVB Stage

Other codes can be added to the system based on programming instructions explained in Section IV.

Card Type 3

1. Card Number — Keypunch a 3 in column 1.

2. Use Location — This describes the area of usability for the particular GSE item. Multiple entries of Descriptor Codes are permitted with each

code separated by a comma. The Descriptor Codes should be centered using columns 4 through 21 and must be as follows:

Descriptor Code	Description
CP	Canoga Park
HB	MDAC-WD, Huntington Beach
IBM	IBM Space Division
KSC	Kennedy Center
MAF	Michoud Assembly Facility
MFG	Manufacturing Facility
MSC	Houston Center
MSFC	Marshall Center
MTF	Mississippi Test Facility
SDF	System Development Facility
STF	Static Test Facility
WTR	Western Test Range
FACT	Factory

Other location codes can be added to the program based on instructions given in Section IV.

3. Functional Code — This code describes the basic function that the GSE will perform for the associated piece of hardware. The Descriptor Code must be in columns 24 through 29 and must be one of the following:

Descriptor Code	Function Name
ACCESS	Access
GASDET	Gas Detector
HANDLE	Handling

PNEUMA	Pneumatics
STORAG	Storage
REFRIG	Refrigeration
SERVIC	Service Gear
TESTCH	Test and Checkout
TRANSP	Transportation
VAC PU[2]	Vacuum Pump

4. Length in inches is placed in columns 22 through 25. If length is not known, use TBD; right justify entry.

5. Width in inches is placed in columns 28 through 41. If width is not known, use TBD; right justify entry.

6. Height in inches is placed in columns 44 through 47. If height is not known, use TBD; right justify entry.

7. Unit Cost — The cost of design, development, and procurement of one GSE piece of gear and year of purchase is given (example: 320K 71). Center the value in the 13 characters provided in columns 50 through 62. If cost is not known, enter TBD in columns 54 through 56.

8. Media — The type of fluid medium that is used with the GSE is given (examples: GN2, RP-1, He). Multiple entries of fluid media Descriptor Codes are allowed with each code separated by a comma. Use columns 65 through 76 and center the entry(s) in the columns provided. The Descriptor Codes are as follows:

Descriptor Code	Description
COOL	Coolanol
ETHGLY	Ethylene Glycol
FREON	Freon
GH2	Gaseous Hydrogen

2. Note blank space in code.

GN2	Gaseous Nitrogen
GO2	Gaseous Oxygen
HE	Helium
HYD	Hydraulic Fluid
H2O	Water
H2OGLY	Water Glycol
ISOALC	Isopropyl Alcohol
LH2	Liquid Hydrogen
LN2	Liquid Nitrogen
ME/H2O	Methanol Water
MMH	Monomethyl Hydrazine
N2O4	Nitgogen Tetraoxide
ORONIT	Oronite
PRES	Preservative Oil
RJ-1	RJ-1
RP-1	RP-1
TRIC	Trichloroethylene
VAC	Vacuum

Make sure that the letter O is punched and not the digit 0. Descriptor Codes contain no zeroes (0).

If no length, width, height, cost, and/or fluid media are associated with the GSE item, punch N/A in the assigned columns.

Card Type 4

1. Card Number — Punch a 4 in column 1.

2. Document Number — This card provides room for up to six document numbers, each number being up to 12 characters long. If a document number is less than 12 characters, leave the remaining spaces blank. Starting in column 4 on the same card, punch the first document number, ending in column 15. Skip a space, that is leave column 16 blank. If a second document exists, use columns 17 through 28. Skip a space and then a third document number. Continue this trend for the remainder of the card, if needed, always noting to skip a space between document numbers.

3. Document Continuation Card — If more than 6 documents are needed needed, a second document card is required. Only one continuation card is allowed. Punch an asterisk in column 1 and pattern the document numbers exactly as stated in item 2 above.

APPENDIX B. GSE DATA MANAGEMENT SYSTEM PROGRAM LISTING

SUBROUTINE DATECK

```
      C ***********************************************************************DTCK    1
      C * THIS ROUTINE CHECKS TO SEE IF A VALID NUMERIC DATE HAS BEEN          *DTCK    2
      C * SUPPLIED BY THE PROGRAM ANALYST                                      *DTCK    3
      C ***********************************************************************DTCK    4
            SUBROUTINE DATECK(NERROR)                                          DTCK    5
            INCLUDE COMON                                                      DTCK    6
      C                                                                        DTCK    7
      C*****DEFINITION OF FUNCTION FLD(I,K,E)  I=STARTING BIT K=BIT WIDTH      DTCK    8
      C*****E=WORD NAME                                                        DTCK    9
      C                                                                        DTCK   10
            WORD=DATX(I)                                                       DTCK   11
            NERROR=0                                                           DTCK   12
      C                                                                        DTCK   13
      C*****STORE THE 1ST CHARACTER OF DATX(I) IN VARIABLE NPART1              DTCK   14
      C                                                                        DTCK   15
            NPART1=FLD(0,6,WORD)                                               DTCK   16
      C                                                                        DTCK   17
      C*****STORE THE 3RD CHARACTER OF DATX(I) IN VARIABLE NPART3              DTCK   18
      C                                                                        DTCK   19
            NPART2=FLD(6,6,WORD)                                               DTCK   20
      C                                                                        DTCK   21
      C*****STORE THE 2ND CHARACTER OF DATX(I) IN VARIABLE NPART2              DTCK   22
      C                                                                        DTCK   23
            NPART3=FLD(12,6,WORD)                                              DTCK   24
      C                                                                        DTCK   25
      C*****CHECK TO SEE IF THE 1ST CHARACTER IS A BLANK                       DTCK   26
      C                                                                        DTCK   27
            IF(NPART1.EQ.5)GO TO 20                                            DTCK   28
            NVALUE=NPART1-48                                                   DTCK   29
      C                                                                        DTCK   30
      C*****CHECK TO SEE IF THE 1ST CHARACTER IS A VALID INTEGER               DTCK   31
      C                                                                        DTCK   32
            IF(NVALUE.GT.9.OR.NVALUE.LT.0)GO TO 30                             DTCK   33
      C                                                                        DTCK   34
      C*****CHECK TO SEE IF THE 2ND CHARACTER IS A SLASH(/)                    DTCK   35
      C                                                                        DTCK   36
            IF(NPART2.EQ.60)GO TO 35                                           DTCK   37
      C                                                                        DTCK   38
      C*****CHECK TO SEE IF THE 2ND CHARACTER IS A BLANK                       DTCK   39
      C                                                                        DTCK   40
      20    IF(NPART2.EQ.5)GO TO 29                                            DTCK   41
            NVALUE=NPART2-48                                                   DTCK   42
      C                                                                        DTCK   43
      C*****CHECK TO SEE IF THE 2ND CHARACTER IS A VALID INTEGER               DTCK   44
      C                                                                        DTCK   45
            IF(NVALUE.GT.9.OR.NVALUE.LT.0)GO TO 30                             DTCK   46
            GO TO 35                                                           DTCK   47
      C                                                                        DTCK   48
      C*****CHECK TO SEE IF THE 3RD CHARACTER IS A BLANK                       DTCK   49
      C                                                                        DTCK   50
      29    IF(NPART3.NE.5)GO TO 35                                            DTCK   51
      C                                                                        DTCK   52
      C*****SET ERROR KEY                                                      DTCK   53
      C                                                                        DTCK   54
      30    NERROR=1                                                           DTCK   55
      35    RETURN                                                             DTCK   56
            END                                                                DTCK   57
```

SUBROUTINE DIMMET

```
      C ***********************************************************************DMMT    1
      C *  SUBROUTINE DIMMET -THIS SUBROUTINE CONVERTS THE DIMENSIONS OF      *DMMT    2
      C * LENGTH,WIDTH,AND HEIGHT TO CENTIMETERS FROM INCHES                   *DMMT    3
      C ***********************************************************************DMMT    4
      C                                                                        DMMT    5
            SUBROUTINE DIMMET(IMETIC,INA)                                      DMMT    6
            DIMENSION IM(4)                                                    DMMT    7
      C                                                                        DMMT    8
      C*****FLD IS A LIBRARY FUNCTION TO OBTAIN BITS FROM ALPHANUMERIC NUMBERSDMMT    9
      C                                                                        DMMT   10
            IM(1)=FLD(0,6,INA)                                                 DMMT   11
            IM(2)=FLD(6,6,INA)                                                 DMMT   12
            IM(3)=FLD(12,6,INA)                                                DMMT   13
            IM(4)=FLD(18,6,INA)                                                DMMT   14
            IADD=0                                                             DMMT   15
            DO 12 I=1,4                                                        DMMT   16
            IF(IM(I).NE.5) GO TO 10                                            DMMT   17
            IADD=IADD+1                                                        DMMT   18
            GO TO 12                                                           DMMT   19
         10 IF(IM(I).EQ.25) GO TO 13                                           DMMT   20
            IF(IM(I).EQ.19) GO TO 13                                           DMMT   21
         12 CONTINUE                                                           DMMT   22
            IF(IADD.EQ.0) GO TO 15                                             DMMT   23
            I=0                                                                DMMT   24
            DO 50 J=1,IADD                                                     DMMT   25
            IF(IM(J).NE.5) GO TO 51                                            DMMT   26
            I=I+1                                                              DMMT   27
         50 CONTINUE                                                           DMMT   28
         51 GO TO (16,17,18,13),IADD                                           DMMT   29
         13 IMETIC=INA                                                         DMMT   30
            GO TO 99                                                           DMMT   31
      C                                                                        DMMT   32
      C*****CALCULATE INTEGER NUMBER(DEPENDING ON NUMBER OF DIGITS)            DMMT   33
      C                                                                        DMMT   34
         15 IREAL=(IM(4)-48)+(IM(3)-48)*10+(IM(2)-48)*100+(IM(1)-48)*1000      DMMT   35
            GO TO 19                                                           DMMT   36
         16 IREAL=(IM(I+3)-48)+(IM(I+2)-48)*10+(IM(I+1)-48)*100                DMMT   37
            GO TO 19                                                           DMMT   38
         17 IREAL=(IM(I+2)-48)+(IM(I+1)-48)*10                                 DMMT   39
            GO TO 19                                                           DMMT   40
         18 IREAL=IM(I+1)-48                                                   DMMT   41
         19 REAL=IREAL                                                         DMMT   42
      C                                                                        DMMT   43
      C*****CONVERT REAL NUMBER TO METRIC                                      DMMT   44
      C                                                                        DMMT   45
            XMETIC=REAL*2.54                                                   DMMT   46
      C                                                                        DMMT   47
      C*****CALL SUBROUTINE TO CONVERT REAL NUMBER TO ALPHANUMERIC NUMBER      DMMT   48
      C                                                                        DMMT   49
            CALL RELALP(IMETIC,XMETIC,4)                                       DMMT   50
         99 RETURN                                                             DMMT   51
            END                                                                DMMT   52
```

SUBROUTINE HEADR1

```
C ********************************************************************HDR1
C *  SUBROUTINE HEADR1 -THIS ROUTINE PRINTS THE COLUMN HEADINGS AND    *HDR1
C * THE CLASS CODE FOOTNOTES THAT ARE PECULIAR TO STAGE SORT 1         *HDR1
C ********************************************************************HDR1
C                                                                     HDR1
C                VARIABLE                   DEFINITION                HDR1
C                ********                   **********                HDR1
C                NKEY            DETERMINES WHEN THE CLASS CODE       HDR1
C                                FOOTNOTE IS TO BE WRITTEN            HDR1
C                                                                     HDR1
C                IEND            DETERMINES WHEN TO WRITE BOTH FOOTNOTES HDR1
C                                ON THE FINAL PAGE OF THE SORT        HDR1
C                                                                     HDR1
C                JKEY            SET EQUAL TO ''.FALSE.'' FOR PERFORMING HDR1
C                                STAGE SORT 2                         HDR1
      COMPILER(DATA=SHORT)                                            HDR1
      SUBROUTINE HEADR1                                               HDR1
      INCLUDE XSTAGE,LIST                                             HDR1
      INCLUDE COMON                                                   HDR1
      INTEGER P                                                       HDR1
      LOGICAL JKEY                                                    HDR1
      COMMON/ONE/NKEY                                                 HDR1
      COMMON/TWO/IEND,JKEY,PKEY                                       HDR1
      P=6                                                             HDR1
    1 IF(IEND-1)25,2,2                                                HDR1
    2 WRITE(P,264)                                                    HDR1
      WRITE(P,265)                                                    HDR1
      JKEY=.FALSE.                                                    HDR1
    5 GO TO 275                                                       HDR1
C                                                                     HDR1
C*****WRITE THE CLASS CODE MNEMONIC REFERENCE AT THE BOTTOM OF THE PAGE HDR1
C                                                                     HDR1
   25 NPAGE=NPAGE+1                                                   HDR1
      IF(NKEY-1) 40,35,36                                             HDR1
   35 WRITE(P,264)                                                    HDR1
   36 WRITE(P,265)                                                    HDR1
      NKEY=2                                                          HDR1
      GO TO 42                                                        HDR1
   40 NKEY=1                                                          HDR1
   42 WRITE(P,272)                                                    HDR1
   43 WRITE(P,260)                                                    HDR1
C                                                                     HDR1
C*****WRITE OUT THE DATE AND THE PAGE NUMBER                          HDR1
C                                                                     HDR1
   45 WRITE(P,215)DATX,PROGM,NPAGE                                    HDR1
C                                                                     HDR1
C***  DETERMINE WHICH MODULE(STAGE) TITLE IS TO BE PRINTED            HDR1
C***  SUBSCRIPTED VARIABLE ''GSETLE'' CONTAINS ALL GSE TITLE HEADINGS HDR1
C                                                                     HDR1
   60 J=MOD*4                                                         HDR1
   65 I=J-2                                                           HDR1
   70 WRITE(P,220)(GSETLE(K), K=I,J)                                  HDR1
C                                                                     HDR1
```

SUBROUTINE HEADR1 (Concluded)

```
      C*****PRINT THE GENERAL COLUMN HEADINGS                                  HDR1  54
      C                                                                        HDR1  55
        75  WRITE(P,225)                                                      HDR1  56
        80  WRITE(P,230)                                                      HDR1  57
        83  WRITE(P,235)                                                      HDR1  58
        84  WRITE(P,240)                                                      HDR1  59
       215  FORMAT(     8X,A6,A2,39X,2A6,1X,7HPROJECT,43X,4HPAGE,1X,I2)       HDR1  60
       220  FORMAT(/,57X,3A6)                                                 HDR1  61
       225  FORMAT(1X,131(1H*))                                               HDR1  62
       230  FORMAT(60X,5HCLASS,3X ,10HFUNCTIONAL,19X,3HUSE,17X,10HAPPLICABLE) HDR1  63
       235  FORMAT(3X,9HE1 NUMBER,20X,11HDESCRIPTION,18X,4HCODE,6X,4HCODE,20X,HDR1  64
           *     8HLOCATION,15X,9HDOCUMENTS)                                 HDR1  65
       240  FORMAT(1X,131(1H*))                                               HDR1  66
       260  FORMAT( )                                                         HDR1  67
       264 FORMAT(//1X,'N/A OR 'NA' = NOT APPLICABLE')                        HDR1  68
       265  FORMAT(1X,'CLASS CODES:   1.STAGE SYSTEMS   2.PAYLOAD MODULE   3.EHDR1  69
           *XPERIMENT')                                                      HDR1  70
       272  FORMAT(1H1,54X,'STAGE SORT ONE TABLE')                            HDR1  71
       275  RETURN                                                            HDR1  72
            END                                                               HDR1  73
```

SUBROUTINE HEADR2


```
C ***********************************************************************HDR2   1
C *  SUBROUTINE HEADR2 -THIS ROUTINE PRINTS THE STAGE SORT 2 COLUMN     *HDR2   2
C * HEADINGS AT THE TOP OF EACH PAGE                                    *HDR2   3
C ***********************************************************************HDR2   4
C                                                                        HDR2   5
C                VARIABLE                  DEFINITION                    HDR2   6
C                ********                  **********                    HDR2   7
C                PKEY          DETERMINES WHEN THE ''NOT APPPLICABLE''   HDR2   8
C                              FOOTNOTE IS TO BE WRITTEN                 HDR2   9
C                                                                        HDR2  10
C                IEND          DETERNINES WHEN TO WRITE THE LAST FOOTNOTE HDR2  11
C                              ON THE FINAL PAGE OF THE SORT             HDR2  12
C                                                                        HDR2  13
C                JKEY          SET EQUAL TO ''.FALSE.'' FOR PERFORMING   HDR2  14
C                              STAGE SORT 1                              HDR2  15
C                                                                        HDR2  16
      COMPILER(DATA=SHORT)                                               HDR2  17
      SUBROUTINE HEADR2                                                  HDR2  18
      INCLUDE XSTAGE,LIST                                                HDR2  19
      INCLUDE COMON                                                      HDR2  20
      INTEGER P,PKEY                                                     HDR2  21
      LOGICAL JKEY                                                       HDR2  22
      COMMON/TWO/IEND,JKEY,PKEY                                          HDR2  23
      P = 6                                                              HDR2  24
      IF(IEND-1)25,20,20                                                 HDR2  25
   20 WRITE(P,264)                                                       HDR2  26
      JKEY=.FALSE.                                                       HDR2  27
      GO TO 275                                                          HDR2  28
   25 NPAGE=NPAGE+1                                                      HDR2  29
      IF(PKEY-1)40,35,36                                                 HDR2  30
   35 WRITE(P,264)                                                       HDR2  31
   36 PKEY=2                                                             HDR2  32
      GO TO 42                                                           HDR2  33
   40 PKEY=1                                                             HDR2  34
   42 WRITE(P,272)                                                       HDR2  35
      WRITE(P,260)                                                       HDR2  36
C                                                                        HDR2  37
C*****WRITE THE DATE AND THE PAGE NUMBER                                 HDR2  38
C                                                                        HDR2  39
      WRITE(P,215)DATX,PROGM,NPAGE                                       HDR2  40
      J=MOD*4                                                            HDR2  41
      I=J-2                                                              HDR2  42
C                                                                        HDR2  43
C*****WRITE THE GSE TITLE                                                HDR2  44
C                                                                        HDR2  45
      WRITE(P,220)(GSETLE(K), K=I,J)                                     HDR2  46
      WRITE(P,225)                                                       HDR2  47
      WRITE(P,230)                                                       HDR2  48
      WRITE(P,235)                                                       HDR2  49
      WRITE(P,240)                                                       HDR2  50
  215 FORMAT(8X,A6,A2,39X,2A6,1X,7HPROJECT,43X,4HPAGE,1X,I2)             HDR2  51
  220 FORMAT(/57X,3A6)                                                   HDR2  52
  225 FORMAT(1X,131(1H*))                                                HDR2  53
  230 FORMAT(83X,'ENVELOPE DIM',14X,'UNIT',6X,'DESIGN')                  HDR2  54
  235 FORMAT(3X,'EI NUMBER',18X,'DESCRIPTION',20X,'MEDIA',7X,'WT',       HDR2  55
     *7X,'L',6X,'W',6X,'H',3X,'QTY',6X,'COST/YR',2X,'RESPONSIBILITY')    HDR2  56
  240 FORMAT(1X,131(1H*))                                                HDR2  57
  260 FORMAT( )                                                          HDR2  58
  264 FORMAT(//1X,'N/A OR NA - NOT APPLICABLE')                          HDR2  59
  272 FORMAT(1H1,54X,'STAGE SORT TWO TABLE')                             HDR2  60
  275 RETURN                                                             HDR2  61
      END                                                                HDR2  62
```

