Source Data Entry/2000

Reference Manual

HEWLETT PACKARD

5303 STEVENS CREEK BLVD., SANTA CLARA, CALIFORNIA, 95050

Part No. 20243-90001
Product No. 20243A

Printed in U.S.A. 2/76

NOTICE

The information contained in this document is subject to change without notice.

HEWLETT-PACKARD MAKES NO WARRANTY OF ANY KIND WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Hewlett-Packard shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance or use of this material.

Hewlett-Packard assumes no responsibility for the use or reliability of its software on equipment that is not furnished by Hewlett-Packard.

LIST OF EFFECTIVE PAGES

The List of Effective Pages gives the most recent date on which the technical material on any given page was altered. If a page is simply re-arranged due to a technical change on a previous page, it is not listed as a changed page. Within the manual, changes are marked with a vertical bar in the margin.

Title	Feb 1976
ii to vii	Feb 1976
1-0 to 1-5	Feb 1976
2-1 to 2-24	Feb 1976
3-1 to 3-7	Feb 1976
4-1 to 4-6	Feb 1976
5-1 to 5-8	Feb 1976
A-1 to A-3	Feb 1976
B-1 to B-2	Feb 1976
C-1 to C-13	Feb 1976
D-1	Feb 1976
I-1 to I-3	Feb 1976

PRINTING HISTORY

New editions incorporate all update material since the previous edition. Update packages, which are issued between editions, contain additional and replacement pages to be merged into the manual by the customer. The date on the title page and back cover changes only when a new edition is published. If minor corrections and updates are incorporated, the manual is reprinted but neither the date on the title page and back cover nor the edition change.

First Edition . . . . Feb 1976

PREFACE

This manual describes Source Data Entry/2000 (SDE/2000), an application designed for entering data into files on the HP 2000 Access System using the HP 2640A and HP 2644A terminals.

All future references to the 2640A terminal in this manual also apply to the 2644A unless specified otherwise. In this application the 2644A can only be used as an interactive display terminal; SDE/2000 does not support use of the tape cartridges on the 2644A terminal.

SDE/2000 provides capabilities for designing forms, displaying forms on the terminal screen, accepting data input from the terminal, checking the data for errors, and storing the error-free data in a file on the HP 2000 Access System. The stored data may be accessed by user-written BASIC language programs, or it may be transmitted to a remote IBM computer supporting HASP Multileaving RJE or a CDC computer supporting UT200 RJE.

The manual is directed to:

- the forms designer who transcribes business forms to the terminal
- the terminal operator who enters data onto the forms
- the applications programmer who works with the data files produced by SDE/2000.

Three additional manuals provide supplementary information for users of SDE/2000.

> *2640A Interactive Display Terminal Owner's Manual,* HP 02640-90011 contains complete instructions for operating the HP 2640A terminal.
>
> *2644A Mini DataStation Owner's Manual,* HP 02644-90001 contains complete instructions for operating the HP 2644A terminal.
>
> *2000 Access/BASIC Reference Manual,* HP 22687-90001 provides information for developing and executing BASIC language programs on the HP 2000 Access System.