SUBROUTINE HEADR3

```
      C ********************************************************************HDR3    1
      C *  SUBROUTINE HEADR3 -THE FUNCTION OF THIS ROUTINE IS TO PRINT THE   *HDR3    2
      C * COLUMN HEADINGS AT THE TOP OF EACH PAGE OF THE FLUID MEDIA SORT    *HDR3    3
      C ********************************************************************HDR3    4
      C                                                                      HDR3    5
      C                VARIABLE                DEFINITION                    HDR3    6
      C                ********                **********                    HDR3    7
      C                                                                      HDR3    8
      C                MESAVE          SUBSCRIPT OF ENTRY IN FLUID MEDIA MNEMONIC HDR3    9
      C                                CODE TABLE                            HDR3   10
      C                                                                      HDR3   11
            COMPILER(DATA=SHORT)                                             HDR3   12
            SUBROUTINE HEADR3                                                HDR3   13
            INCLUDE COMON                                                    HDR3   14
            INCLUDE XMETAB,LIST                                              HDR3   15
            COMMON/THREE/MESAVE                                              HDR3   16
            NPAGE = NPAGE + 1                                                HDR3   17
            WRITE(6,100)                                                     HDR3   18
        100 FORMAT(1H1,59),'MEDIA SORT TABLE')                               HDR3   19
            WRITE(6,150) DATX,NPAGE                                          HDR3   20
        150 FORMAT(/8X,A6,A2,46X,'ALL PROGRAMS',47X,'PAGE',1X,I2)            HDR3   21
            JJ = MESAVE + 4                                                  HDR3   22
            KK = MESAVE + 6                                                  HDR3   23
            WRITE(6,200)(TABMED(MM),MM = JJ,KK)                              HDR3   24
        200 FORMAT(/59X,3A6)                                                 HDR3   25
            WRITE(6,250)                                                     HDR3   26
        250 FORMAT(1X,13(1H*))                                               HDR3   27
            WRITE(6,260)                                                     HDR3   28
        260 FORMAT(68X,'FUNCTIONAL',16X,'ENVELOPE DIM',14X,'UNIT',6X,'DESIGN')HDR3   29
            WRITE(6,270)                                                     HDR3   30
        270 FORMAT(3X,'EI NUMBER',18X,'DESCRIPTION',20X,'CODE ',7X,'WT',     HDR3   31
           *7X,'L',6X,'W',6X,'H',3X,'QTY',6X,'COST/YR',2X,'RESPONSIBILITY')  HDR3   32
            WRITE(6,250)                                                     HDR3   33
            RETURN                                                           HDR3   34
            END                                                              HDR3   35
```

SUBROUTINE HEADR4

```
      C ********.**.*******.*****************************************************HDR4    1
      C *  SUBROUTINE HEADR4 -THE FUNCTION OF THIS ROUTINE IS TO PRINT THE    *HDR4    2
      C * COLUMN HEADINGS AT THE TOP OF EACH PAGE OF THE FUNCTIONAL CODE SORT*HDR4    3
      C ********.**.*******.*****************************************************HDR4    4
      C                                                                         HDR4    5
      C              VARIABLE                  DEFINITION                     HDR4    6
      C              ********                  **********                     HDR4    7
      C                                                                         HDR4    8
      C              ISAVE          SUBSCRIPT OF ENTRY IN FUNCTIONAL CODE      HDR4    9
      C                             MNEMONIC TABLE                             HDR4   10
      C                                                                        - HDR4   11
            COMPILER(DATA=SHORT)                                              HDR4   12
            SUBROUTINE HEADR4                                                 HDR4   13
            INCLUDE COMON                                                     HDR4   14
            INCLUDE FUNTAB.LIST                                               HDR4   15
            COMMON/FOUR/ISAVE                                                 HDR4   16
            NPAGE = NPAGE + 1                                                 HDR4   17
            WRITE(6,100)                                                      HDR4   18
        100 FORMAT('1H1',54X,'FUNCTIONAL SORT TABLE')                         HDR4   19
            WRITE(6,150)DATX,NPAGE                                            HDR4   20
        150 FORMAT(/8X,A5,A2,43X,'ALL PROGRAMS',47X,'PAGE',1X,I2)             HDR4   21
            JJ = ISAVE + 1                                                    HDR4   22
            KK = ISAVE + 4                                                    HDR4   23
            WRITE(6,200)(FUCOTA(MM),MM = JJ,KK)                               HDR4   24
        200 FORMAT(/54X,4A6)                                                  HDR4   25
            WRITE(6,250)                                                      HDR4   26
        250 FORMAT(1X,13('1H*'))                                              HDR4   27
            WRITE(6,260)                                                      HDR4   28
        260 FORMAT(8JX,'ENVELOPE DIM',14X,'UNIT',6X,'DESIGN')                 HDR4   29
            WRITE(6,270)                                                      HDR4   30
        270 FORMAT(3X,'E| NUMBER',18X,'DESCRIPTION',20X,'MEDIA',7X,'WT',      HDR4   31
           *7X,'L',6X,'W',6X,'H',3X,'QTY',6X,'COST/YR',2X,'RESPONSIBILITY')   HDR4   32
            WRITE(6,250)                                                      HDR4   33
            RETURN                                                            HDR4   34
            END                                                               HDR4   35
```

SUBROUTINE HEADR5

```
C ****************************************************************
C *  SUBROUTINE HEADR5 -THE FUNCTION OF THIS ROUTINE IS TO PRINT THE   *
C * COLUMN HEADINGS AT THE TOP OF EACH PAGE OF THE CLASS CODE SORT    *
C ****************************************************************
C
C                  VARIABLE                 DEFINITION
C                  ********                 **********
C
C                  IHAVIT          SUBSCRIPT OF ENTRY IN CLASS CODE MNEMONIC
C                                  TABLE
C
      COMPILER(DATA=SHORT)
      SUBROUTINE HEADR5
      INCLUDE COMON
      INCLUDE CLASCD,LIST
      COMMON/FIVE/IHAVIT
      NPAGE = NPAGE + 1
      WRITE(6,100)
  100 FORMAT(1H1,54X,'CLASS CODE SORT TABLE')
      WRITE(6,150) DATX,NPAGE
  150 FORMAT(/8X,A6,A2,43X,'ALL PROGRAMS',47X,'PAGE',1X,I2)
      JJ = IHAVIT + 1
      KK = IHAVIT + 3
      WRITE(6,200)(CLACOD(MM), MM = JJ,KK)
  200 FORMAT(/60X,3A6)
      WRITE(6,250)
  250 FORMAT(1X,131(1H*))
      WRITE(6,260)
  260 FORMAT(64X,'FUNCTION',43X,'DESIGN')
      WRITE(6,270)
  270 FORMAT(3X,'EI NUMBER',20X,'DESCRIPTION',23X,'CODE',9X,'STAGE',11X,
     1'MEDIA',9X,'QTY',5X,'RESPONSIBILITY')
      WRITE(6,250)
      RETURN
      END
```

SUBROUTINE HEADR6

```
C ****************************************************************************HDR6
C * SUBROUTINE TO PRINT HEADING FOR APPLICABLE DOCUMENT SORT TABLE          *HDR6
C ****************************************************************************HDR6
C                                                                          HDR6
C                 VARIABLE                    DEFINITION                   HDR6
C                 ********                    **********                   HDR6
C                 IFHR6          DETERMINES WHEN THE HEADING IS TO BE      HDR6
C                                PRINTED FOR EACH PAGE OF THE APPLICABLE   HDR6
C                                DOCUMENT SORT                             HDR6
C                                IF IFHR6 VALUE IS ZERO,PRINT HEADING ONLY HDR6
C                                IF IFHR6 VALUE IS GREATER THAN ZERO,PRINT HDR6
C                                FOOTNOTES AND THEN A NEW HEADING          HDR6
      COMPILER(DATA=SHORT)                                                 HDR6
      SUBROUTINE HEADR6                                                    HDR6
      INCLUDE COMON                                                        HDR6
      INCLUDE CODEST                                                       HDR6
      COMMON/SIX/ IFHR6                                                    HDR6
      IF(IFHR6-1) 10,11,11                                                 HDR6
   10 CONTINUE                                                             HDR6
      IFHR6 = IFHR6 + 1                                                    HDR6
   12 CONTINUE                                                             HDR6
      NPAGE = NPAGE + 1                                                    HDR6
C                                                                          HDR6
C*****PRINT MAIN HEADING AT TOP OF EACH PAGE                               HDR6
C                                                                          HDR6
      WRITE(6,800)                                                         HDR6
  800 FORMAT(1H1,51X,'APPLICABLE DOCUMENT SORT TABLE')                     HDR6
      WRITE(6,801) DATX,DESCOD(NUMBER),DESCO2(NUMBER),NPAGE                HDR6
  801 FORMAT(/8X,A6,A2,38X,'DOCUMENT  ',2A6,42X,'PAGE ',I2)                HDR6
      WRITE(6,805)                                                         HDR6
  805 FORMAT(/59X,'ALL PROGRAMS')                                          HDR6
      WRITE (6,802)                                                        HDR6
  802 FORMAT(1X,131(1H*))                                                  HDR6
      WRITE (6,803)                                                        HDR6
  803 FORMAT(78X,'CLASS',9X,'FUNCTIONAL',15X,'APPLICABLE')                 HDR6
      WRITE(6,804)                                                         HDR6
  804 FORMAT(3X,'EI NUMBER',28X,'DESCRIPTION',28X,'CODE',12X,              HDR6
     1'CODE',19X,'DOCUMENTS')                                              HDR6
      WRITE(6,802)                                                         HDR6
C                                                                          HDR6
C*****RETURN TO SUBROUTINE HEADS                                           HDR6
C                                                                          HDR6
      RETURN                                                               HDR6
   11 CONTINUE                                                             HDR6
C                                                                          HDR6
C*****PRINT FOOTNOTE AT BOTTOM OF EACH PAGE                                HDR6
C                                                                          HDR6
      WRITE(6,806)                                                         HDR6
  806 FORMAT(1X,'CLASS CODES:   1.STAGE SYSTEMS   2.PAYLOAD MODULE    ',   HDR6
     *'3.EXPERIMENT')                                                      HDR6
      GO TO 12                                                             HDR6
      END                                                                  HDR6
```

SUBROUTINE HEADR7

```
C *****************************************************************HDR7
C *  SUBROUTINE HEADR7 -THE FUNCTION OF THIS ROUTINE IS TO PRINT THE   *HDR7
C * COLUMN HEADINGS AT THE TOP OF EACH PAGE OF THE USE LOCATION SORT   *HDR7
C *****************************************************************HDR7
C                                                                   HDR7
C               VARIABLE                 DEFINITION                 HDR7
C               ********                 **********                 HDR7
C                                                                   HDR7
C               KESAVE         SUBSCRIPT OF ENTRY IN USE LOCATION MNEMONICHDR7
C                              CODE TABLE                           HDR7
C                                                                   HDR7
      COMPILER(DATA=SHORT)                                          HDR7
      SUBROUTINE HEADR7                                             HDR7
      INCLUDE COMON                                                 HDR7
      INCLUDE USLOCT,LIST                                           HDR7
      COMMON/SEVEN/KESAVE                                           HDR7
      NPAGE = NPAGE + 1                                             HDR7
      WRITE(6,100)                                                  HDR7
  100 FORMAT(1H1,52X,'USE LOCATION SORT TABLE')                     HDR7
      WRITE(6,150) DATX,NPAGE                                       HDR7
  150 FORMAT(/8X,A6,A2,43X,'ALL PROGRAMS',47X,'PAGE',1X,I2)         HDR7
      JJ = KESAVE + 3                                               HDR7
      KK = KESAVE + 6                                               HDR7
      WRITE(6,200)(PLTAB(MM),MM = JJ,KK)                            HDR7
  200 FORMAT(/54X,5A6)                                              HDR7
      WRITE(6,250)                                                  HDR7
  250 FORMAT(1X,13)(1H*))                                           HDR7
      WRITE(6,260)                                                  HDR7
 260  FORMAT(58X,'FUNCTIONAL',15X,'ENVELOPE DIM',24X,        'DESIGN')HDR7
      WRITE(6,270)                                                  HDR7
 270  FORMAT(3X,'EI NUMBER',18X,'DESCRIPTION',20X,'CODE ',7X,'WT', HDR7
     *7X,'L',6X,'W',6X,'H',8X,'MEDIA',          8X,'RESPONSIBILITY')HDR7
      WRITE(6,250)                                                  HDR7
      RETURN                                                        HDR7
      END                                                           HDR7
```

SUBROUTINE HEADS

```
      SUBROUTINE HEADS                                                  HDS   1
C ***********************************************************************HDS   2
C *  SUBROUTINE HEADS CALLS THE APPRROPRIATE HEADING SUBROUTINE FOR    *HDS   3
C * USE BY THE RESPECTIVE PRINT ROUTINE                                *HDS   4
C ***********************************************************************HDS   5
      INCLUDE CODEST,LIST                                               HDS   6
      GO TO (8,16,24,32,40,48,56),NARGU                                 HDS   7
    8 CONTINUE                                                          HDS   8
      CALL HEADR1                                                       HDS   9
      GO TO 80                                                          HDS  10
   16 CONTINUE                                                          HDS  11
      CALL HEADR2                                                       HDS  12
      GO TO 80                                                          HDS  13
   24 CONTINUE                                                          HDS  14
      CALL HEADR3                                                       HDS  15
      GO TO 80                                                          HDS  16
   32 CONTINUE                                                          HDS  17
      CALL HEADR4                                                       HDS  18
      GO TO 80                                                          HDS  19
   40 CONTINUE                                                          HDS  20
      CALL HEADR5                                                       HDS  21
      GO TO 80                                                          HDS  22
   48 CONTINUE                                                          HDS  23
      CALL HEADR6                                                       HDS  24
      GO TO 80                                                          HDS  25
   56 CONTINUE                                                          HDS  26
      CALL HEADR7                                                       HDS  27
      GO TO 80                                                          HDS  28
   80 CONTINUE                                                          HDS  29
      RETURN                                                            HDS  30
      END                                                               HDS  31
```

SUBROUTINE LINCON

```
      C ***********************************************************************LNCN    1
      C *  SUBROUTINE LINCON -THIS ROUTINE DETERMINES WHETHER TO START A NEW *LNCN    2
      C * PAGE OR CONTINUE WRITING ON THE CURRENT PAGE                        *LNCN    3
      C ***********************************************************************LNCN    4
      C                                                                        LNCN    5
      C               VARIABLE                DEFINITION                     LNCN    6
      C               ********                **********                     LNCN    7
      C                                                                        LNCN    8
      C               LADD          IS THE LINE COUNTER USED BY ALL ROUTINES LNCN    9
      C                                                                        LNCN   10
            COMPILER(DATA=SHORT)                                               LNCN   11
            SUBROUTINE LINCON                                                  LNCN   12
            INCLUDE COMON                                                      LNCN   13
      C                                                                        LNCN   14
      C*****DETERMINE IF THIS IS THE TOP OF A NEW PAGE                         LNCN   15
      C                                                                        LNCN   16
        80  IF(LADD-1) 90,90,100                                              LNCN   17
         90 CONTINUE                                                         LNCN   18
            LADD = 10                                                        LNCN   19
            CALL HEADS                                                       LNCN   20
            RETURN                                                           LNCN   21
        100 CONTINUE                                                         LNCN   22
            ITEST = NLINES                                                   LNCN   23
            LADD1 = LADD + ITEST                                             LNCN   24
      C                                                                      LNCN   25
      C*****DETERMINE IF THERE IS ENOUGH SPACE ON THE CURRENT PAGE TO        LNCN   26
      C***** COMPLETE PRINTOUT                                               LNCN   27
      C                                                                      LNCN   28
            IF(LADD1-49) 10,20,20                                            LNCN   29
         10 CONTINUE                                                         LNCN   30
            RETURN                                                           LNCN   31
         20 CONTINUE                                                         LNCN   32
            LADD = 0                                                         LNCN   33
      C                                                                      LNCN   34
      C*****WRITE THE FOOTNOTE AT THE BOTTOM OF THE PAGE AND PRINT THE NEW   LNCN   35
      C***** PAGE HEADING                                                    LNCN   36
      C                                                                      LNCN   37
        200  WRITE(6,201)                                                   LNCN   38
        201  FORMAT(//1X,'N/A OR NA = NOT APPLICABLE')                       LNCN   39
        250  CALL HEADS                                                     LNCN   40
             LADD=10                                                         LNCN   41
        260  RETURN                                                         LNCN   42
             END                                                            LNCN   43
```

MAIN PROGRAM

```
C ********************************************************************MAIN   1
C *  MAIN -THIS IS THE MAINLINE TO THE GSE DATA MANAGEMENT PROGRAM      *MAIN   2
C *                                                                    *MAIN   3
C *        FUNCTIONS OF THE MAINLINE:                                  *MAIN   4
C *                                                                    *MAIN   5
C *  (1) READ IN INFORMATION PROVIDED BY THE USER ON THE PROGRAM       *MAIN   6
C *      INSTRUCTION DATA CARDS                                        *MAIN   7
C *                                                                    *MAIN   8
C *  (2) DETERMINE WHETHER TO UPDATE THE MASTER DATA FILE              *MAIN   9
C *                                                                    *MAIN  10
C *  (3) PROCESS, SORT, AND MERGE ALL GSE INFORMATION BY MEANS OF ITS  *MAIN  11
C *      SUPPORT ROUTINES                                              *MAIN  12
C *                                                                    *MAIN  13
C *  (4) DETERMINE WHETHER OR NOT TO GENERATE A NEW TAPE               *MAIN  14
C ********************************************************************MAIN  15
      INCLUDE MAIN,LIST                                                 MAIN  16
      INCLUDE COMON,LIST                                                MAIN  17
      INCLUDE CODEST,LIST                                               MAIN  18
      LOGICAL ANSWER                                                    MAIN  19
      INTEGER READ,WRITE                                                MAIN  20
      DIMENSION CIMAGE(4)                                               MAIN  21
      DATA CIMAGE/*ADD CONTROL-FILE*                                    MAIN  22
      READ=5                                                            MAIN  23
      WRITE=6                                                           MAIN  24
      NARGU=0                                                           MAIN  25
C                                                                       MAIN  26
C*****READ IN THE PROGRAM INSTRUCTION VARIABLES                         MAIN  27
C                                                                       MAIN  28
    1 READ(READ,4)DATX,ANSWER                                           MAIN  29
    4 FORMAT(A6,A2,2X,A1)                                               MAIN  30
C                                                                       MAIN  31
C*****DETERMINE IF THE DATE VARIABLE IS VALID                           MAIN  32
C                                                                       MAIN  33
      CALL DATECK(NERROR)                                               MAIN  34
      IF(NERROR.EQ.1)GO TO 73                                           MAIN  35
    5 CONTINUE                                                          MAIN  36
C                                                                       MAIN  37
C*****READ IN THE SORT MNEMONICS CODES                                  MAIN  38
C                                                                       MAIN  39
      DO 7 I=1,MAXSRT                                                   MAIN  40
      READ(READ,6,ERR=73,END=10)CDESRT(I),DESCOD(I),DESCO2(I)           MAIN  41
    6 FORMAT(A6,4X,2A6)                                                 MAIN  42
      NUMSRT=I                                                          MAIN  43
    7 CONTINUE                                                          MAIN  44
   10 CONTINUE                                                          MAIN  45
C                                                                       MAIN  46
C*****UPDATE THE MASTER DATA FILE                                       MAIN  47
C                                                                       MAIN  48
   15 CALL UPDATE                                                       MAIN  49
C                                                                       MAIN  50
C*****CONVERT THE REQUESTED SORT MNEMONICS TO INTEGER CODES             MAIN  51
C                                                                       MAIN  52
   20 DO 60 I=1,NUMSRT                                                  MAIN  53
```

MAIN PROGRAM (Concluded)

```
54          25 DO 40 J=1,9                                                  MAIN  54
55          30 IF(CDESRT(I).EQ.XMOD(J))CDESRT(I)=J                          MAIN  55
56          35 IF(CDESRT(I).EQ.J) GO TO 60                                  MAIN  56
57          40 CONTINUE                                                     MAIN  57
58          45 WRITE(WRITE,50) CDESRT(I)                                    MAIN  58
59          50 FORMAT(1HI,5X,19H*** MNEMONIC NAME ,A6,17H IS MISSPELLED***) MAIN  59
60          55 WRITE(WRITE,56)                                              MAIN  60
61          56 FORMAT(//,5X,'*** CONTINUING TO PROCESS OTHER SORTS ***')    MAIN  61
62          57 CDESRT(I)=10                                                 MAIN  62
63          60 CONTINUE                                                     MAIN  63
64       C                                                                  MAIN  64
65       C*****PERFORM SORTS                                                MAIN  65
66       C                                                                  MAIN  66
67          65 CALL MERGE                                                   MAIN  67
68       C                                                                  MAIN  68
69       C*****DETERNINE IF A NEW TAPE IS TO BE GENERATED                   MAIN  69
70       C                                                                  MAIN  70
71             NTEST=FLD(0,6,ANSWER)                                        MAIN  71
72             IF(NTEST.EQ.25) GO TO 70                                     MAIN  72
73             IF(NTEST.EQ.5) GO TO 75                                      MAIN  73
74             WRITE(6,100)                                                 MAIN  74
75          100  FORMAT(1HI,10X,'*** ERROR IN TAPE OPTION INSTRUCTION ..PAN INCORRE MAIN  75
76              ICT CHARACTER WAS USED ***')                                MAIN  76
77             WRITE(6,101)                                                 MAIN  77
78          101  FORMAT(//10X,'*** NO TAPE CREATED ON THIS RUN ***')        MAIN  78
79             69 GO TO 75                                                  MAIN  79
80       C                                                                  MAIN  80
81       C*****ADD TO THE RUNSTREAM THE SPECIAL CONTROL CARDS NEEDED TO GENERATE MAIN  81
82       C*****A NEW TAPE                                                   MAIN  82
83       C                                                                  MAIN  83
84          70 CALL CSPREQ(IMAGE)                                           MAIN  84
85             WRITE(6,71)                                                  MAIN  85
86          71 FORMAT(1HI,10X,'<<<<< USER MESSAGE... AN OUTPUT TAPE HAS BEEN CREA MAIN  86
87            *TED ON THIS RUN >>>>>')                                      MAIN  87
88          72 GO TO 75                                                     MAIN  88
89          73 WRITE(WRITE,74)                                              MAIN  89
90          74 FORMAT(10X,'*** ERROR IN PROGRAM INSTRUCTIONS ...THE DATE IS MISS MAIN  90
91            *ING OR IS PUNCHED INCORRECTLY***')                           MAIN  91
92          75 STOP                                                         MAIN  92
93          80 END                                                          MAIN  93
```

SUBROUTINE MASTER

```
      C ***********************************************************************MSTR    1
      C *  MASTER -THIS ROUTINE PRINTS OUT A SUMMARY OF ALL GSE ENTITIES      *MSTR    2
      C * CONTAINED IN THE MASTER DATA FILE                                   *MSTR    3
      C ***********************************************************************MSTR    4
      C                                                                        MSTR    5
            COMPILER(DATA=SHORT)                                               MSTR    6
            SUBROUTINE MASTER                                                  MSTR    7
            INCLUDE COMON                                                      MSTR    8
            INCLUDE EQUIV                                                      MSTR    9
            INCLUDE PROGRM                                                     MSTR   10
            INCLUDE XSTAGE                                                     MSTR   11
            REWIND 4                                                           MSTR   12
     1      READ(4,END=99)WORDS                                               MSTR   13
            WRITE(6,100)                                                       MSTR   14
      100   FORMAT(1H1)                                                       MSTR   15
            NCHECK=0                                                           MSTR   16
            WRITE(6,105)                                                       MSTR   17
      105   FORMAT(30X,'MASTER SUMMARY TABLE',//)                             MSTR   18
            WRITE(6,101)                                                       MSTR   19
      101   FORMAT(1X,80(1H*))                                                MSTR   20
            WRITE(6,102)                                                       MSTR   21
      102   FORMAT(5X,'ENTITY #   EI  NUMBER      PROGRAM                STAGE  MSTR   22
           *      REFERENCE DATE')                                           MSTR   23
            WRITE(6,101)                                                       MSTR   24
            GO TO 9                                                            MSTR   25
      3     CONTINUE                                                         MSTR   26
      C                                                                        MSTR   27
      C*****READ THE MASTER DATA FILE                                          MSTR   28
      C                                                                        MSTR   29
            READ(4,END=99)WORDS                                                MSTR   30
      9     NRED=NRED+1                                                      MSTR   31
      C                                                                        MSTR   32
      C*****DETERMINE IF A VALID PROGRAM CODE EXISTS                           MSTR   33
      C                                                                        MSTR   34
            IF(NPRO.GT.5.OR.NPRO.LE.0)NPRO=6                                   MSTR   35
            IF(NPRO.EQ.6) WRITE(6,106)                                         MSTR   36
       106  FORMAT(10X,'***ERROR*** PROGRAM CODE VALUE IS INVALID')           MSTR   37
            L=NPRO *2                                                          MSTR   38
            M=L-1                                                              MSTR   39
            DO 10 I=1,NSTAGE                                                   MSTR   40
            J=I*4-3                                                            MSTR   41
      C                                                                        MSTR   42
      C*****DETERMINE WHAT STAGE NAME IS TO BE PRINTED OUT                     MSTR   43
      C                                                                        MSTR   44
            IF(MOD.EQ.GSETLE(J))MOD=I                                          MSTR   45
            IF(MOD.EQ.1)GO TO 11                                               MSTR   46
      10    CONTINUE                                                         MSTR   47
            WRITE(6,104)                                                       MSTR   48
      104   FORMAT(10X,' ***ERROR***STAGE NAME COULD NOT BE DETERMINED')     MSTR   49
            NCHECK=NCHECK+1                                                    MSTR   50
            MOD=1                                                              MSTR   51
      11    J=MOD*4                                                          MSTR   52
            I=J-2                                                              MSTR   53
            WRITE(6,103)NRED,EINUM,(AGENDU(K),K=M,L),(GSETLE(K),K=I,J),      MSTR   54
           * DATE(1),DATE(2)                                                   MSTR   55
      103   FORMAT(6X,I4,5X,2A6,A1,3X,2A6,3X,3A6,3X,A6,A2)                   MSTR   56
            NCHECK=NCHECK+1                                                    MSTR   57
            IF(NCHECK.EQ.45)GO TO 1                                            MSTR   58
            GO TO 3                                                          MSTR   59
      99    RETURN                                                           MSTR   60
            END                                                              MSTR   61
```

SUBROUTINE MERGE

```
      C ********************************************************************MERG    1
      C *  SUBROUTINE MERGE -THIS ROUTINE DETERMINES WHICH SORT LEG TO       *MERG    2
      C * PERFORM. THE ARRAY CALLED ''CDESRT'' REPRESENTS THE FOLLOWING      *MERG    3
      C * INTEGER CODES:                                                     *MERG    4
      C *                                                                    *MERG    5
      C *            INTEGER CODE              TYPE OF SORT                  *MERG    6
      C *            ************              ************                  *MERG    7
      C *                  1                   STAGE SORT 1                  *MERG    8
      C *                  2                   STAGE SORT 2                  *MERG    9
      C *                  3                   FLUID MEDIA SORT              *MERG   10
      C *                  4                   FUNCTIONAL CODE SORT          *MERG   11
      C *                  5                   CLASS CODE SORT               *MERG   12
      C *                  6                   APPLICABLE DOCUMENT SORT      *MERG   13
      C *                  7                   USE LOCATION SORT             *MERG   14
      C *                  8                   MASTER SUMMARY TABLE          *MERG   15
      C *                                                                    *MERG   16
      C ********************************************************************MERG   17
            SUBROUTINE MERGE                                                 MERG   18
            INCLUDE CODEST,LIST                                              MERG   19
            DO 60 I=1,NUMSRT                                                 MERG   20
            NUMBER=I                                                         MERG   21
            NUMSRI=MAXSRT+2                                                  MERG   22
            IF(CDESRT(I).LE.0.OR.CDESRT(I).GT.NUMSRI)GO TO 68                MERG   23
          3 NARGU=CDESRT(I)                                                  MERG   24
            GO TO (8,16,24,32,40,48,56,59,74,60),NARGU                       MERG   25
          8 CALL SORT1                                                       MERG   26
          9 WRITE(6,10)                                                      MERG   27
         10 FORMAT(1H1)                                                      MERG   28
         12 GO TO 60                                                         MERG   29
         16 CALL SORT2                                                       MERG   30
         17 WRITE(6,10)                                                      MERG   31
         20 GO TO 60                                                         MERG   32
         24 CALL SORT3                                                       MERG   33
         25 WRITE(6,10)                                                      MERG   34
         28 GO TO 60                                                         MERG   35
         32 CALL SORT4                                                       MERG   36
         33 WRITE(6,10)                                                      MERG   37
         36 GO TO 60                                                         MERG   38
         40 CALL SORT5                                                       MERG   39
         41 WRITE(6,10)                                                      MERG   40
         44 GO TO 60                                                         MERG   41
         48 CALL SORT6                                                       MERG   42
         49 WRITE(6,10)                                                      MERG   43
         52 GO TO 60                                                         MERG   44
         56 CALL SORT7                                                       MERG   45
         57 WRITE(6,10)                                                      MERG   46
            GO TO 60                                                         MERG   47
         59 CALL MASTER                                                      MERG   48
            WRITE(6,10)                                                      MERG   49
            GO TO 60                                                         MERG   50
         60 CONTINUE                                                         MERG   51
         64 GO TO 74                                                         MERG   52
         68 WRITE(6,72)                                                      MERG   53
         72 FORMAT(10X,'***ERROR***  CURRENT SORT COULD NOT BE PERFORMED DUE TMERG   54
           1O AN INCORRECT INTEGER SORT CODE IN SUBROUTINE MERGE')           MERG   55
         73 GO TO 76                                                         MERG   56
         74 WRITE(6,75)NUMBER                                                MERG   57
         75 FORMAT(10X,'*** THIS RUN COMPLETED',I3,' SORTS ***')             MERG   58
         76 RETURN                                                           MERG   59
         80 END                                                              MERG   60
```

MAP ELEMENT PART

```
1        LIB SYS$*MSFC$.
2       IN MAIN
3       IN MBFO$$
```

SUBROUTINE PRINT1

```
C ***********************************************************************PRT1    1
C *  SUBROUTINE PRINT1 -THIS ROUTINE PRINTS CERTAIN GSE ATTRIBUTES FOR *PRT1    2
C * THOSE E1 NUMBERS TO BE FOUND IN STAGE SORT 1                       *PRT1    3
C ***********************************************************************PRT1    4
C                                                                       PRT1    5
C               VARIABLE                DEFINITION                      PRT1    6
C               ********                **********                      PRT1    7
C                                                                       PRT1    8
C               NKEY            TELLS SUBROUTINE HEADR1 WHEN TO WRITE THE PRT1    9
C                               CLASS CODE FOOTNOTE                     PRT1   10
C                                                                       PRT1   11
      COMPILER(DATA=SHORT)                                              PRT1   12
      SUBROUTINE PRINT1                                                 PRT1   13
      INCLUDE XSTAGE,LIST                                               PRT1   14
      INCLUDE COMON                                                     PRT1   15
      INTEGER P,OLD,BASE,RULE                                           PRT1   16
      COMMON/ONE/NKEY                                                   PRT1   17
      DATA BLANK/'                                                      PRT1   18
      P=6                                                               PRT1   19
      NX=6                                                              PRT1   20
      LINES=1                                                           PRT1   21
C                                                                       PRT1   22
C*****DETERMINE IF THE STAGE NAME HAS CHANGED                           PRT1   23
C                                                                       PRT1   24
    7   IF(MOD.EQ.OLD)GO TO 86                                          PRT1   25
        LADD=0                                                          PRT1   26
        IF(OLD.EQ.0)    GO TO 22                                        PRT1   27
   16   NKEY=1                                                          PRT1   28
   18   GO TO 24                                                        PRT1   29
   22   NKEY=0                                                          PRT1   30
   24   CALL LINCON                                                     PRT1   31
C                                                                       PRT1   32
C*****DETERMINE WHETHER THE NUMBER OF APPLICABLE DOCUMENTS IS GREATER   PRT1   33
C*****THAN THE NUMBER OF LINES OF DESCRIPTION                           PRT1   34
C                                                                       PRT1   35
   86   DO 89 I=1,24,2                                                  PRT1   36
   87   IF(APDOC(I)-BLANK)88,90,88                                      PRT1   37
   88   JTAB=I                                                          PRT1   38
   89   CONTINUE                                                        PRT1   39
   90   NAPDOC=(JTAB+1)/2                                               PRT1   40
   91   IF(NAPDOC-NLINES)96,92,92                                       PRT1   41
   92   LDIFF=NAPDOC-NLINES                                             PRT1   42
        LADD=LADD+LDIFF                                                 PRT1   43
   96   CALL LINCON                                                     PRT1   44
C                                                                       PRT1   45
C*****WRITE THE FIRST LINE OF EQUIPMENT INFORMATION                     PRT1   46
C                                                                       PRT1   47
  100   WRITE(P,245)(EINUM(J),J=1,3),(TITLE(K),K=1,NX),NCODE,FCTCDE,USLOC,PRT1   48
     *                        (APDOC(N),N=1,2)                         PRT1   49
```

SUBROUTINE PRINT1 (Concluded)

```
      C                                                                 PRT1  50
      C*****WRITE THE REMAINING LINES OF DESCRIPTION                    PRT1  51
      C                                                                 PRT1  52
  115  IF(NLINES.EQ.1) GO TO 165                                        PRT1  53
  120  DO 155 LTYPE=2,NLINES                                            PRT1  54
  130  BASE=NX+1                                                        PRT1  55
  135  RULE=LTYPE*6                                                     PRT1  56
  136  IF(LTYPE.GT.12) GO TO 146                                        PRT1  57
  140  WRITE(P,250)(TITLE(JWORDS),JWORDS=BASE,RULE),(APDOC(2*LTYPE-1)),(APRT1  58
     *PDOC(2*LTYPE))                                                   PRT1  59
  141  GO TO 150                                                        PRT1  60
  146  WRITE(P,251)(TITLE(JWRITE),JWRITE=BASE,RULE)                     PRT1  61
  150  NX=RULE                                                          PRT1  62
  155  CONTINUE                                                         PRT1  63
  157  LINES=NLINES                                                     PRT1  64
  160  IF(LINES.GE.12) GO TO 185                                        PRT1  65
      C                                                                 PRT1  66
      C*****WRITE THE REMAINING APPLICABLE DOCUMENTS                    PRT1  67
      C                                                                 PRT1  68
  165  IF(APDOC(2*LINES+1)-BLANK) 170,185,170                           PRT1  69
  170  WRITE(P,255)(APDOC(2*LINES+1)),(APDOC(2*LINES+2))                PRT1  70
  175  LINES=LINES+1                                                    PRT1  71
  177  IF(LINES.GE.12) GO TO 185                                        PRT1  72
  180  GO TO 165                                                        PRT1  73
  185  WRITE(P,260)                                                     PRT1  74
  190  LADD=LADD+1+NLINES                                               PRT1  75
      C                                                                 PRT1  76
      C*****SAVE THE CURRENT STAGE NAME                                 PRT1  77
      C                                                                 PRT1  78
  200  OLD=MOD                                                          PRT1  79
  245  FORMAT(1X,2A6,A1,6X,6A6,6X,I3,5X,A6,14X,18A1,8X,2A6)            PRT1  80
  250  FORMAT(20X,6A6,60X,A6,A6)                                        PRT1  81
  251  FORMAT(20X,6A6)                                                  PRT1  82
  265  FORMAT(116X,A6,A6)                                               PRT1  83
  260  FORMAT( )                                                        PRT1  84
      C                                                                 PRT1  85
      C*****RETURN TO SORT 1  ROUTINE                                   PRT1  86
      C                                                                 PRT1  87
  275  RETURN                                                           PRT1  88
  280  END                                                              PRT1  89
```

SUBROUTINE PRINT2

```
C ********************************************************************PRT2   1
C *  SUBROUTINE PRINT2 -THIS ROUTINE PRINTS CERTAIN GSE ATTRIBUTES FOR *PRT2   2
C * THOSE EI NUMBERS TO BE FOUND IN STAGE SORT 2                       *PRT2   3
C ********************************************************************PRT2   4
C                                                                      PRT2   5
C               VARIABLE                DEFINITION                     PRT2   6
C               ********                **********                     PRT2   7
C                                                                      PRT2   8
C               PKEY            TELLS SUBROUTINE HEADR2 WHEN TO WRITE THE PRT2   9
C                               ''NOT APPLICABLE'' FOOTNOTE            PRT2  10
C                                                                      PRT2  11
      COMPILER(DATA=SHORT)                                             PRT2  12
      SUBROUTINE PRINT2                                                PRT2  13
      INCLUDE COMON                                                    PRT2  14
      INTEGER     P,OLD,PKEY                                           PRT2  15
      LOGICAL JKEY                                                     PRT2  16
      COMMON/TWO/IEND,JKEY,PKEY                                        PRT2  17
      P=6                                                              PRT2  18
      NX=6                                                             PRT2  19
   60 IF(MOD.EQ.OLD) GO TO 70                                          PRT2  20
   65 LADD=0                                                           PRT2  21
      IF(OLD.EQ.0) GO TO 68                                            PRT2  22
   66 PKEY=1                                                           PRT2  23
   67 GO TO 70                                                         PRT2  24
   68 PKEY=0                                                           PRT2  25
   70 CALL LINCON                                                      PRT2  26
      CALL WGTMET(IWGHTM,WGHT)                                         PRT2  27
      CALL DIMMET(ILNGTM,LNGTH)                                        PRT2  28
      CALL DIMMET(IWIDTM,WIDTH)                                        PRT2  29
      CALL DIMMET(IHGHTM,HGHT)                                         PRT2  30
  190 WRITE(P,375)(EINUM(J),J=1,3),(TITLE(K),K=1,NX),MEDIA,WGHT,LNGTH, PRT2  31
     1WIDTH,HGHT,QUAN,UNCST,DESCON                                     PRT2  32
  200 IF(NLINES.EQ.1)GO TO 300                                         PRT2  33
      IF(NLINES.EQ.2) GO TO 299                                        PRT2  34
      WRITE(6,501)(TITLE(K),K=7,12),IWGHTM,ILNGTM,IWIDTM,IHGHTM        PRT2  35
  501 FORMAT(18X,6A6,16X,A6,'KG',1X,A4,'CM',1X,A4,'CM',1X,A4,'CM')     PRT2  36
      NX=12                                                            PRT2  37
  210 DO 270 LYNER=3,NLINES                                            PRT2  38
      LSTART=NX+1                                                      PRT2  39
      NEND=LYNER*6                                                     PRT2  40
      WRITE(P,388)(TITLE(JWORDS),JWORDS=LSTART,NEND)                   PRT2  41
      NX=NEND                                                          PRT2  42
  270 CONTINUE                                                         PRT2  43
      LADD=LADD+NLINES                                                 PRT2  44
      GO TO 50                                                         PRT2  45
  299 CONTINUE                                                         PRT2  46
      WRITE(6,501)(TITLE(K),K=7,12),IWGHTM,ILNGTM,IWIDTM,IHGHTM        PRT2  47
      LADD=LADD+NLINES                                                 PRT2  48
      GO TO 50                                                         PRT2  49
  300 LADD=LADD+NLINES+1                                               PRT2  50
      WRITE(6,500) IWGHTM,ILNGTM,IWIDTM,IHGHTM                         PRT2  51
  500 FORMAT(70X,A6,'KG',1X,A4,'CM',1X,A4,'CM',1X,A4,'CM')             PRT2  52
   50 LADD=LADD+1                                                      PRT2  53
      WRITE(P,392)                                                     PRT2  54
      OLD=MOD                                                          PRT2  55
  375 FORMAT(1X,2A6,A1,4X,6A6,3X,12A1,1X,A6,'LB',1X,A4,'IN',1X,A4,     PRT2  56
     1'IN',1X,A4,'IN',1X,A2,2X,2A6,A1,2X,2A6)                          PRT2  57
  388 FORMAT (18X,6A6)                                                 PRT2  58
  392 FORMAT( )                                                        PRT2  59
  400 RETURN                                                           PRT2  60
  450 END                                                              PRT2  61
```