CONTENTS

Section I Page

INTRODUCING SDE/2000

Section II Page

DESIGNING FORMS

Section III Page

SDE/2000 EDIT RULES

Section IV Page

HOW TO ENTER DATA

Section V Page

USER PROGRAMS

Appendix A Page

LOADING SDE/2000

Appendix B Page

Appendix C Page

EXAMPLES

Appendix D Page

ILLUSTRATIONS

TABLES

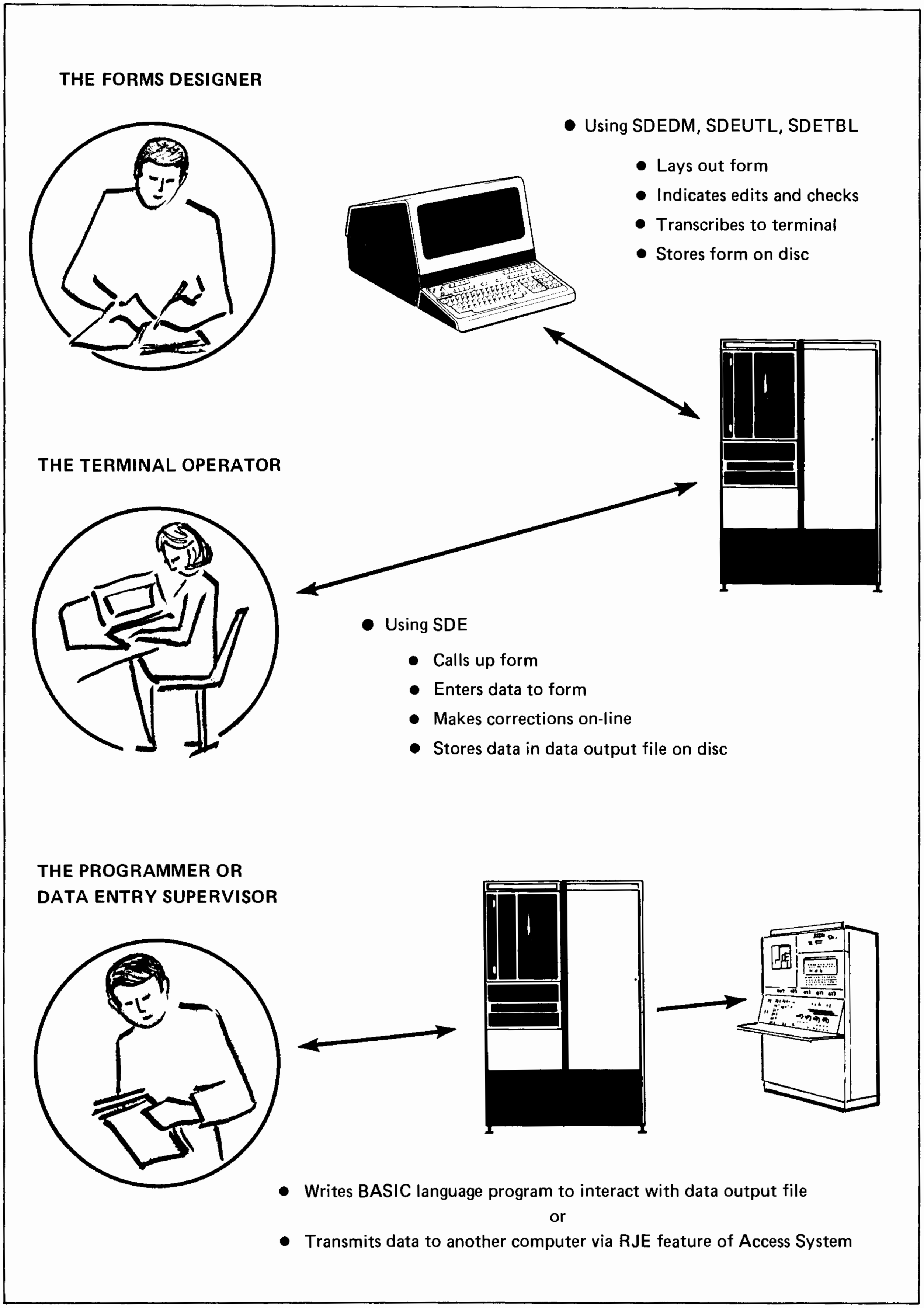

THE FORMS DESIGNER
● Using SDEDM, SDEUTL, SDETBL
● Lays out form
● Indicates edits and checks
● Transcribes to terminal
● Stores form on disc
THE TERMINAL OPERATOR
● Using SDE
● Calls up form
● Enters data to form
● Makes corrections on-line
● Stores data in data output file on disc
THE PROGRAMMER OR
DATA ENTRY SUPERVISOR
● Writes BASIC language program to interact with data output file
or
● Transmits data to another computer via RJE feature of Access System

Figure 1-1. SDE/2000 Users

INTRODUCING SDE/2000

SECTION I

Source Data Entry/2000 -- SDE/2000 -- is a multi-purpose data input system.

- It provides capabilities for designing forms on the 2640A terminal . . . . no programming knowledge is required by the forms designer.

- It displays the forms on the 2640A terminal, allowing a terminal operator to enter data as if he were typing directly on the form.

- It edits the entered data immediately and allows the terminal operator to correct errors, or it stores data for later correction.

- Once the data is collected and edited by SDE/2000, it can be used in several ways. . . .

  - SDE/2000 can chain to user-written programs to access the data directly
    -or-
  - data can be stored on disc to be accessed by user-written programs later
    -or-
  - data can be sent to a remote IBM computer supporting HASP Multileaving RJE or a CDC computer supporting UT200 RJE.

To accomplish these functions, SDE/2000 consists of four modules:

SDEDM	for designing forms
SDEUTL	for displaying, listing and modifying forms
SDETBL	for accessing table files
SDE	for entering data

Refer to figure 1-1 for a simplified overview of SDE/2000 functions and users.

1-1. TERMINOLOGY

In general business terminology, a *file* is a collection of information arranged in some order. Files are divided into *records* and records are divided into *fields*. On 2000 Access, a file is an area of external memory (such as disc) where information can be stored and retrieved. A record is a division of a file, and a record consists of one or more fields. In this manual each individual entry on a form will be called a field. The 2640A terminal has two types of fields: *protected fields* where the display cannot be changed by the terminal user and *unprotected fields* where the data is entered. A *page* is the information displayed on the 2640A terminal. Up to three pages may be held in the 2640A memory. These three pages are called a *screen*. The file where the form is stored is called the *screen file*.

1-2. DESIGNING FORMS

SDE/2000 takes care of the programming while the designer concentrates on the layout of the form. Figure 1-2 illustrates an example of a document ideally suited for SDE/2000, and figure 1-3 shows how the designer uses the SDE/2000 Forms Layout Sheet to lay out the document.

Large forms may be segmented into a series of automatically linked screens or each screen can be stored in up to three pages of local 2640A terminal memory. In the example of the purchase requisition order, the form was split into two screens. One screen contains the shipping and billing information, and another screen contains the quantity, part number, description and price information.

After deciding the layout of the form, the designer executes SDEDM to enter information about the position and contents of the data fields, and selects the error-checking functions (edit rules) that apply to each input field. You may specify up to five separate error-checking functions or edit rules for each input field.

After SDEDM has stored the form, the designer executes SDEUTL to display the form and modify it if necessary. SDEUTL may also be used to list the form and provide the designer with a hard copy of the screen file.

Data fields that are highly repetitive and complex, or that require special formatting can be stored and automatically supplied from a table. This saves data entry time and reduces errors. The document displayed in figure 1-2 shows examples of several table look-ups. The designer executes SDETBL to make table files available to SDE/2000.

Designing forms is described in Sections II and III, and further examples are included in Appendix C.

1-3. DATA ENTRY PROGRAM

Once the form has been designed and stored on the Access System, a terminal operator can begin data entry by executing SDE. This interactive program displays the form on the 2640A terminal, checks the data for errors and stores the data in a designated file.

The method of entering data is described in Section IV.

1-4. ACCESSING SDE/2000 DATA

The data collected by SDE can be used in several ways. User-written BASIC language programs can read and write the edited output files produced by SDE/2000. Moreover, user-written programs can chain to SDE and use the data directly, thus eliminating the need to store edited output in files.

Another application is to send the edited data to a remote IBM computer supporting HASP/RJE or a CDC computer supporting UT200 RJE. This can be performed under the control of a user's 2000 Access/BASIC program, or by use of the Telecommunications Supervisory Package/2000 for HASP Multileaving RJE (TSP/2000).

Refer to Section V for the exact method of chaining to the SDE program. Section V also describes the formats of SDE/2000 files which are needed for writing interface programs.