SUBROUTINE PRINT3

```
C ***********************************************************************PRT3   1
C *  SUBROUTINE PRINT3 -THIS ROUTINE PRINTS CERTAIN GSE ATTRIBUTES FOR *PRT3   2
C * THOSE EI NUMBERS TO BE FOUND IN THE FLUID MEDIA SORT               *PRT3   3
C ***********************************************************************PRT3   4
      COMPILER(DATA=SHORT)                                               PRT3   5
      SUBROUTINE PRINT3                                                  PRT3   6
      INCLUDE COMON                                                      PRT3   7
      NX = 6                                                             PRT3   8
      CALL WGTMET(IWGHTM,WGHT)                                           PRT3   9
      CALL DIMMET(ILNGTM,LNGTH)                                          PRT3  10
      CALL DIMMET(IWIDTM,WIDTH)                                          PRT3  11
      CALL DIMMET(IHGHTM,HGHT)                                           PRT3  12
      CALL LINCON                                                        PRT3  13
      WRITE(6,101)                                                       PRT3  14
  101 FORMAT(1X)                                                         PRT3  15
      LADD = LADD + 1                                                    PRT3  16
      WRITE(6,100) (EINUM(J),J=1,3),(TITLE(K),K=1,NX),FCTCDE,WGHT,LNGTH, PRT3  17
     *WIDTH,HGHT,QUAN,UNCST,DESCON                                       PRT3  18
  100 FORMAT(1X,2A6,A1,4X,6A6,5X,  A6,5X,A6,'LB',1X,A4,'IN',1X,A4,       PRT3  19
     1'IN',1X,A4,'IN',1X,A2,2X,2A6,A1,2X,2A6)                            PRT3  20
      NX=12                                                              PRT3  21
      LADD = LADD + 1                                                    PRT3  22
      IF(NLINES.EQ.1) GO TO 30                                           PRT3  23
      IF(NLINES.EQ.2) GO TO 40                                           PRT3  24
      WRITE(6,501)(TITLE(K),K=7,12),IWGHTM,ILNGTM,IWIDTM,IHGHTM          PRT3  25
  501 FORMAT(18X,6A6,16X,A6,'KG',1X,A4,'CM',1X,A4,'CM',1X,A4,'CM')       PRT3  26
      LADD=LADD+1                                                        PRT3  27
      GO TO 20                                                           PRT3  28
   30 WRITE(6,500) IWGHTM,ILNGTM,IWIDTM,IHGHTM                           PRT3  29
  500 FORMAT(70X,A6,'KG',1X,A4,'CM',1X,A4,'CM',1X,A4,'CM')               PRT3  30
   31 LADD=LADD+1                                                        PRT3  31
      RETURN                                                             PRT3  32
   40 WRITE(6,501)(TITLE(K),K=7,12),IWGHTM,ILNGTM,IWIDTM,IHGHTM          PRT3  33
      GO TO 31                                                           PRT3  34
   20 CONTINUE                                                           PRT3  35
      DO 50 I=3,NLINES                                                   PRT3  36
      ILOW = NX + 1                                                      PRT3  37
      IHIGH = I * 6                                                      PRT3  38
      WRITE(6,150)(TITLE(JW),JW = ILOW,IHIGH)                            PRT3  39
  150 FORMAT(18X,6A6)                                                    PRT3  40
      LADD = LADD + 1                                                    PRT3  41
      NX = IHIGH                                                         PRT3  42
   50 CONTINUE                                                           PRT3  43
      RETURN                                                             PRT3  44
      END                                                                PRT3  45
```

SUBROUTINE PRINT4

```
      C ***********************************************************************PRT4    1
      C *  SUBROUTINE PRINT4 -THIS ROUTINE PRINTS CERTAIN GSE ATTRIBUTES FOR *PRT4    2
      C * THOSE EI NUMBERS TO BE FOUND IN THE FUNCTIONAL CODE SORT            *PRT4    3
      C ***********************************************************************PRT4    4
            COMPILER(DATA=SHORT)                                               PRT4    5
            SUBROUTINE PRINT4                                                  PRT4    6
            INCLUDE COMON                                                      PRT4    7
            NX = 6                                                             PRT4    8
            CALL WGTMET(IWGHTM,WGHT)                                           PRT4    9
            CALL DIMMET(ILNGTM,LNGTH)                                          PRT4   10
            CALL DIMMET(IWIDTM,WIDTH)                                          PRT4   11
            CALL DIMMET(IHGHTM,HGHT)                                           PRT4   12
            CALL LINCON                                                        PRT4   13
            WRITE(6,101)                                                       PRT4   14
        101 FORMAT(1X)                                                         PRT4   15
            LADD = LADD + 1                                                    PRT4   16
            WRITE(6,100)(EINUM(J), J =1,3),(TITLE(K),K=1,NX),MEDIA,WGHT,LNGTH,PRT4   17
           1WIDTH,HGHT,JUAN,UNCST,DESCON                                       PRT4   18
        100 FORMAT(1X,2A6,A1,4X,6A6,3X,12A1,1X,A6,'LB',1X,A4,'IN',1X,A4,       PRT4   19
           1'IN',1X,A4,'IN',1X,A2,2X,2A6,A1,2X,2A6)                            PRT4   20
            NX=12                                                              PRT4   21
            LADD = LADD + 1                                                    PRT4   22
            IF(NLINES.EQ.1) GO TO 30                                           PRT4   23
            IF(NLINES.EQ.2) GO TO 40                                           PRT4   24
            WRITE(6,501)(TITLE(K),K=7,12),IWGHTM,ILNGTM,IWIDTM,IHGHTM          PRT4   25
        501 FORMAT(18X,6A6,16X,A6,'KG',1X,A4,'CM',1X,A4,'CM',1X,A4,'CM')       PRT4   26
            LADD=LADD+1                                                        PRT4   27
            GO TO 20                                                           PRT4   28
         30 WRITE(6,500) IWGHTM,ILNGTM,IWIDTM,IHGHTM                           PRT4   29
        500 FORMAT(70X,A6,'KG',1X,A4,'CM',1X,A4,'CM',1X,A4,'CM')               PRT4   30
         31 LADD=LADD+1                                                        PRT4   31
            RETURN                                                             PRT4   32
         40 WRITE(6,501)(TITLE(K),K=7,12),IWGHTM,ILNGTM,IWIDTM,IHGHTM          PRT4   33
            GO TO 31                                                           PRT4   34
         20 CONTINUE                                                           PRT4   35
            DO 50 I=3,NLINES                                                   PRT4   36
            ILOW = NX + 1                                                      PRT4   37
            IHIGH = I * 6                                                      PRT4   38
            WRITE(6,150)(TITLE(JWORDS),JWORDS =ILOW,IHIGH)                     PRT4   39
        150 FORMAT(18X,6A6)                                                    PRT4   40
            LADD = LADD + 1                                                    PRT4   41
            NX = IHIGH                                                         PRT4   42
         50 CONTINUE                                                           PRT4   43
            RETURN                                                             PRT4   44
            END                                                                PRT4   45
```

SUBROUTINE PRINT5

```
      C ***********************************************************************PRT5    1
      C *  SUBROUTINE PRINT5 -THIS ROUTINE PRINTS CERTAIN GSE ATTRIBUTES FOR *PRT5    2
      C * THOSE EI NUMBERS TO BE FOUND IN THE CLASS CODE SORT                 *PRT5    3
      C ***********************************************************************PRT5    4
            COMPILER(DATA=SHORT)                                              PRT5    5
            SUBROUTINE PRINT5                                                 PRT5    6
            INCLUDE COMMON                                                    PRT5    7
            NX = 6                                                            PRT5    8
            CALL LINCON                                                       PRT5    9
            WRITE(6,101)                                                      PRT5   10
        101 FORMAT(1X)                                                        PRT5   11
            LADD = LADD + 1                                                   PRT5   12
            WRITE(6,100)(EINUM(J),J=1,3),(TITLE(K),K=1,NX),FCTCDE,MOD,MEDIA,  PRT5   13
           1QUAN,DESCON                                                       PRT5   14
        100 FORMAT(1X,2A6,A1,6X,6A6,10X,A6,7X,A6,6X,12A1,6X,A2,6X,2A6)        PRT5   15
            LADD = LADD + 1                                                   PRT5   16
            LINEL = NLINES-1                                                  PRT5   17
            IF(LINEL-11 )0,20,20                                              PRT5   18
         10 CONTINUE                                                          PRT5   19
            RETURN                                                            PRT5   20
         20 CONTINUE                                                          PRT5   21
            DO 50   I = 2, NLINES                                             PRT5   22
            ILOW = NX+1                                                       PRT5   23
            IHIGH = I * 6                                                     PRT5   24
            WRITE(6,150)(TITLE(JWORDS),JWORDS = ILOW,IHIGH)                   PRT5   25
        150 FORMAT(20X,6A6)                                                   PRT5   26
            LADD = LADD + 1                                                   PRT5   27
            NX = IHIGH                                                        PRT5   28
         50 CONTINUE                                                          PRT5   29
            RETURN                                                            PRT5   30
            END                                                               PRT5   31
```

SUBROUTINE PRINT6

```
C ****************************************************************PRT6    1
C * SUBROUTINE TO PRINT OUT ALL INFORMATION ASSOCIATED WITH THE USER'S *PRT6    2
C * SPECIFIED APPLICABLE DOCUMENT(S)                                  *PRT6    3
C ****************************************************************PRT6    4
      COMPILER(DATA=SHORT)                                            PRT6    5
      SUBROUTINE PRINT6                                               PRT6    6
      INCLUDE COMJN                                                   PRT6    7
      DATA BLANK/'      '/                                            PRT6    8
C                                                                     PRT6    9
C*****CALL SUBROUTINE USED FOR LINE CONTROL                           PRT6   10
C                                                                     PRT6   11
      CALL LINCON                                                     PRT6   12
C                                                                     PRT6   13
C*****DETERMINE NUMBER OF APPLICABLE DOCUMENTS ASSOCIATED WITH EI NUMBERPRT6   14
C                                                                     PRT6   15
      DO 99 I=1,31,2                                                  PRT6   16
      IF(APDOC(I)-BLANK) 88,90,88                                     PRT6   17
   88 JTAB=I                                                          PRT6   18
   99 CONTINUE                                                        PRT6   19
   90 NAPDOC = (JTAB+1)/2                                             PRT6   20
      IF(NAPDOC-NLINES) 96,92,92                                      PRT6   21
C                                                                     PRT6   22
C*****MAKE CORRECTION IF NUMBER OF APPLICABLE DOCUMENTS IS GREATER    PRT6   23
C***** THAN NUMBER OF LINES OF DESCRIPTION                            PRT6   24
C                                                                     PRT6   25
   92 LDIFF = NAPDOC-NLINES                                           PRT6   26
      LADD = LADD+LDIFF                                               PRT6   27
C                                                                     PRT6   28
C*****CALL SUBROUTINE USED FOR LINE CONTROL                           PRT6   29
C                                                                     PRT6   30
   96 CALL LINCON                                                     PRT6   31
      LINESL = NLINES-1                                               PRT6   32
      NX = 6                                                          PRT6   33
      NXL = 1                                                         PRT6   34
      LADD = LADD + 1                                                 PRT6   35
C                                                                     PRT6   36
C*****PRINT OUT INFORMATION ASSOCIATED WITH SPECIFIED                 PRT6   37
C***** APPLICABLE DOCUMENT                                            PRT6   38
C                                                                     PRT6   39
      WRITE(6,900)(EINUM(J),J=1,3),(TITLE(K),K=1,NX),NCODE,FCTCDE,(APDOCPRT6   40
     1(N),N=1,2)                                                      PRT6   41
  900 FORMAT(1X,2A6,A1,14X,6A6,14X,I3,13X,A6,16X,2A6)                 PRT6   42
      I=3                                                             PRT6   43
   15 IF(APDOC(I)-BLANK)2,3,2                                         PRT6   44
    3 IF(LINESL-1)16,5,5                                              PRT6   45
    5 DO 10 N=1,LINESL                                                PRT6   46
      NXL=NXL+6                                                       PRT6   47
      NX = NX+6                                                       PRT6   48
      WRITE(6,901)(TITLE(K),K=NXL,NX)                                 PRT6   49
  901 FORMAT(28X,6A6)                                                 PRT6   50
      LADD = LADD + 1                                                 PRT6   51
   10 CONTINUE                                                        PRT6   52
   16 WRITE(6,904)                                                    PRT6   53
```

SUBROUTINE PRINT6 (Concluded)

```
      904 FORMAT ( )                                                    PRT6  54
          LADD = LADD + 1                                               PRT6  55
      C                                                                 PRT6  56
      C*****RETURN TO SUBROUTINE SORT6                                  PRT6  57
      C                                                                 PRT6  58
          RETURN                                                        PRT6  59
        2 IF(LINESL-1)6,7,7                                             PRT6  60
        6 L=I+1                                                         PRT6  61
          WRITE(6,903)(APDOC(J),J=I,L)                                  PRT6  62
      903 FORMAT (116X,2A6)                                             PRT6  63
          LADD = LADD + 1                                               PRT6  64
          I = I+2                                                       PRT6  65
          GO TO 15                                                      PRT6  66
        7 LINESL = LINESL-1                                             PRT6  67
          NXL=NXL+6                                                     PRT6  68
          NX = NX+6                                                     PRT6  69
          L=I+1                                                         PRT6  70
          WRITE (6,902) (TITLE(K),K=NXL,NX),(APDOC(J),J=I,L)            PRT6  71
      902 FORMAT(28X,6A6,52X,2A6)                                       PRT6  72
          LADD = LADD + 1                                               PRT6  73
          I=I+2                                                         PRT6  74
          GO TO 15                                                      PRT6  75
          END                                                           PRT6  76
```

SUBROUTINE PRINT7

```
C ***************************************************************************PRT7
C *  SUBROUTINE PRINT7 -THIS ROUTINE PRINTS CERTAIN GSE ATTRIBUTES FOR *PRT7
C * THOSE EI NUMBERS TO BE FOUND IN THE USE LOCATION SORT            *PRT7
C ***************************************************************************PRT7
      COMPILER(DATA=SHORT)                                                PRT7
      SUBROUTINE PRINT7                                                   PRT7
      INCLUDE COMON                                                       PRT7
      NX = 6                                                              PRT7
      CALL WGTMET(IWGHTM,WGHT)                                            PRT7
      CALL DIMMET(ILNGTM,LNGTH)                                           PRT7
      CALL DIMMET(IWIDTM,WIDTH)                                           PRT7
      CALL DIMMET(IHGHTM,HGHT)                                            PRT7
      CALL LINCON                                                         PRT7
      WRITE(6,101)                                                        PRT7
  101 FORMAT(1X)                                                          PRT7
      LADD = LADD + 1                                                     PRT7
      WRITE(6,100) (EINUM(J),J=1,3),(TITLE(K),K=1,NX),FCTCDE,WGHT,LNGTH,PRT7
     1WIDTH,HGHT,MEDIA,DESCON                                             PRT7
  100 FORMAT(1X,2A6,A1,4X,6A6,5X,  A6,5X,A6,'LB',1X,A4,'IN',1X,A4,       PRT7
     1'IN',1X,A4,'IN',4X,12A1,4X         ,2A6)                            PRT7
      NX=12                                                               PRT7
      LADD = LADD + 1                                                     PRT7
      IF(NLINES.EQ.1) GO TO 30                                            PRT7
      IF(NLINES.EQ.2) GO TO 40                                            PRT7
      WRITE(6,501)(TITLE(K),K=7,12),IWGHTM,ILNGTM,IWIDTM,IHGHTM           PRT7
  501 FORMAT(18X,6A6,16X,A6,'KG',1X,A4,'CM',1X,A4,'CM',1X,A4,'CM')         PRT7
      LADD=LADD+1                                                         PRT7
      GO TO 20                                                            PRT7
   30 WRITE(6,500) IWGHTM,ILNGTM,IWIDTM,IHGHTM                            PRT7
  500 FORMAT(70X,A6,'KG',1X,A4,'CM',1X,A4,'CM',1X,A4,'CM')                 PRT7
   31 LADD=LADD+1                                                         PRT7
      RETURN                                                              PRT7
   40 WRITE(6,501)(TITLE(K),K=7,12),IWGHTM,ILNGTM,IWIDTM,IHGHTM           PRT7
      GO TO 31                                                            PRT7
   20 CONTINUE                                                            PRT7
      DO 50 I=3,NLINES                                                    PRT7
      ILOW = NX + 1                                                       PRT7
      IHIGH = I * 6                                                       PRT7
      WRITE(6,150)(TITLE(JW),JW = ILOW,IHIGH)                             PRT7
  150 FORMAT(18X,6A6)                                                     PRT7
      LADD = LADD + 1                                                     PRT7
      NX = IHIGH                                                          PRT7
   50 CONTINUE                                                            PRT7
      RETURN                                                              PRT7
      END                                                                 PRT7
```

SUBROUTINE RELALP

```
      C ***********************************************************************RLLP    1
      C *  SUBROUTINE RELALP -THIS SUBROUTINE CONVERTS REAL NUMBERS TO        *RLLP    2
      C * ALPHANUMERIC NUMBERS                                                *RLLP    3
      C ***********************************************************************RLLP    4
      C                                                                       RLLP    5
            SUBROUTINE RELALP(IRES,NFAL,K)                                    RLLP    6
            DIMENSION IX(6)                                                   RLLP    7
            I=REAL                                                            RLLP    8
            DO 1 J=1,K                                                        RLLP    9
      C                                                                       RLLP   10
      C*****MOD IS A LIBRARY FUNCTION TO GIVE RESIDUAL NUMBER FROM DIVISION   RLLP   11
      C                                                                       RLLP   12
            IX(J)=MOD(I,10)                                                   RLLP   13
            I=I/10                                                            RLLP   14
    1       CONTINUE                                                          RLLP   15
            IRES=0                                                            RLLP   16
            ICHK=0                                                            RLLP   17
            DO 2 J=K,1,-1                                                     RLLP   18
            IF(ICHK.NE.0.OR.J.EQ.1)GO TO 3                                    RLLP   19
            IF(IX(J).NE.0) GO TO 4                                            RLLP   20
            IX(J)=5                                                           RLLP   21
            GO TO 5                                                           RLLP   22
    4       ICHK=1                                                            RLLP   23
    3       IX(J)=IX(J)+48                                                    RLLP   24
    5       IRES=IRES*64+IX(J)                                                RLLP   25
    2       CONTINUE                                                          RLLP   26
            IF(K.EQ.6) GO TO 10                                               RLLP   27
            IRES=(IRES*64+5)*64+5                                             RLLP   28
   10       RETURN                                                            RLLP   29
            END                                                               RLLP   30
```

SUBROUTINE SORT1

```
      C ********.********.**********************************************************SRT1    1
      C *  SUBROUTINE SORT1 -THIS ROUTINE DOES ALL THE SORTING OF THE GSE TO *SRT1    2
      C * BE FOUND IN STAGE SORT 1                                           *SRT1    3
      C *                                                                    *SRT1    4
      C *  NOTE: THIS ROUTINE, WHEN CALLED FROM SUBROUTINE SORT2, PERFORMS   *SRT1    5
      C * THE SAME SORT ON GSE TO BE FOUND IN STAGE SORT2                    *SRT1    6
      C ********.********.**********************************************************SRT1    7
      C                                                                       SRT1    8
      C              VARIABLE                DEFINITION                       SRT1    9
      C              ***.****                **********                       SRT1   10
      C                                                                       SRT1   11
      C              JKEY          IF THIS LOGICAL VARIABLE IS SET EQUAL TO  SRT1   12
      C                             '.TRUE.' SUBROUTINE PRINT2 IS CALLED     SRT1   13
      C                                                                       SRT1   14
      C                           IF '.FALSE.' SUBROUTINE PRINT1 IS CALLED    SRT1   15
      C                                                                       SRT1   16
      C              IEND         DETERMINES WHEN THE PRINT ROUTINES ARE TO   SRT1   17
      C                           WRITE BOTH FOOTNOTES ON THE LAST PAGE OF    SRT1   18
      C                           THE SORT                                    SRT1   19
      C                                                                       SRT1   20
            COMPILER(DATA=SHORT)                                              SRT1   21
            SUBROUTINE SORT1                                                  SRT1   22
            INCLUDE XSTAGE.LIST                                               SRT1   23
            INCLUDE PROGRM.LIST                                               SRT1   24
            INCLUDE COMON                                                     SRT1   25
            INCLUDE EQUIV                                                     SRT1   26
            LOGICAL JKEY                                                      SRT1   27
            COMMON/TAG/IEND,JKEY,PKEY                                         SRT1   28
      C                                                                       SRT1   29
      C*****DEFINE THE RECORD FILE SIZE, SPECIFY KEY FIELDS, AND INITIALIZE   SRT1   30
      C*****THE SORT/MERGE PACKAGE                                            SRT1   31
      C                                                                       SRT1   32
            CALL RSZ#(1,WORDS)                                                SRT1   33
            CALL KEY#(1,35,36,'B','A',1)                                      SRT1   34
            CALL KEY#(0,35,36,'A','A',2)                                      SRT1   35
            CALL KEY#(2,35,108,'A','A',3)                                     SRT1   36
            CALL SRTOPN                                                       SRT1   37
            REWIND 4                                                          SRT1   38
            IEND=0                                                            SRT1   39
            NPAGE=0                                                           SRT1   40
            LADD=0                                                            SRT1   41
      75    READ(    4,ERR=92,END=86)WORDS                                    SRT1   42
      C                                                                       SRT1   43
      C*****RELEASE THE WORDS TO BE SORTED ON THE RECORD FILE                 SRT1   44
      C                                                                       SRT1   45
            CALL SRTREL(WORDS)                                                SRT1   46
            GO TO 75                                                          SRT1   47
      C                                                                       SRT1   48
      C*****START SORT OF DATA FILE                                           SRT1   49
      C                                                                       SRT1   50
       86   CALL SRTSRT                                                       SRT1   51
       87   CONTINUE                                                          SRT1   52
      C                                                                       SRT1   53
```

SUBROUTINE SORT1 (Concluded)

```
54      C*****RETRIEVE THE SORTED WORDS FROM THE RECORD FILE                    SRT1  54
55      C                                                                       SRT1  55
56            CALL SRTRET(WORDS,$89)                                            SRT1  56
57            DO 10 I=1,NSTAGE                                                  SRT1  57
58            KEEPX=(I*4)-3                                                     SRT1  58
59            IF(MOD.EW.GSETLE(KEEPX))MOD=I                                     SRT1  59
60            IF(MOD.EW.I) GO TO 11                                             SRT1  60
61         10 CONTINUE                                                          SRT1  61
62            WRITE(6,100)MOD,EINUM                                             SRT1  62
63        100 FORMAT(1HI,10X,'***ERROR***THE PROCEDURE TABLE HAS BEEN SEARCHED A SRT1  63
64           1NO STAGENAME:',2X,A6,2X,'EI NO.:',2X,A6,A1,2X,'COULD NOT BE FOUND'SRT1  64
65           2)                                                                 SRT1  65
66            GO TO 87                                                          SRT1  66
67         11 CONTINUE                                                          SRT1  67
68            IF(NPRO.LE.0.OR.NPRO.GE. 6) WRITE(6,120)                          SRT1  68
69        120 FORMAT(/,10X,'***ERROR*** PROJECT CODE NO. IS IN ERROR AND THEREBYSRT1  69
70           *CANNOT BE ACCQUIRED FROM THE PROCEDURE TABLE')                    SRT1  70
71            IF(NPRO.LE.0.OR.NPRO.GE. 6) NPRO=6                                SRT1  71
72            JUMP=NPRO*2                                                       SRT1  72
73            INDEX=JUMP-1                                                      SRT1  73
74            PROGM(1)= AGENDU(INDEX)                                           SRT1  74
75            PROGM(2)= AGENDU(JUMP)                                            SRT1  75
76            IF(JKEY) GO TO 90                                                 SRT1  76
77            CALL PRINT1                                                       SRT1  77
78            GO TO 87                                                          SRT1  78
79         90 CALL PRINT2                                                       SRT1  79
80            GO TO 87                                                          SRT1  80
81         89 IF(JKEY) GO TO 95                                                 SRT1  81
82            IEND=1                                                            SRT1  82
83            CALL HEADR1                                                       SRT1  83
84            WRITE(6,110)                                                      SRT1  84
85        110 FORMAT(1HI,9X,'*** COMPLETION OF STAGE SORT 1 ***')               SRT1  85
86            GO TO 99                                                          SRT1  86
87         95 IEND=1                                                            SRT1  87
88            CALL HEADR2                                                       SRT1  88
89            WRITE(6,111)                                                      SRT1  89
90        111 FORMAT(1HI,9X,'*** COMPLETION OF STAGE SORT 2 ***')               SRT1  90
91            GO TO 99                                                          SRT1  91
92         92 WRITE(6,109)                                                      SRT1  92
93        109 FORMAT(1HI,10X,'***ERROR*** ERROR WHILE READING DATA BASE FILE 4')SRT1  93
94         99 RETURN                                                            SRT1  94
95            END                                                               SRT1  95
```

SUBROUTINE SORT2

```
      C ***********************************************************************SRT2    1
      C *  SUBROUTINE SORT2 -THIS ROUTINE SETS THE VARAIBLE KEY FOR PERFORM- *SRT2    2
      C * ING STAGE SORT 2                                                    *SRT2    3
      C ***********************************************************************SRT2    4
      C                                                                        SRT2    5
      C              VARIABLE                DEFINITION                        SRT2    6
      C              ********                **********                        SRT2    7
      C              JKEY          WHEN SET EQUAL TO '.TRUE.' A STAGE SORT 2SRT2    8
      C                            CAN THEN BE PERFORMED                     SRT2    9
      C                                                                      SRT2   10
            SUBROUTINE SORT2                                                 SRT2   11
            LOGICAL JKEY                                                     SRT2   12
            COMMON/TAG/IEND,JKEY,PKEY                                        SRT2   13
            JKEY=.TRUE.                                                      SRT2   14
            CALL SORT1                                                       SRT2   15
            RETURN                                                           SRT2   16
            END                                                              SRT2   17
```

SUBROUTINE SORT3

```
      C ********************************************************************SRT3    1
      C *  SUBROUTINE SORT3 -THIS ROUTINE DOES ALL THE SORTING OF THE GSE TO *SRT3    2
      C * BE FOUND IN THE FLUID MEDIA SORT                                   *SRT3    3
      C ********************************************************************SRT3    4
            COMPILER(DATA=SHORT)                                             SRT3    5
            SUBROUTINE SORT3                                                 SRT3    6
            INCLUDE CODEST                                                   SRT3    7
            INCLUDE COMON                                                    SRT3    8
            INCLUDE EQUIV                                                    SRT3    9
            INCLUDE XMETA8,LIST                                              SRT3   10
            COMMON/THREE/MESAVE                                              SRT3   11
            LADD = 0                                                         SRT3   12
            NPAGE = 0                                                        SRT3   13
            NSET=0                                                           SRT3   14
            CHMED = DESCOD(NUMBER)                                           SRT3   15
            DO 10  I = 1,MECODS                                              SRT3   16
            MESAVE = (I*7)-6                                                 SRT3   17
            IF(CHMED.EQ.TABMED(MESAVE)) GO TO 30                             SRT3   18
         10 CONTINUE                                                         SRT3   19
            WRITE(6,100) CHMED                                               SRT3   20
        100 FORMAT(//10X,'***DESCRIPTER NAME:'2X,A6,2X,'IN ERROR***')        SRT3   21
            RETURN                                                           SRT3   22
         30 CONTINUE                                                         SRT3   23
            XLET1 = TABMED(MESAVE  + 1)                                      SRT3   24
            XLET2 = TABMED(MESAVE  + 2)                                      SRT3   25
            XLET3 = TABMED(MESAVE+3)                                         SRT3   26
            CALL RSZW(IWORDS)                                                SRT3   27
            CALL KEYW(2,35,108,'A','A',1)                                    SRT3   28
            CALL SRTOPN                                                      SRT3   29
            REWIND 4                                                         SRT3   30
         40 READ(4,END=50)WORDS                                              SRT3   31
            CALL SRTREL(WORDS)                                               SRT3   32
            GO TO 40                                                         SRT3   33
         50 CALL SRTSRT                                                      SRT3   34
         52 CALL SRTRET(WORDS,$70)                                           SRT3   35
      C                                                                      SRT3   36
      C*****INITIALIZE                                                       SRT3   37
      C                                                                      SRT3   38
            I = 0                                                            SRT3   39
          1 CONTINUE                                                         SRT3   40
            I = I + 1                                                        SRT3   41
      C                                                                      SRT3   42
      C*****CHECK REMAINING FIELD SIZE                                       SRT3   43
      C                                                                      SRT3   44
          2 CONTINUE                                                         SRT3   45
            IF(I.EQ.12) GO TO 52                                             SRT3   46
            IF(XLET1-MEDIA(I)) 1,6,1                                         SRT3   47
          6 CONTINUE                                                         SRT3   48
            I = I +1                                                         SRT3   49
            IF(I.EQ.12) GO TO 52                                             SRT3   50
      C                                                                      SRT3   51
      C*****CHECK NEXT LETTER                                                SRT3   52
      C                                                                      SRT3   53
```

SUBROUTINE SORT3 (Concluded)

```
54            IF(XLET2-MEDIA(I)) 2,11,2                                    SRT3  54
55         11 CONTINUE                                                     SRT3  55
56            I=I+1                                                        SRT3  56
57            IF(I.EQ.12) GO TO 52                                         SRT3  57
58         C                                                               SRT3  58
59         C*****CHECK THIRD LETTER                                        SRT3  59
60         C                                                               SRT3  60
61            IF (XLET3-MEDIA(I)) 2,12,2                                   SRT3  61
62         12 CONTINUE                                                     SRT3  62
63            NSET=1                                                       SRT3  63
64            CALL PRINT3                                                  SRT3  64
65            GO TO 52                                                     SRT3  65
66         C                                                               SRT3  66
67         C                                                               SRT3  67
68         C                                                               SRT3  68
69         C*****HAVE SEARCHED DATA-THE SORT IS NOW COMPLETE               SRT3  69
70         C                                                               SRT3  70
71         70 CONTINUE                                                     SRT3  71
72            IF(NSET.EQ.0)WRITE(6,36)CHMED                                SRT3  72
73          36   FORMAT(//10X,'***MASTER FILE CONTAINED NO DATA WITH ',A6,' FLUID MSRT3  73
74              1EDIA SORT***')                                           SRT3  74
75            WRITE(6,37)                                                  SRT3  75
76         37 FORMAT(1H1,9X,'*** COMPLETION OF MEDIA SORT ***')            SRT3  76
77            RETURN                                                       SRT3  77
78            END                                                          SRT3  78
```

SUBROUTINE SORT4

```
      C ********.********.**************************************************SRT4    1
      C *  SUBROUTINE SORT4 -THIS ROUTINE DOES ALL THE SORTING OF THE GSE TO *SRT4    2
      C * BE FOUND IN THE FUNCTIONAL CODE SORT                               *SRT4    3
      C ********.********.**************************************************SRT4    4
            COMPILER(DATA=SHORT)                                             SRT4    5
            SUBROUTINE SORT4                                                 SRT4    6
            INCLUDE COMON                                                    SRT4    7
            INCLUDE EQUIV                                                    SRT4    8
            INCLUDE CODEST                                                   SRT4    9
            INCLUDE FUNTAB,LIST                                              SRT4   10
            COMMON/FOUR/ISAVE                                                SRT4   11
            NPAGE = 0                                                        SRT4   12
            LADD = 0                                                         SRT4   13
            NSET=0                                                           SRT4   14
            FUNCK = DESCOD(NUMBER)                                           SRT4   15
            DO 10  I = 1,NUCODS                                              SRT4   16
            ISAVE = (I*5)-4                                                  SRT4   17
            IF(FUNCK.EQ.FUCOTA(ISAVE)) GO TO 30                              SRT4   18
         10 CONTINUE                                                         SRT4   19
            WRITE(6,100) FUNCK                                               SRT4   20
        100 FORMAT(//10X,'***DESCRIPTER NAME:',2X,A6,2X,'IN ERROR***')       SRT4   21
            RETURN                                                           SRT4   22
         30 CONTINUE                                                         SRT4   23
            CALL HSZW(IWORDS)                                                SRT4   24
            CALL KEYW(23,35,36,'A','A',1)                                    SRT4   25
            CALL KEYW(2,35,108,'A','A',2)                                    SRT4   26
            CALL SRTOPN                                                      SRT4   27
            REWIND 4                                                         SRT4   28
         40 READ(4,END = 50) WORDS                                           SRT4   29
            CALL SRTREL(WORDS)                                               SRT4   30
            GO TO 40                                                         SRT4   31
         50 CALL SRTSRT                                                      SRT4   32
         52 CALL SRTRET(WORDS,$70)                                           SRT4   33
            IF(FUNCK.EQ.FCTCDE) GO TO 60                                     SRT4   34
            GO TO 52                                                         SRT4   35
         60 CONTINUE                                                         SRT4   36
            NSET=1                                                           SRT4   37
            CALL PRINT4                                                      SRT4   38
            GO TO 52                                                         SRT4   39
         70 CONTINUE                                                         SRT4   40
            IF(NSET.EQ.0)WRITE(6,79)FUNCK                                    SRT4   41
         79   FORMAT(//10X,'***MASTER FILE CONTAINED NO DATA WITH ',A6,' FUNCTIOSRT4   42
           1NAL CODE SORT***')                                               SRT4   43
            WRITE(6,80)                                                      SRT4   44
         80 FORMAT(1H1,9X,'*** COMPLETION OF FUNCTIONAL CODE SORT ***')      SRT4   45
            RETURN                                                           SRT4   46
            END                                                              SRT4   47
```

SUBROUTINE SORT5

```
      C ********************************************************************SRTS    1
      C *  SUBROUTINE SORT5 -THIS ROUTINE DOES ALL THE SORTING OF THE GSE TO *SRTS    2
      C * BE FOUND IN THE CLASS CODE SORT                                    *SRTS    3
      C ********************************************************************SRTS    4
            COMPILER(DATA=SHORT)                                             SRTS    5
            SUBROUTINE SORT5                                                 SRTS    6
            INCLUDE COMON                                                    SRTS    7
            INCLUDE EQUIV                                                    SRTS    8
            INCLUDE CODEST                                                   SRTS    9
            INCLUDE CLASCD,LIST                                              SRTS   10
            COMMON/FIVE/IHAVIT                                               SRTS   11
            LADD = 0                                                         SRTS   12
            NPAGE = 0                                                        SRTS   13
            NSET=0                                                           SRTS   14
            DO 10   I = 1, JUCODS                                            SRTS   15
            IHAVIT = (I*4)-3                                                 SRTS   17
            IF(DESCOD(NUMBER).EQ.CLACOD(IHAVIT)) GO TO 30                    SRTS   18
         10 CONTINUE                                                         SRTS   19
            WRITE(6,100)DESCOD(NUMBER)                                       SRTS   20
        100 FORMAT(//1OX,'***DESCRIPTER NAME',2X,A6,2X,'IN ERROR***')        SRTS   21
            RETURN                                                           SRTS   22
         30 CONTINUE                                                         SRTS   23
            CALL RSZW(IWORDS)                                                SRTS   24
            CALL KEYW(5,35,36,'B','A',1)                                     SRTS   25
            CALL KEYW(2,35,108,'A','A',2)                                    SRTS   26
            CALL SRTOPN                                                      SRTS   27
            REWIND 4                                                         SRTS   28
         40 READ(4,END=50)WORDS                                              SRTS   29
            CALL SRTREL(WORDS)                                               SRTS   30
            GO TO 40                                                         SRTS   31
         50 CALL SRTSRT                                                      SRTS   32
         52 CALL SRTRET(WORDS,$70)                                           SRTS   33
            IF(ICODE.EQ.NCODE) GO TO 60                                      SRTS   34
            GO TO 52                                                         SRTS   35
         60 CONTINUE                                                         SRTS   36
            NSET=1                                                           SRTS   37
            CALL PRINTS                                                      SRTS   38
            GO TO 52                                                         SRTS   39
         70 CONTINUE                                                         SRTS   40
            IF(NSET.EQ.0)WRITE(6,38)ICODE                                    SRTS   41
         38   FORMAT(//10X,'***MASTER FILE CONTAINED NO DATA WITH CLASS CODE',I2SRTS   42
           1,'***')                                                          SRTS   43
            WRITE(6,37)                                                      SRTS   44
         37 FORMAT(1H1,9X,'*** COMPLETION OF CLASS CODE SORT ***')           SRTS   45
            RETURN                                                           SRTS   46
            END                                                              SRTS   47
```