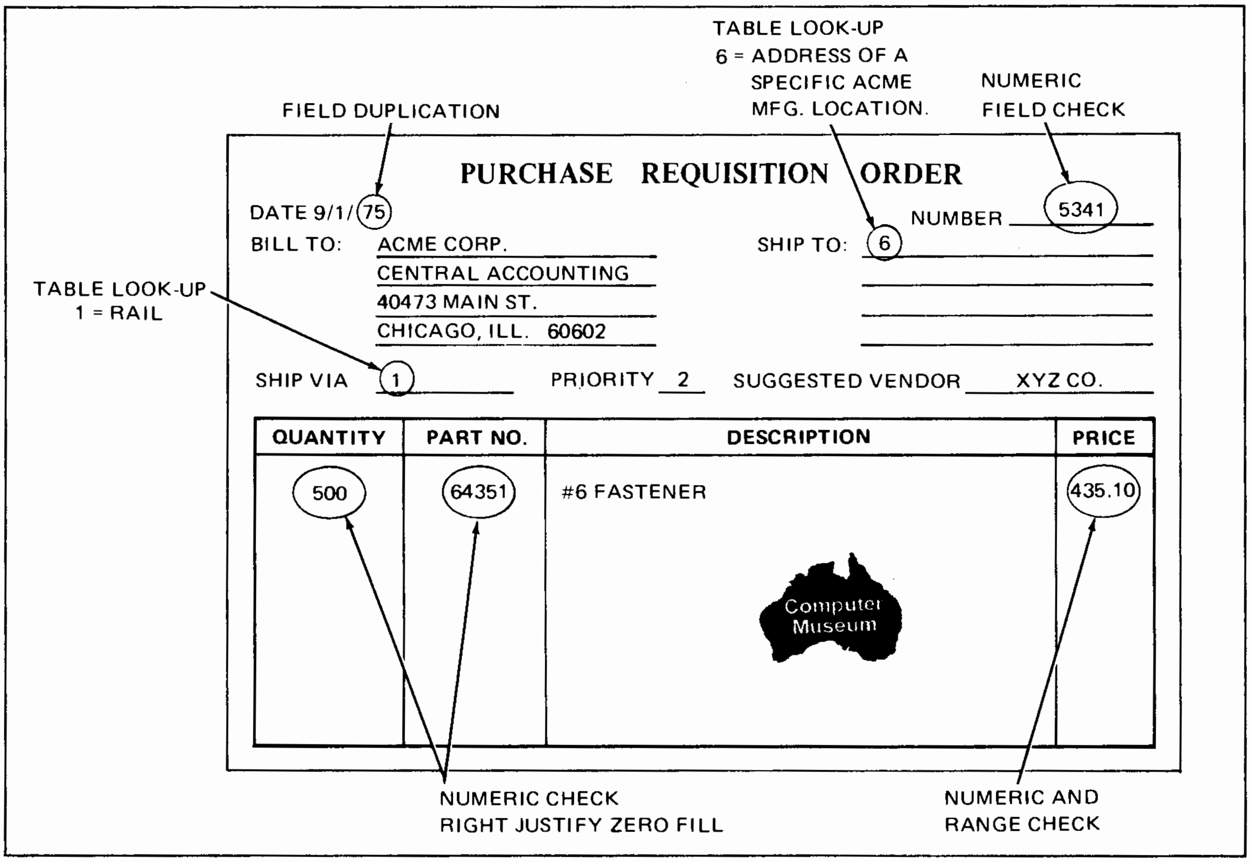


TABLE LOOK-UP
6 = ADDRESS OF A SPECIFIC ACME MFG. LOCATION.