SUBROUTINE SORT6

```
      C ***********************************************************************SRT6    1
      C * SUBROUTINE TO FIND THE GSE INFORMATION ASSOCIATED WITH THE           *SRT6    2
      C *  APPLICABLE DOCUMENT SORT                                            *SRT6    3
      C ***********************************************************************SRT6    4
      C                                                                         SRT6    5
      C              VARIABLE                DEFINITION                       SRT6    6
      C              ********                **********                       SRT6    7
      C                                                                        SRT6    8
      C              IERR          DETERMINES IF THE SPECIFIED APPLICABLE     SRT6    9
      C                            DOCUMENT WAS FOUND                         SRT6   10
      C                                                                       SRT6   11
            COMPILER(DATA=SHORT)                                              SRT6   12
            SUBROUTINE SORT6                                                  SRT6   13
            INCLUDE COMON                                                     SRT6   14
            INCLUDE EQUIV                                                     SRT6   15
            INCLUDE CODEST                                                    SRT6   16
            COMMON/SIX/ IFHR6                                                 SRT6   17
            DATA BLANK/'      '/                                              SRT6   18
            LADD = 0                                                          SRT6   19
            NPAGE = 0                                                         SRT6   20
            IFHR6 = 0                                                         SRT6   21
      C                                                                       SRT6   22
      C*****IERR IS VARIABLE WHICH DETERMINES IF SPECIFIED APPLICABLE          SRT6   23
      C***** DOCUMENT WAS FOUND                                               SRT6   24
      C                                                                       SRT6   25
            IERR = 0                                                          SRT6   26
            REWIND 4                                                          SRT6   27
      C                                                                       SRT6   28
      C*****READ TAPE DATA FROM FILE 4                                        SRT6   29
      C                                                                       SRT6   30
       1    READ(4,END=98)WORDS                                              SRT6   31
            I=1                                                               SRT6   32
      C                                                                       SRT6   33
      C*****DETERMINE WHICH GSE EQUIPMENT IS ASSOCIATED WITH SPECIFIED        SRT6   34
      C***** APPLICABLE DOCUMENT                                              SRT6   35
      C                                                                       SRT6   36
       9    IF(APDOC(I)-BLANK)10,1,10                                        SRT6   37
       10   CONTINUE                                                         SRT6   38
            IF(APDOC(I)-DESCOD(NUMBER)) 11,12,11                             SRT6   39
       11   I=I+2                                                            SRT6   40
            GO TO 9                                                          SRT6   41
       12   J=I+1                                                            SRT6   42
            IF(APDOC(J)-DESCO2(NUMBER)) 13,14,13                             SRT6   43
       13   CONTINUE                                                         SRT6   44
            I=J+1                                                            SRT6   45
            GO TO 9                                                          SRT6   46
      C                                                                       SRT6   47
      C*****CALL SUBROUTINE TO PRINT INFORMATION ASSOCIATED WITH SPECIFIED     SRT6   48
      C***** APPLICABLE DOCUMENT                                              SRT6   49
      C                                                                       SRT6   50
       14   CALL PRINT6                                                      SRT6   51
            IERR = IERR + 1                                                  SRT6   52
            GO TO 1                                                          SRT6   53
```

SUBROUTINE SORT6 (Concluded)

```
      98 CONTINUE                                                        SRT6  54
C                                                                        SRT6  55
C*****IF IERR VALUE IS ZERO,APPLICABLE DOCUMENT WAS NOT FOUND            SRT6  56
C*****IF IERR VALUE IS GREATER THAN ZERO,APPLICABLE DOCUMENT WAS FOUND   SRT6  57
C                                                                        SRT6  58
      IF(IERR-1) 20,21,21                                                SRT6  59
C                                                                        SRT6  60
C*****PRINT ERROR - DOCUMENT SPECIFIED COULD NOT BE FOUND                SRT6  61
C                                                                        SRT6  62
   20 WRITE(6,601) DESCOD(NUMBER),DESCO2(NUMBER)                         SRT6  63
  601 FORMAT(1HI,10X,'***ERROR***DOCUMENT ',2A6,' COULD NOT BE FOUND***'SRT6  64
     1)                                                                  SRT6  65
C                                                                        SRT6  66
C*****RETURN TO SUBROUTINE MERGE                                         SRT6  67
C                                                                        SRT6  68
      RETURN                                                             SRT6  69
   21 CONTINUE                                                           SRT6  70
C                                                                        SRT6  71
C*****PRINT FOOTNOTES FOR LAST PAGE                                      SRT6  72
C                                                                        SRT6  73
      WRITE(6,602)                                                       SRT6  74
  602 FORMAT(//1X,'N/A OR NA - NOT APPLICABLE')                          SRT6  75
      WRITE(6,806)                                                       SRT6  76
  806 FORMAT(1X,'CLASS CODES:   1.STAGE SYSTEMS   2.PAYLOAD MODULE   ',  SRT6  77
     1'3.EXPERIMENT')                                                    SRT6  78
C                                                                        SRT6  79
C*****PRINT MESSAGE THAT DOCUMENT SORT IS COMPLETED                      SRT6  80
C                                                                        SRT6  81
      WRITE(6,600) DESCOD(NUMBER),DESCO2(NUMBER)                         SRT6  82
  600 FORMAT(1HI,10X,'***COMPLETION OF APPLICABLE DOCUMENT ',*           SRT6  83
     12A6,' SORT***')                                                    SRT6  84
C                                                                        SRT6  85
C*****RETURN TO SUBROUTINE MERGE                                         SRT6  86
C                                                                        SRT6  87
      RETURN                                                             SRT6  88
      END                                                                SRT6  89
```

SUBROUTINE SORT7

```
C ***********************************************************************SRT7   1
C *  SUBROUTINE SORT7 -THIS ROUTINE DOES ALL THE SORTING OF THE GSE TO *SRT7   2
C * BE FOUND IN THE USE LOCATION SORT                                  *SRT7   3
C ***********************************************************************SRT7   4
      COMPILER(DATA=SHORT)                                              SRT7   5
      SUBROUTINE SORT7                                                  SRT7   6
      INCLUDE COMON                                                     SRT7   7
      INCLUDE EQUIV                                                     SRT7   8
      INCLUDE CODEST                                                    SRT7   9
      INCLUDE USLOCT,LIST                                               SRT7  10
      COMMON/SEVEN/KESAVE                                               SRT7  11
      LADD = 0                                                          SRT7  12
      NPAGE = 0                                                         SRT7  13
      NSET=0                                                            SRT7  14
      CHLOC = DESCOD(NUMBER)                                            SRT7  15
      DO 10  I = 1,LOCOD                                                SRT7  16
      KESAVE =(I*7)-6                                                   SRT7  17
      IF(CHLOC.EQ.PLTAB(KESAVE)) GO TO 30                               SRT7  18
   10 CONTINUE                                                          SRT7  19
      WRITE(6,100) CHLOC                                                SRT7  20
  100 FORMAT(//10X,'***DESCRIPTER NAME:'2X,A6,2X,'IN ERROR***')         SRT7  21
      RETURN                                                            SRT7  22
   30 CONTINUE                                                          SRT7  23
      CALL RSZW(IWORDS)                                                 SRT7  24
      CALL KEYW(2,35,108,'A','A',1)                                     SRT7  25
      CALL SRTOPN                                                       SRT7  26
      REWIND 4                                                          SRT7  27
   40 READ(4,END=50)WORDS                                               SRT7  28
      CALL SRTREL(WORDS)                                                SRT7  29
      GO TO 40                                                          SRT7  30
   50 CALL SRTSRT                                                       SRT7  31
   52 CALL SRTRET(WORDS,$70)                                            SRT7  32
      USCK1 = PLTAB(KESAVE+1)                                           SRT7  33
      USCK2 = PLTAB(KESAVE+2)                                           SRT7  34
      I=0                                                               SRT7  35
    1 CONTINUE                                                          SRT7  36
      I = I +1                                                          SRT7  37
    2 CONTINUE                                                          SRT7  38
      IF(I.EQ.18) GO TO 20                                              SRT7  39
      IF(USCK1-USLOC(I))1,6,1                                           SRT7  40
    6 CONTINUE                                                          SRT7  41
      I = I + 1                                                         SRT7  42
      IF(USCK2 - USLOC(I))2,11,2                                        SRT7  43
   11 CONTINUE                                                          SRT7  44
      NSET=1                                                            SRT7  45
      CALL PRINT7                                                       SRT7  46
      GO TO 52                                                          SRT7  47
   20 CONTINUE                                                          SRT7  48
      GO TO 52                                                          SRT7  49
   70 CONTINUE                                                          SRT7  50
      IF(NSET.EQ.0)WRITE(6,36)CHLOC                                     SRT7  51
 36   FORMAT(//10X,'***MASTER DATA FILE CONTAINED NO DATA WITH ',A6,' USSRT7  52
     1E LOCATION SORT***')                                              SRT7  53
      WRITE(6,37)                                                       SRT7  54
   37 FORMAT(1H1,9X,'*** COMPLETION OF USE LOCATION SORT ***')          SRT7  55
      RETURN                                                            SRT7  56
      END                                                               SRT7  57
```

PROCEDURE TABLE

```
      MAIN PROC                                                         TBLE   1
C ***********************************************************************TBLE   2
C *   SORT CODE MNEMONIC PROCEDURE TABLE                               *TBLE   3
C *                                                                    *TBLE   4
C *   THE ARRAY ''XMOD'' CONTAINS THE STANDARD SORT CODE MNEMONICS     *TBLE   5
C *                                                                    *TBLE   6
C *             MNEMONIC                TYPE OF SORT                   *TBLE   7
C *             ********                ************                   *TBLE   8
C *                                                                    *TBLE   9
C *             STAGE1              STAGE SORT 1                       *TBLE  10
C *             STAGE2              STAGE SORT 2                       *TBLE  11
C *             MEDIAS              FLUID MEDIA SORT                   *TBLE  12
C *             FUNCTL              FUNCTIONAL CODE SORT               *TBLE  13
C *             CLASSC              CLASS CODE SORT                    *TBLE  14
C *             APPDOC              APPLICABLE DOCUMENT SORT           *TBLE  15
C *             USELOC              USE LOCATION SORT                  *TBLE  16
C *             SUMARY              MASTER SUMMARY TABLE               *TBLE  17
C *                                                                    *TBLE  18
C *                 ROUTINES THAT USE MAIN PROC                        *TBLE  19
C *                 ***************************                        *TBLE  20
C *                         MAINLINE                                   *TBLE  21
C *                                                                    *TBLE  22
C ***********************************************************************TBLE  23
      INTEGER XMOD(9)                                                   TBLE  24
      DATA XMOD/'STAGE1','STAGE2','MEDIAS','FUNCTL','CLASSC','APPDOC',  TBLE  25
     * 'USELOC','SUMARY','      '/                                      TBLE  26
      END                                                               TBLE  27
      CODEST PROC                                                       TBLE  28
C ***********************************************************************TBLE  29
C *   SORT CODE ARRAYS AND DESCRIPTER NAME PROCEDURE TABLE             *TBLE  30
C *                                                                    *TBLE  31
C *             VARIABLE                DEFINITION                     *TBLE  32
C *             ********                **********                     *TBLE  33
C *                                                                    *TBLE  34
C *             MAXSRT        SETS THE MAXIMUM NUMBER OF SORTS TO BE   *TBLE  35
C *                           PERFORMED PER RUN                        *TBLE  36
C *                                                                    *TBLE  37
C *             CDESRT        THIS ARRAY CONTAINS THE SORT CODE        *TBLE  38
C *                           MNEMONICS IN THE ORDER SPECIFIED BY THE  *TBLE  39
C *                            USER                                    *TBLE  40
C *                                                                    *TBLE  41
C *             NARGU         CONTAINS THE SORT CODE MNEMONIC FOR THE  *TBLE  42
C *                           CURRENT SORT                             *TBLE  43
C *                                                                    *TBLE  44
C *             DESCOD        CONTAINS THE FIRST WORD OF THE DESCRIPTER *TBLE  45
C *                           NAME FOUND ON THE SORT CODE MNEMONIC DATA *TBLE  46
C *                            CARD                                    *TBLE  47
C *                                                                    *TBLE  48
C *             DESCO2        CONTAINS THE SECOND WORD OF THE          *TBLE  49
C *                           DESCRIPTER NAME                          *TBLE  50
C *                                                                    *TBLE  51
C *             NUMBER        IS THE INTEGER CODE FOR THE CURRENT SORT *TBLE  52
C *                                                                    *TBLE  53
C *             NUMSRT        ACTUAL NUMBER OF SORTS THE USER REQUESTS *TBLE  54
C *                                                                    *TBLE  55
C *                 ROUTINES THAT USE CODEST PROC                      *TBLE  56
C *                 *****************************                      *TBLE  57
C *                         HEADS                                      *TBLE  58
C *                         HEADR6                                     *TBLE  59
```

PROCEDURE TABLE (Continued)

```
0060      C *                          MAINLINE                                    *TBLE  60
0061      C *                          MERGE                                       *TBLE  61
0062      C *                          SORT3                                       *TBLE  62
0063      C *                          SORT4                                       *TBLE  63
0064      C *                          SORT5                                       *TBLE  64
0065      C *                          SORT6                                       *TBLE  65
0066      C *                          SORT7                                       *TBLE  66
0067      C *                                                                      *TBLE  67
0068      C ***************************************************************************TBLE  68
0069            PARAMETER MAXSRT = 8                                                TBLE  69
0070            INTEGER CDESRT                                                      TBLE  70
0071            COMMON/CODE/CDESRT(MAXSRT),NARGU,DESCOD(MAXSRT),NUMBER,NUMSRT       TBLE  71
0072            COMMON/CODE/DESCO2(MAXSRT)                                          TBLE  72
0073       END                                                                      TBLE  73
PE0074    XSTAGE PROC                                                             TBLE  74
0075      C ***************************************************************************TBLE  75
0076      C *   STAGENAME PROCEDURE TABLE                                          *TBLE  76
0077      C *                                                                      *TBLE  77
0078      C *   THE ARRAY ''GSETLE'' CONTAINS THE VARIOUS GSE STAGENAME TITLES     *TBLE  78
0079      C *                                                                      *TBLE  79
0080      C *            VARIABLE                 DEFINITION                      *TBLE  80
0081      C *            ********                 **********                      *TBLE  81
0082      C *                                                                      *TBLE  82
0083      C *            NSTAGE        THE NUMBER OF STAGENAMES APPEARING IN     *TBLE  83
0084      C *                          THIS TABLE + 1                            *TBLE  84
0085      C *                                                                      *TBLE  85
0086      C *                ROUTINES THAT USE XSTAGE PROC                         *TBLE  86
0087      C *                ****************************                      *TBLE  87
0088      C *                        MASTER                                        *TBLE  88
0089      C *                        HEADR1                                        *TBLE  89
0090      C *                        HEADR2                                        *TBLE  90
0091      C *                        SORT1                                         *TBLE  91
0092      C *                        PRINT1                                        *TBLE  92
0093      C *                                                                      *TBLE  93
0094      C ***************************************************************************TBLE  94
0095            INTEGER GSETLE(100)                                                 TBLE  95
0096            NSTAGE=12                                                           TBLE  96
0097            DATA GSETLE/'      ','**UNKN','OWN** ','      '                     TBLE  97
0098           *           ,'SRB  ','S. R. ','BOOSTE','R    '                     TBLE  98
0099           *           ,'ET   ','EXTERN','AL TAN','K    '                     TBLE  99
0100           *           ,'SSME ','SHUTTL','E MAIN',' ENGN.'                    TBLE 100
0101           *           ,'AM   ',' AIRLO','CK MOD','ULE  '                     TBLE 101
0102           *           ,'OMS  ',' ORBIT','AL WOR','KSHOP '                    TBLE 102
0103           *           ,'ATM  ','APOLLO',' TELE.',' MOUNT'                    TBLE 103
0104           *           ,'MDA  ','     ',' MDA  ','     '                      TBLE 104
0105           *           ,'IU   ',' INSTR','UMENT ','UNIT '                     TBLE 105
0106           *           ,'S-IC ',' S-IC ','STAGE ','     '                     TBLE 106
0107           *           ,'S-IVB','S-IVB ','STAGE ','     '                     TBLE 107
0108           *           ,'S-2  ',' S-II ','STAGE ','     '/                    TBLE 108
0109      C    *           ,'THIS SPACE IS FOR THE CONTINUATION OF GSE DESCRIPTIVE TBLE 109
0110      C    *           ,'                                                     TBLE 110
0111      C    *           ,'                                                     TBLE 111
0112      C    *           ,'                                                     TBLE 112
0113      C    *           ,'                                                     TBLE 113
0114      C    *           ,'                                                     TBLE 114
0115      C    *           ,'                                                     TBLE 115
0116      C    *           ,'                                                     TBLE 116
0117      C    *           ,'THIS SPACE IS FOR THE CONTINUATION OF GSE DESCRIPTIVE TBLE 117
0118       END                                                                    TBLE 118
```

```
XMETAB PROC
C ******************************************************************
C *   FLUID MEDIA PROCEDURE TABLE                                  *
C *                                                                *
C *   THE ARRAY ''TABMED'' CONTAINS THE VARIOUS FLUID MEDIA CODES  *
C *                                                                *
C *            VARIABLE                    DEFINITION              *
C *            ********                    **********              *
C *            MECODS        THE NUMBER OF FLUID MEDIAS APPEARING IN *
C *                           THIS TABLE                           *
C *                                                                *
C *                 ROUTINES THAT USE XMETAB PROC                  *
C *                 *****************************                  *
C *                           HEADR3 AND SORT3                     *
C *                                                                *
C ******************************************************************
      DIMENSION TABMED(252)
      MECODS=22
      DATA(TABMED(K),K=1,126)/
     * 'GN2   ','G','N','2','GASEOU','S NITR','OGEN  ',
     * 'LN2   ','L','N','2','LIQUID',' NITRO','GEN   ',
     * 'ETHGLY','E','T','H','ETHYLE','NE GLY','COL   ',
     * 'RJ-1  ','R','J','-','      ','RJ-1  ','      ',
     * 'H2O   ','H','2','O','      ','WATER ','      ',
     * 'HE    ','H','E',' ','      ','HELIUM','      ',
     * 'VAC   ','V','A','C','      ','VACUUM','      ',
     * 'RP-1  ','R','P','-','      ','RP-1  ','      ',
     * 'HYD   ','H','Y','D','HYDRAU','LIC FL','UID   ',
     * 'ORONIT','O','R','O','      ','ORONIT','E     ',
     * 'H2OGLY','O','G','L','   WAT','ER GLY','COL   ',
     * 'ISOALC','I','S','O','ISOPRO','PYL AL','COHOL ',
     * 'FREON ','F','R','E','      ','FREON ','      ',
     * 'GO2   ','G','O','2','GASEOU','S OXYG','EN    ',
     * 'ME/H2O','M','E','/','METHAN','OL WAT','ER    ',
     * 'MMH   ','M','M','H','MONOME','THYLHY','DRAZNE',
     * 'N2O4  ','N','2','O','NITROG','ENTETR','AOXIDE',
     * 'LH2   ','L','H','2','LIQUID',' HYDRO','GEN   ',/
      DATA(TABMED(K),K=127,154)/
     * 'GH2   ','G','H','2','GASEOU','S HYDR','OGEN  ',
     * 'COOL  ','C','O','O','      ','COOLAN','OL    ',
     * 'TRIC  ','T','R','I','TRICHL','ORETHL','ENE   ',
     * 'PRES  ','P','R','E','PRESER','VATIVE',' OIL  ',/
  END
FUNTAB PROC
C ******************************************************************
C *   FUNCTIONAL CODE PROCEDURE TABLE                              *
C *                                                                *
C *   THE ARRAY ''FUCOTA'' CONTAINS THE VARIOUS FUNCTIONAL CODES   *
C *                                                                *
C *            VARIABLE                    DEFINITION              *
C *            ********                    **********              *
C *                                                                *
C *            NUCODS        THE NUMBER OF FUNCTIONAL CODES APPEARING *
C *                          IN THIS TABLE                         *
C *                                                                *
C *                 ROUTINES THAT USE FUNTAB PROC                  *
C *                 *****************************                  *
C *                           HEADR4 AND SORT4                     *
C *                                                                *
C ******************************************************************
      DIMENSION FUCOTA(50)
```

```
0180            NUCODS = 9                                                      TBLE 180
0181            DATA FUCOTA/'VAC PU','      ','VACUUM',' PUMP ','      ',      TBLE 181
0182           *            'REFRIG','REFRIG','ERATIO','N EQUI','PMENT ',      TBLE 182
0183           *            'SERVIC','SERVIC','ING EQ','UIPMEN','T     ',      TBLE 183
0184           *            'PNEUMA','PNEUMA','TIC EQ','UIPMEN','T     ',      TBLE 184
0185           *            'TESTCH','TEST-C','HECKOU','T EQUI','PMENT ',      TBLE 185
0186           *            'HANDLE','HANDLI','NG EQU','IPMENT','      ',      TBLE 186
0187           *            'ACCESS','      ','   ACC','ESS   ','      ',      TBLE 187
0188           *            'TRANSP','     T','RANSPO','RTATIO','N     ',      TBLE 188
0189           *            'GASDET','GAS DE','TECTOR',' EQUIP','MENT  '/      TBLE 189
0190            END                                                             TBLE 190
PEU191          CLASCD PROC                                                     TBLE 191
0192            C ***********************************************************TBLE 192
0193            C *   CLASS CODE  PROCEDURE TABLE                              *TBLE 193
0194            C *                                                            *TBLE 194
0195            C *   THE ARRAY ''CLACOD'' CONTAINS THE VARIOUS CLASS CODES    *TBLE 195
0196            C *                                                            *TBLE 196
0197            C *            VARIABLE                DEFINITION              *TBLE 197
0198            C *            ********                **********              *TBLE 198
0199            C *                                                            *TBLE 199
0200            C *            JUCODS        THE NUMBER OF CLASS CODES APPEARING IN*TBLE 200
0201            C *                          THIS TABLE                        *TBLE 201
0202            C *                                                            *TBLE 202
0203            C *               ROUTINES THAT USE CLASCD PROC                *TBLE 203
0204            C *               ****************************                 *TBLE 204
0205            C *                       HEADR5 AND SORT5                     *TBLE 205
0206            C *                                                            *TBLE 206
0207            C ***********************************************************TBLE 207
0208                 DIMENSION CLACOD(32)                                       TBLE 208
0209                 JUCODS = 8                                                 TBLE 209
0210                 DATA CLACOD/'1      ','STAGE ','SYSTEM','S     ',        TBLE 210
0211                *            '2     ','PAYLOA','D MODU','LE    ',        TBLE 211
0212                *            '3     ','    EX','PERIME','NT    ',        TBLE 212
0213                *            '4     ','      ',' TEST ','      ',        TBLE 213
0214                *            '5     ','      ','FACTOR','Y     ',        TBLE 214
0215                *            '6     ','GSE TE','ST SUP','PORT  ',        TBLE 215
0216                *            '7     ','FACTOR','Y & TE','ST    ',        TBLE 216
0217                *            '8     ','FAC. T','EST & ','LAUNCH'/        TBLE 217
0218            END                                                             TBLE 218
PEU219          USLOCT PROC                                                     TBLE 219
0220            C ***********************************************************TBLE 220
0221            C *   USE LOCATION PROCEDURE TABLE                             *TBLE 221
0222            C *                                                            *TBLE 222
0223            C *   THE ARRAY ''PLTAB'' CONTAINS THE VARIOUS USE LOCATION NAMES*TBLE 223
0224            C *                                                            *TBLE 224
0225            C *            VARIABLE                DEFINITION              *TBLE 225
0226            C *            ********                **********              *TBLE 226
0227            C *                                                            *TBLE 227
0228            C *            LOCOD         THE NUMBER OF USE LOCATION CODES *TBLE 228
0229            C *                          APPEARING IN THIS TABLE           *TBLE 229
0230            C *                                                            *TBLE 230
0231            C *               ROUTINES THAT USE USLOCT PROC                *TBLE 231
0232            C *               ****************************                 *TBLE 232
0233            C *                       HEADR7 AND SORT7                     *TBLE 233
0234            C *                                                            *TBLE 234
0235            C ***********************************************************TBLE 235
0236                 DIMENSION PLTAB(100)                                       TBLE 236
0237                 LOCOD = 10                                                 TBLE 237
0238                 DATA PLTAB/'KSC   ','K','S','      ','KENNED','Y CENT','ER    ',  TBLE 238
0239                *          'HSFC  ','S','F','    MA','RSHALL',' CENTE','R     ',  TBLE 239
```

```
0240            *              'MTF    ','M','T','MISSIS','SIPPI ','TEST F','ACLITY',  TBLE 240
0241            *              'SDF    ','S','D','SYSTEM',' DEVEL','OP. FA','CILITY',  TBLE 241
0242            *              'IBM    ','I','B','   IBM',' SPACE',' DIVIS','ION   ',  TBLE 242
0243            *              'HB     ','H','B','MDAC-W','D HUNT','INGTON','BEACH ',  TBLE 243
0244            *              'MFG    ','M','F','MANUFA','CTURIN','G FACI','LITY  ',  TBLE 244
0245            *              'CP     ','C','P','      ','CANOGA',' PARK ','      ',  TBLE 245
0246            *              'MAF    ','M','A','MICHOU','D ASSE','MBLY F','AC.   ',  TBLE 246
0247            *              'MSC    ','M','S','      ','HOUSTO','N CENT','ER    '/  TBLE 247
0248             END                                                                     TBLE 248
PE0249          PROGRM PROC                                                             TBLE 249
0250            C ***********************************************************************TBLE 250
0251            C *   PROGRAM NAMES PROCEDURE TABLE                                     *TBLE 251
0252            C *                                                                     *TBLE 252
0253            C *   THE ARRAY 'AGENDU' CONTAINS THE VARIOUS GSE PROGRAM NAMES         *TBLE 253
0254            C *                                                                     *TBLE 254
0255            C *                  ROUTINES THAT USE PROGRM PROC                      *TBLE 255
0256            C *                  ****************************                       *TBLE 256
0257            C *                          MASTER                                     *TBLE 257
0258            C *                          SORT1                                      *TBLE 258
0259            C *                                                                     *TBLE 259
0260            C ***********************************************************************TBLE 260
0261                  DIMENSION AGENDU(12)                                              TBLE 261
0262                  DATA AGENDU/'SATURN',' '                                          TBLE 262
0263                 *           ,'SKYLAB',' '                                          TBLE 263
0264                 *           ,'SHUTTL','E'                                          TBLE 264
0265                 *           ,'  HEAO',' '                                        TBLE 265
0266                 *           ,'SORTIE',' LAB '                                      TBLE 266
0267                 *           ,'UNKNOW','N   '/                                      TBLE 267
0268             END                                                                    TBLE 268
PE0269          EQUIV PROC                                                             TBLE 269
0270            C ***********************************************************************TBLE 270
0271            C *  ** EQUIVALENCE **STATEMENT PROCEDURE TABLE                        *TBLE 271
0272            C *                                                                     *TBLE 272
0273            C *   THE ARRAY **WORDS** IS DIMENSIONED FOR ALL THE GSE ATTRIBUTES     *TBLE 273
0274            C *   THAT PERTAIN TO AN E1 NUMBER ON A RECORD IN THE DATA BASE FILE    *TBLE 274
0275            C *                                                                     *TBLE 275
0276            C * **IWORDS** ESTABLISHES THE NUMBER OF WORDS (GSE ATTRIBUTES) PER     *TBLE 276
0277            C * RECORD TO BE WRITTEN ON THE MASTER DATA BASE FILE                   *TBLE 277
0278            C *                                                                     *TBLE 278
0279            C *                  ROUTINES THAT USE EQUIV PROC                       *TBLE 279
0280            C *                  ****************************                       *TBLE 280
0281            C *                          MASTER                                     *TBLE 281
0282            C *                          UPDATE                                     *TBLE 282
0283            C *                          SORT1                                      *TBLE 283
0284            C *                          SORT3                                      *TBLE 284
0285            C *                          SORT4                                      *TBLE 285
0286            C *                          SORT5                                      *TBLE 286
0287            C *                          SORT6                                      *TBLE 287
0288            C *                          SORT7                                      *TBLE 288
0289            C *                                                                     *TBLE 289
0290            C ***********************************************************************TBLE 290
0291                  PARAMETER IWORDS=268                                              TBLE 291
0292                  DIMENSION WORDS(IWORDS)                                           TBLE 292
0293                  EQUIVALENCE (WORDS(1),NPRO),(WORDS(2),EINUM(1)),(WORDS(5),NCODE),(TBLE 293
0294                 *WORDS(6),MOD),(WORDS(7),QUAN),(WORDS(8),SITLE(1)), (WORDS(80),DESCTBLE 294
0295                 *ON(1 )),(WORDS(82),USLOC(1)),(WORDS(100),APDOC(1)),(WORDS(124),   TBLE 295
0296                 * PROGM(1))          ,(WORDS(126),NLINES),(WORDS(127),DATE(1)),(WORDTBLE 296
0297                 *S(129),MEDIA(1)),(WORDS(141),WGHT),(WORDS(142),LNGTH),(WORDS(143),TBLE 297
0298                 *WIDTH),(WORDS(144),HGHT),(WORDS(145),UNCST(1)),(WORDS(148),FCTCDE)TBLE 298
0299                 * ,(WORDS(149),TITLE(1))                                           TBLE 299
0300             END                                                                   TBLE 300
```

```
      COMON PROC                                                        TBLE 301
C ***********************************************************************TBLE 302
C *   '' COMMON '' STATEMENT PROCEDURE TABLE                            *TBLE 303
C *                                                                     *TBLE 304
C *   THIS TABLE ALLOWS FOR A CONVENIENT AREA IN THE PROGRAM THAT WILL *TBLE 305
C *   ENABLE EACH ROUTINE TO ACCQUIRE THOSE PROGRAM VARIABLES IN        *TBLE 306
C *   ''COMMON'' THAT IT NEEDS                                          *TBLE 307
C *                                                                     *TBLE 308
C *              VARIABLE               DEFINITION                      *TBLE 309
C *              ********               **********                      *TBLE 310
C *                                                                     *TBLE 311
C *              NPRO        PROGRAM CODE                               *TBLE 312
C *              EINUM       END ITEM NUMBER                            *TBLE 313
C *              TITLE       DESCRIPTION                                *TBLE 314
C *              DESCON      DESIGN RESPONSIBILITY                      *TBLE 315
C *              NCODE       CLASS CODE                                 *TBLE 316
C *              QUAN        QUANTITY                                   *TBLE 317
C *              WGHT        WEIGHT                                     *TBLE 318
C *              MOD         STAGENAME                                  *TBLE 319
C *              USLOC       USE LOCATION                               *TBLE 320
C *              FCTCDE      FUNCTIONAL CODE                            *TBLE 321
C *              LNGTH       LENGTH                                     *TBLE 322
C *              WIDTH       WIDTH                                      *TBLE 323
C *              HGHT        HEIGHT                                     *TBLE 324
C *              UNCST       UNIT COST                                  *TBLE 325
C *              MEDIA       FLUID MEDIA                                *TBLE 326
C *              APDOC       APPLICABLE DOCUMENT                        *TBLE 327
C *              SITLE       DUMMY VARIABLE(NOT USED)                   *TBLE 328
C *              DATE        DATE                                       *TBLE 329
C *              PROGM       PROGRAM TITLE                              *TBLE 330
C *              NLINES      THE NUMBER OF LINES OF DESCRIPTION         *TBLE 331
C *              NPAGE       PAGE COUNT                                 *TBLE 332
C *              LADD        LINE COUNTER                               *TBLE 333
C *              DATX        INPUT DATE                                 *TBLE 334
C *                                                                     *TBLE 335
C *                 ROUTINES THAT USE COMON PROC                        *TBLE 336
C *               ********************************                      *TBLE 337
C *               ALL ROUTINES EXCEPT HEADS & MERGE                     *TBLE 338
C *                                                                     *TBLE 339
C ***********************************************************************TBLE 340
      REAL MEDIA                                                        TBLE 341
      INTEGER QUAN,WIDTH,HGHT,WGHT                                      TBLE 342
      COMMON    NPRO,EINUM(3),NCODE,MOD,QUAN,SITLE(72),DESCON(2)        TBLE 343
      COMMON USLOC(18),        APDOC(24),PROGM(2), NLINES,DATE(2)       TBLE 344
      COMMON MEDIA(12),WGHT,LNGTH,WIDTH,HGHT,UNCST(3),FCTCDE            TBLE 345
      COMMON TITLE(120),DATX(2),NPAGE,LADD                              TBLE 346
       END                                                              TBLE 347
```

SUBROUTINE UPDATE

```
C *********************************************************************UPDT   1
C *  SUBROUTINE UPDATE -THIS ROUTINE UPDATES THE MASTER DATA FILE WITH *UPDT   2
C * THE NEW INPUT INFORMATION SUPPLIED BY THE PROGRAM ANALYST          *UPDT   3
C *********************************************************************UPDT   4
      SUBROUTINE UPDATE                                                 UPDT   5
      INCLUDE COMON                                                     UPDT   6
      INCLUDE EQUIV                                                     UPDT   7
      INTEGER R                                                         UPDT   8
      DIMENSION DWORDS(IWORDS)                                          UPDT   9
      DATA NCRD/'$'/                                                    UPDT  10
      DATA NCR1/'1'/                                                    UPDT  11
      DATA NCR2/'2'/                                                    UPDT  12
      DATA NCR4/'4'/                                                    UPDT  13
      DATA NCRA/'*'/                                                    UPDT  14
      DATA BLANK/'                                                      UPDT  15
      R=2                                                               UPDT  16
      NR=2                                                              UPDT  17
      NERROR=0                                                          UPDT  18
      NCARD=0                                                           UPDT  19
      NTOT=0                                                            UPDT  20
      NNDATA=0                                                          UPDT  21
      NDDATA=0                                                          UPDT  22
      NUDATA=0                                                          UPDT  23
      REWIND 1                                                          UPDT  24
      REWIND 2                                                          UPDT  25
      REWIND 3                                                          UPDT  26
      REWIND 4                                                          UPDT  27
C                                                                       UPDT  28
C*****READ TAPE DATA FROM FILE 1                                        UPDT  29
C                                                                       UPDT  30
   10 READ(1,ERR=90,END=30)WORDS                                        UPDT  31
      NOUT=3                                                            UPDT  32
      WRITE(3,ERR=91)WORDS                                              UPDT  33
      GO TO 10                                                          UPDT  34
   30 CONTINUE                                                          UPDT  35
      NLINES=1                                                          UPDT  36
      DO 33 K=7,120                                                     UPDT  37
   33 TITLE(K)=BLANK                                                    UPDT  38
      DO 34 K=13,24                                                     UPDT  39
   34 APDOC(K)=BLANK                                                    UPDT  40
      DATE(1)=BLANK                                                     UPDT  41
      DATE(2)=BLANK                                                     UPDT  42
C                                                                       UPDT  43
C*****READ INPUT DATA FROM CARD FORMAT                                  UPDT  44
C                                                                       UPDT  45
C                                                                       UPDT  46
C*****READ CARD TYPE 1                                                  UPDT  47
C                                                                       UPDT  48
      NCARD=NCARD+1                                                     UPDT  49
   35 READ(NR,200,ERR=95,END=85)NCR,NPRO,EINUM,(TITLE(I),I=1,6)         UPDT  50
  200 FORMAT(I1,2X,I1,2X,2A6,A1,2X,6A6)                                 UPDT  51
      IF(NCR.NE.1)GO TO 95                                              UPDT  52
      IF(NPRO.EQ.0)NR=2                                                 UPDT  53
      IF(NPRO.EQ.0)GO TO 80                                             UPDT  54
   36 NCARD=NCARD+1                                                     UPDT  55
      NFLAG=0                                                           UPDT  56
      READ(R,201,ERR=95,END=85)NCR                                      UPDT  57
  201 FORMAT(A1)                                                        UPDT  58
      IF(NCR.EQ.NCRD)GO TO 40                                           UPDT  59
      IF(NCR.EQ.NCR2)GO TO 45                                           UPDT  60
      IF(NCR.NE.NCR2)GO TO 95                                           UPDT  61
C                                                                       UPDT  62
C*****READ CONTIUATION OF CARD 1                                        UPDT  63
C                                                                       UPDT  64
```

```
      40    NJ=NLINES*6+1                                                     UPDT  65
            MJ=NLINES*6+6                                                     UPDT  66
            IF(NLINES.GT.19)GO TO 98                                          UPDT  67
              READ(0,202,ERR=95,END=85)(TITLE(J),J=NJ,MJ)                     UPDT  68
     202    FORMAT(2IX,6A6)                                                   UPDT  69
           NLINES = NLINES + 1                                                UPDT  70
            GO TO 36                                                          UPDT  71
     C                                                                        UPDT  72
     C*****READ CARD TYPE 2                                                   UPDT  73
     C                                                                        UPDT  74
      45      READ(U,203,ERR=95,END=85)DESCON,NCODE,QUAN,WGHT,MOD             UPDT  75
     203    FORMAT(3X,2A6,2X,I1,2X,A2,2X,A6,2X,A6)                           UPDT  76
     C                                                                        UPDT  77
     C*****READ CARD TYPE 3                                                   UPDT  78
     C                                                                        UPDT  79
            NCARD=NCARD+1                                                     UPDT  80
            READ(R,204,ERR=95,END=85)NCR,USLOC,FCTCDE,LNGTH,WIDTH,HGHT,UNCST, UPDT  81
           *  MEDIA                                                          UPDT  82
     204    FORMAT(I1,2X,18A1,2X,A4,2X,A4,2X,A4,2X,A4,2X,A4,2X,2A6,A1,2X,12A1) UPDT 83
            NCRCK=NCR4                                                        UPDT  84
            IF(NCR.NE.3)GO TO 95                                              UPDT  85
     C                                                                        UPDT  86
     C*****READ CARD TYPE 4                                                   UPDT  87
     C                                                                        UPDT  88
      50    NCARD=NCARD+1                                                     UPDT  89
            NR=2                                                              UPDT  90
            READ(R,201,ERR=82,END=82)NCR                                      UPDT  91
            IF(NCR.EQ.NCR4)GO TO 60                                           UPDT  92
            IF(NCRCK.EQ.NCR4  ) GO TO 95                                      UPDT  93
            IF(NCR.EQ.NCRA)GO TO 70                                           UPDT  94
            IF(NCR.EQ.NCR1)NR=0                                               UPDT  95
            IF(NCR.EQ.NCR1)NCARD=NCARD-1                                      UPDT  96
            IF(NCR.EQ.NCR1)GO TO 80                                           UPDT  97
            GO TO 95                                                          UPDT  98
      60    J=1                                                               UPDT  99
            K=12                                                              UPDT 100
            READ(0,205,ERR=95,END=85)NCR,(APDOC(I),I=J,K)                     UPDT 101
     205    FORMAT(A1,2X,2A6,1X,2A6,1X,2A6,1X,2A6,1X,2A6,1X,2A6)              UPDT 102
            NCRCK=BLANK                                                       UPDT 103
            GO TO 50                                                          UPDT 104
      70    J=13                                                              UPDT 105
            K=24                                                              UPDT 106
            READ(0,206,ERR=95,END=85)NCR,(APDOC(I),I=J,K)                     UPDT 107
            GO TO 80                                                          UPDT 108
     C                                                                        UPDT 109
     C*****ERROR MESSAGES                                                     UPDT 110
     C                                                                        UPDT 111
      90    WRITE(6,100)                                                      UPDT 112
     100    FORMAT(1H1,///' ***ERROR WHILE READING INPUT FILE ! ***',//       UPDT 113
           * ' ***EXECUTION TERMINATED***')                                   UPDT 114
      99    STOP                                                              UPDT 115
      91    WRITE(6,101)NOUT                                                  UPDT 116
     101    FORMAT(1H1,///' ***ERROR WHILE WRITING ON OUTPUT FILE ',I1,       UPDT 117
           * '***',//' ***EXECUTION TERMINATED***')                           UPDT 118
            GO TO 99                                                          UPDT 119
```

```
      95    CONTINUE                                                          UPDT 120
            IF(NFLAG.EQ.1)GO TO 30                                            UPDT 121
            NFLAG=1                                                           UPDT 122
            NR=2                                                              UPDT 123
            IF(NCARD.EQ.1.AND.NCR.EQ.0.AND.EINUM(2).EQ.BLANK)GO TO 30         UPDT 124
            WRITE(6,102)NCARD                                                 UPDT 125
     102    FORMAT(/' ***ERROR WHILE READING INPUT FILE 2***'/' ***CARD # ',  UPDT 126
           * I4,                    ' IS OUT OF ORDER OR IS PUNCHED INCORRECTLUPDT 127
           *Y.***)                                                            UPDT 128
            NERROR = NERROR + 1                                               UPDT 129
            IF(NERROR.GT.30)GO TO 96                                          UPDT 130
            GO TO 30                                                          UPDT 131
     96     WRITE(6,103)                                                      UPDT 132
     103    FORMAT(1HI,///' ***EXCESSIVE INPUT ERRORS, EXECUTION TERMINATED***UPDT 133
           *')                                                                UPDT 134
            GO TO 99                                                          UPDT 135
     97     WRITE(6,105)                                                      UPDT 136
     105    FORMAT(1HI,///' ***ERROR WHILE READING INPUT FILE 3 ***',/        UPDT 137
           * ' ***EXECUTION TERMINATED***')                                   UPDT 138
            GO TO 99                                                          UPDT 139
     98     WRITE(6,121)EINUM                                                 UPDT 140
     121    FORMAT(///' ***MORE THAN 20 LINES OF DESCRIPTION ASSOCIATED WITH  UPDT 141
           *EI NO. '2A6,A1,', EXECUTION CONTINUING***')                       UPDT 142
            GO TO 30                                                          UPDT 143
     C                                                                        UPDT 144
     C******WRITE INPUT DATA ON FILE 3                                        UPDT 145
     C                                                                        UPDT 146
     80     NOUT=3                                                            UPDT 147
            WRITE(3,ERR=91)WORDS                                              UPDT 148
            GO TO 30                                                          UPDT 149
     82     NOUT=3                                                            UPDT 150
            WRITE(3,ERR=91)WORDS                                              UPDT 151
     C                                                                        UPDT 152
     C*****ALL INPUT DATA HAS BEEN READ AND MERGED, ---BEGIN SORTING          UPDT 153
     C*****GENERATE A GOOD DATA BASE FILE (FILE 4)                            UPDT 154
     C                                                                        UPDT 155
     85     CONTINUE                                                          UPDT 156
            END FILE 3                                                        UPDT 157
            REWIND 3                                                          UPDT 158
            CALL RSZW(IWORDS)                                                 UPDT 159
            CALL KEYW(2,35,108,'A','A',1)                                     UPDT 160
            CALL KEYW(128,35,72,'A','A',2)                                    UPDT 161
            CALL SRTOPN                                                       UPDT 162
     83     READ(3,ERR=97,END=84)WORDS                                        UPDT 163
            CALL SRTREL(WORDS)                                                UPDT 164
            GO TO 83                                                          UPDT 165
     84     CALL SRTSRT                                                       UPDT 166
     89     CALL SRTRET(WORDS,$160)                                           UPDT 167
     81     CONTINUE                                                          UPDT 168
            IF(DATE(1).EQ.BLANK.AND.DATE(2).EQ.BLANK)GO TO 86                 UPDT 169
            GO TO 153                                                         UPDT 170
     152    NNDATA=NNDATA+1                                                   UPDT 171
            DWORDS(127)=DATX(1)                                               UPDT 172
            DWORDS(128)=DATX(2)                                               UPDT 173
            GO TO 155                                                         UPDT 174
```

SUBROUTINE UPDATE (Concluded)

```
      153   NTOT=NTOT+1                                                    UPDT 175
                NOUT=4                                                     UPDT 176
                WRITE(4,ERR=91)WORDS                                       UPDT 177
                GO TO 89                                                   UPDT 178
      86        CONTINUE                                                   UPDT 179
                DO 87 I=1,1WORDS                                           UPDT 180
      87        DWORDS(I)=WORDS(I)                                         UPDT 181
                CALL SRTRET(WORDS,$151)                                    UPDT 182
                DO 88 I=1,3                                                UPDT 183
                IF(DWORDS(I+1).NE.EINUM(I))GO TO 150                       UPDT 184
      88        CONTINUE                                                   UPDT 185
                IF(DWORDS(1).EQ.0)NDDATA=NDDATA+1                          UPDT 186
                IF(DWORDS(1).EQ.0)GO TO 89                                 UPDT 187
                DWORDS(127)=DATX(1)                                        UPDT 188
                DWORDS(128)=DATX(2)                                        UPDT 189
                NUDATA=NUDATA+1                                            UPDT 190
                NTOT=NTOT+1                                                UPDT 191
                NOUT=4                                                     UPDT 192
                WRITE(4,ERR=91)DWORDS                                      UPDT 193
                GO TO 89                                                   UPDT 194
      150       CONTINUE                                                   UPDT 195
                DWORDS(127)=DATX(1)                                        UPDT 196
                DWORDS(128)=DATX(2)                                        UPDT 197
                NNDATA=NNDATA+1                                            UPDT 198
                NTOT=NTOT+1                                                UPDT 199
                NOUT=4                                                     UPDT 200
                WRITE(4,ERR=91)DWORDS                                      UPDT 201
                GO TO 81                                                   UPDT 202
      151       CONTINUE                                                   UPDT 203
                IF(DATE(1).EQ.BLANK.AND.DATE(2).EQ.BLANK)GO TO 152         UPDT 204
      155       NTOT=NTOT+1                                                UPDT 205
                NOUT=4                                                     UPDT 206
                WRITE(4,ERR=91)DWORDS                                      UPDT 207
      160       CONTINUE                                                   UPDT 208
                END FILE 4                                                 UPDT 209
                WRITE(6,107)NNDATA,NUDATA,NDDATA,NTOT                      UPDT 210
      107       FORMAT(//' ***GSE ATTRIBUTE DATA FILE HAS BEEN UPDATED***' UPDT 211
               * / 1X,I4,' NEW ENTITIES',                                  UPDT 212
               * / 1X,I4,' UPDATED ENTITIES',                              UPDT 213
               * / 1X,I4,' DELETED ENTITIES',                              UPDT 214
               * / 1X,I4,' TOTAL ENTITIES USED DURING THIS RUN')           UPDT 215
                RETURN                                                     UPDT 216
                END                                                        UPDT 217
```

SUBROUTINE WGTMET


```
      C ***********************************************************************WGMT    1
      C *  SUBROUTINE WGTMET -THIS SUBROUTINE CONVERTS THE WEIGHT UNITS TO    *WGMT    2
      C * KILOGRAMS FROM POUNDS                                               *WGMT    3
      C ***********************************************************************WGMT    4
      C                                                                        WGMT    5
            SUBROUTINE WGTMET(IWTMET,IWT)                                      WGMT    6
            DIMENSION IW(6)                                                    WGMT    7
      C                                                                        WGMT    8
      C*****FLD IS A LIBRARY FUNCTION TO OBTAIN BITS FROM ALPHANUMERIC NUMBERS WGMT    9
      C                                                                        WGMT   10
            IW(1)=FLD(0,6,IWT)                                                 WGMT   11
            IW(2)=FLD(6,6,IWT)                                                 WGMT   12
            IW(3)=FLD(12,6,IWT)                                                WGMT   13
            IW(4)=FLD(18,6,IWT)                                                WGMT   14
            IW(5)=FLD(24,6,IWT)                                                WGMT   15
            IW(6)=FLD(30,6,IWT)                                                WGMT   16
            IADD=0                                                             WGMT   17
            DO 1 K=1,6                                                         WGMT   18
            IF(IW(K).NE.5) GO TO 2                                             WGMT   19
            IADD=IADD+1                                                        WGMT   20
            GO TO 1                                                            WGMT   21
       2    IF(IW(K).EQ.25) GO TO 3                                            WGMT   22
            IF(IW(K).EQ.19) GO TO 3                                            WGMT   23
       1    CONTINUE                                                           WGMT   24
            IF(IADD.EQ.0) GO TO 90                                             WGMT   25
            I=0                                                                WGMT   26
            DO 80 J=1,IADD                                                     WGMT   27
            IF(IW(J).NE.5) GO TO 81                                            WGMT   28
            I=I+1                                                              WGMT   29
       80   CONTINUE                                                           WGMT   30
       81   GO TO (91,92,93,94,95,3),IADD                                      WGMT   31
      C                                                                        WGMT   32
      C*****CALCULATE INTEGER NUMBER(DEPENDING ON NUMBER OF DIGITS)            WGMT   33
      C                                                                        WGMT   34
       90   IREAL=(IW(6)-48)+(IW(5)-48)*10+(IW(4)-48)*100+(IW(3)-48)*1000+     WGMT   35
           1((IW(2)-48)*10000+(IW(1)-48)*100000                                WGMT   36
            GO TO 96                                                           WGMT   37
       91   IREAL=(IW(I+5)-48)+(IW(I+4)-48)*10+(IW(I+3)-48)*100+(IW(I+2)-48)*  WGMT   38
           1*1000+(IW(I+1)-48)*10000                                           WGMT   39
            GO TO 96                                                           WGMT   40
       92   IREAL=(IW(I+4)-48)+(IW(I+3)-48)*10+(IW(I+2)-48)*100+(IW(I+1)-48)   WGMT   41
           1*1000                                                              WGMT   42
            GO TO 96                                                           WGMT   43
       93   IREAL=(IW(I+3)-48)+(IW(I+2)-48)*10+(IW(I+1)-48)*100                WGMT   44
            GO TO 96                                                           WGMT   45
       94   IREAL=(IW(I+2)-48)+(IW(I+1)-48)*10                                 WGMT   46
            GO TO 96                                                           WGMT   47
       95   IREAL=IW(I+1)-48                                                   WGMT   48
       96   REAL=IREAL                                                         WGMT   49
      C                                                                        WGMT   50
      C*****CONVERT REAL NUMBER TO METRIC                                      WGMT   51
      C                                                                        WGMT   52
            XWTMET=REAL*0.45359237                                             WGMT   53
      C                                                                        WGMT   54
      C*****CALL SUBROUTINE TO CONVERT REAL NUMBER TO ALPHANUMERIC NUMBER      WGMT   55
      C                                                                        WGMT   56
            CALL RELALP(IWTMET,XWTMET,6)                                       WGMT   57
            GO TO 999                                                          WGMT   58
       3    IWTMET=IWT                                                         WGMT   59
       999  RETURN                                                             WGMT   60
            END                                                                WGMT   61
```

APPENDIX C. SUPPLEMENT RUN PROCEDURES

APPENDIX C. SUPPLEMENT RUN PROCEDURES (The Univac 1108 Run Cards)

Before a run can be submitted to a computer operator at the Univac 1108 remotes or at central site (Computation Laboratory), two special computer cards must appear in front of all other cards in the card-deck runstream. They are simply referred to as Run Card No. 1 (Fig. C-1) and Run Card No. 2 (Fig. C-2).

The following procedure should aid in completing these forms correctly:

1. Punch Run Card No. 1 exactly as shown in Figure C-1 except that keypuncher punches the first six letters of his last name in card columns 24 through 29. If the name is less than six characters, the space should be filled with some arbitrary letters.

2. Punch Run Card No. 2 exactly as shown in Figure C-2 on the preprinted green run card except for any comment. Note that one really needs only to duplicate Run Card No. 2 from Run Card No. 1 except card columns 65 through 80 are left blank.

3. Printed information must be added to Run Card No. 2 as shown in Figure C-2. Write "32" for CORESIZE and "207" or "225" for BIN NO. Write RUN " 1 of 1." Mark an "X" in the NO columns of PUNCH $ and PLOTS. Write the current tape number (4 or 5 digits) under INPUT TAPES. Remember that these two cards precede all other cards in the runstream.

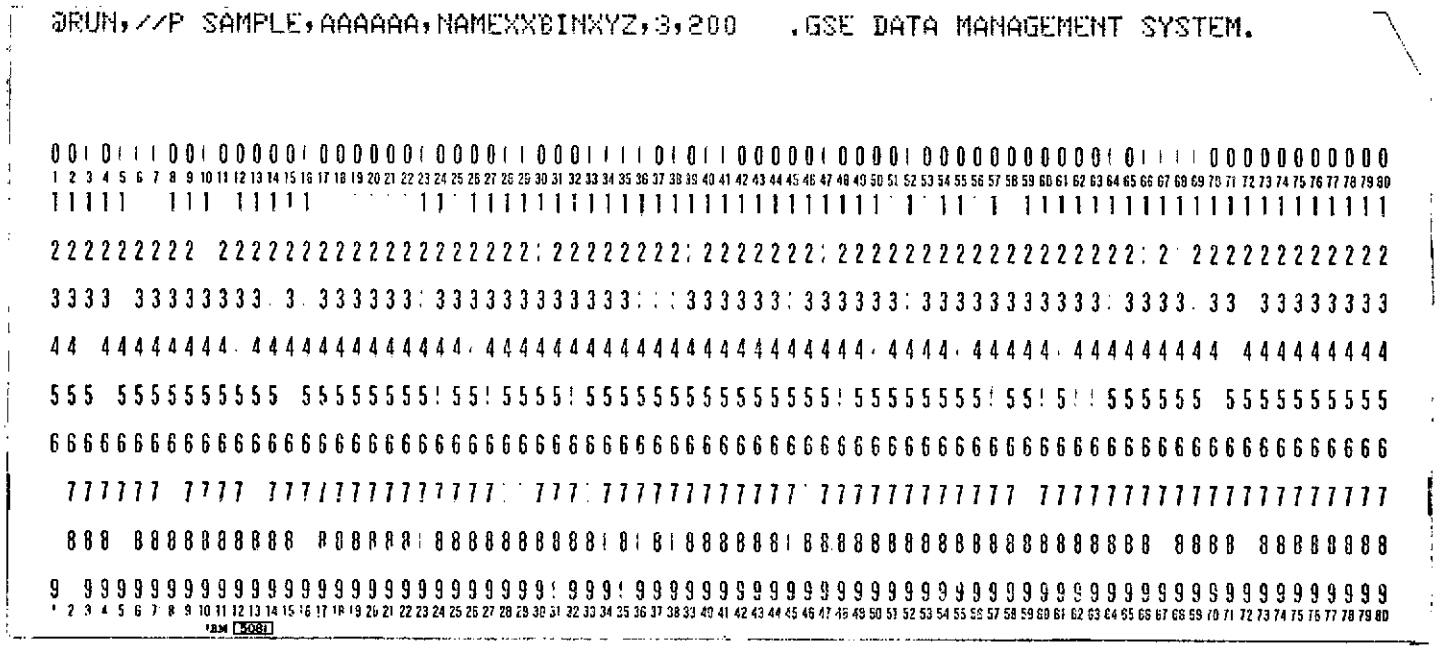

Figure C-1. Sample Run Card No. 1.

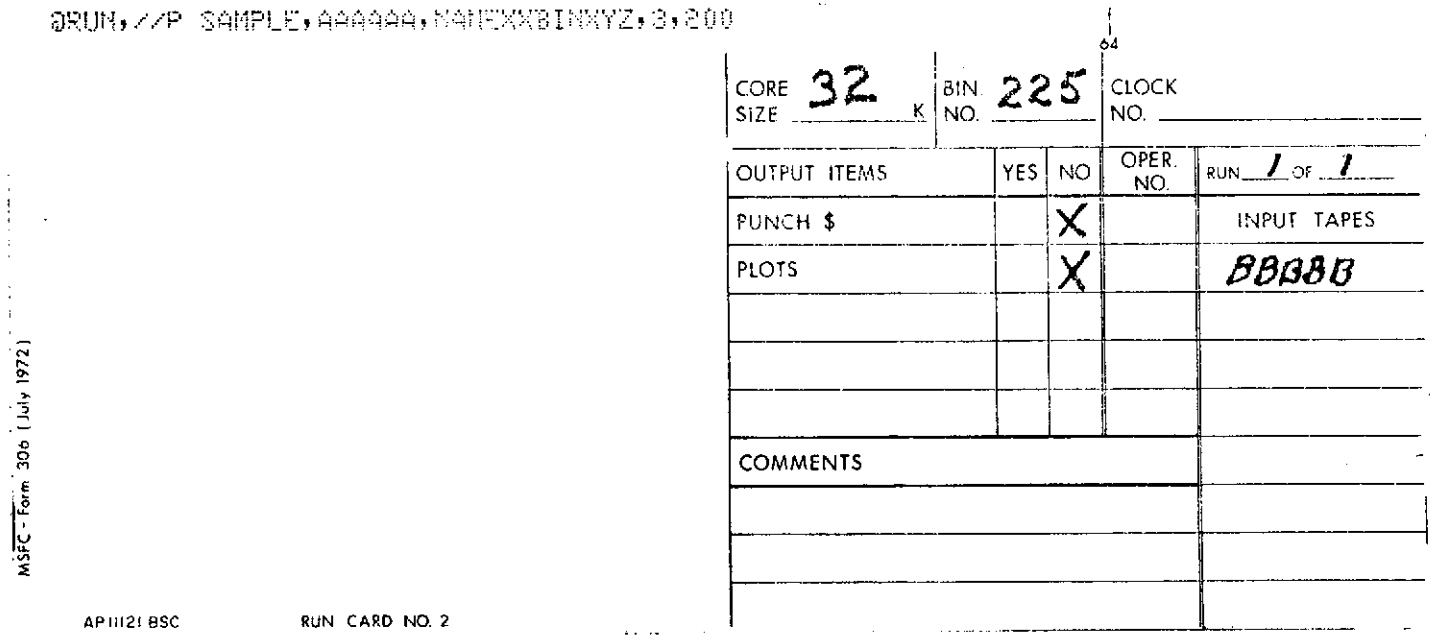

Figure C-2. Sample Run Card No. 2.

APPROVAL

GSE DATA MANAGEMENT SYSTEM PROGRAMMERS'/USERS' MANUAL

By Ronald A. Schlagheck, Bert D. Dolerhie, Jr., and Fred J. Ghiglieri

The information in this report has been reviewed for security classification. Review of any information concerning Department of Defense or Atomic Energy Commission programs has been made by the MSFC Security Classification Officer. This report, in its entirety, has been determined to be unclassified.

This document has also been reviewed and approved for technical accuracy.

O. M. HARDAGE
Chief, Operations Planning and Analysis Branch

F. KURTZ
Chief, Operations Development Division

H. E. THOMASON
Systems Analysis and Integration Laboratory

☆ U.S. GOVERNMENT PRINTING OFFICE 1974 — 640-444 / 46 REGION NO. 4