NUMERIC FIELD CHECK

FIELD DUPLICATION

TABLE LOOK-UP
1 = RAIL

PURCHASE REQUISITION ORDER

DATE 9/1/75

NUMBER 5341

BILL TO: ACME CORP.
CENTRAL ACCOUNTING
40473 MAIN ST.
CHICAGO, ILL. 60602

SHIP TO: 6

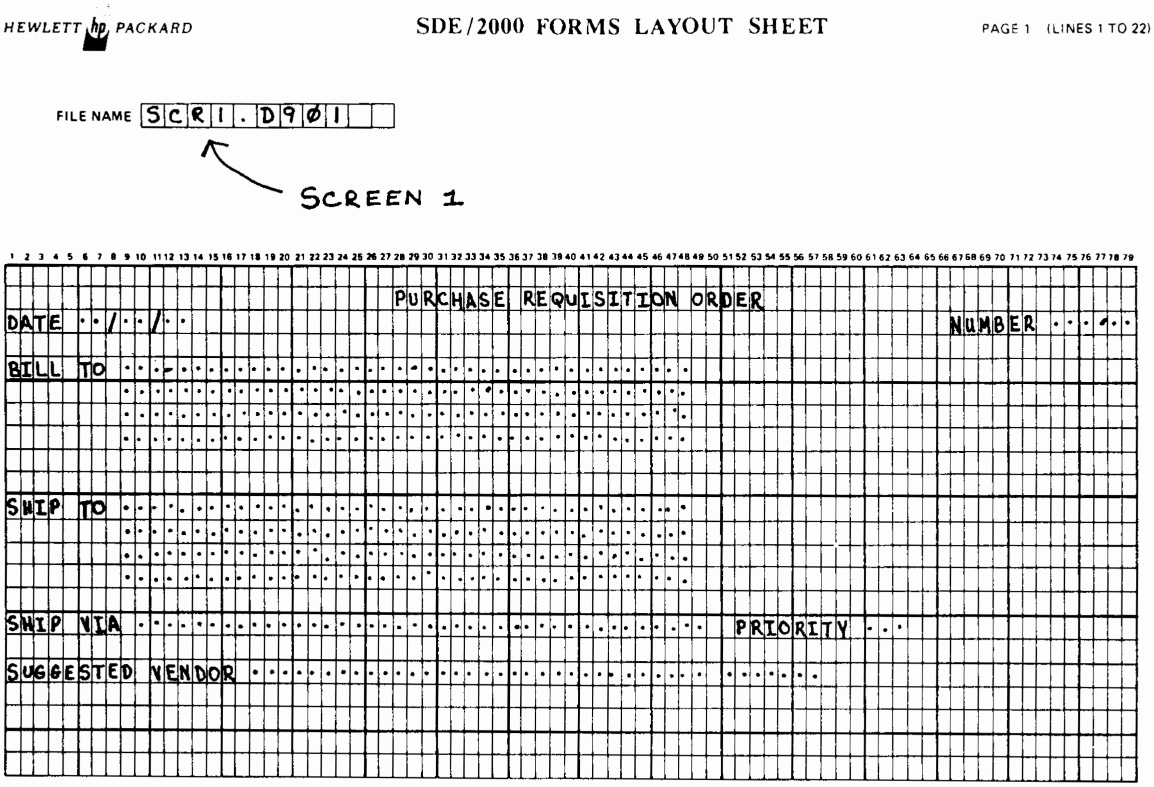
SHIP VIA 1 PRIORITY 2 SUGGESTED VENDOR XYZ CO.

QUANTITY	PART NO.	DESCRIPTION	PRICE
500	64351	#6 FASTENER	435.10

NUMERIC CHECK
RIGHT JUSTIFY ZERO FILL

NUMERIC AND
RANGE CHECK

Figure 1-2. Purchase Requisition Order



HEWLETT hp PACKARD

SDE/2000 FORMS LAYOUT SHEET

PAGE 1 (LINES 1 TO 22)

FILE NAME SCR1.D9Ø1

SCREEN 1

```
 1
 2                            PURCHASE REQUISITION ORDER
 3 DATE ../../..                                                     NUMBER ......
 4
 5 BILL TO ........................................
 6         ........................................
 7         ........................................
 8         ........................................
 9
10
11 SHIP TO ........................................
12         ........................................
13         ........................................
14         ........................................
15
16 SHIP VIA  ........................................ PRIORITY ...
17
18 SUGGESTED VENDOR ........................................
19
20
21
22
```

Figure 1-3. Forms Layout Sheet

1-5. HARDWARE AND SOFTWARE REQUIREMENTS

SDE/2000 is written in HP 2000 Access/BASIC and runs on the HP 2000 Access system. The only I/O devices supported for SDE/2000 are the HP 2640A and HP 2644A terminals.

- IF YOU ARE USING THE HP 2640A

  The 2640A must have the following features:

Option 001	128 Character Set
13231A	Display Enhancements
13234A	Terminal Memory Module (+4K)

  The 2640A memory can be expanded to a maximum of 8K. The additional memory may be needed for complex forms.

  The 2640A terminal must be internally strapped for page mode operation. Normally this option is disabled and the terminal is set for line-field operation when the BLOCK MODE key is depressed. To convert the terminal to page mode, a jumper must be removed from the Keyboard Interface PCA. This procedure is described in Appendix A.

- IF YOU ARE USING THE HP 2644A

  The 2644A can only be used as an interactive display terminal; SDE/2000 does not support use of the tape cartridges on this terminal.

  The 2644A must have the following features:

Option 001	128 Character Set
13231A	Display Enhancements

  The 2644A is provided with 4K of memory and is not expandable.

  The 2644A terminal must be internally strapped for page mode operation. To convert the terminal to page mode, a jumper must be removed from the Keyboard Interface PCA. In the 2644A there are additional jumpers on this board that affect block mode operation and should be checked. These procedures are described in Appendix A.

  All future references to the 2640A in this manual also apply to the 2644A unless specified otherwise.

1-6. ACCOUNT STRUCTURE

SDE/2000 requires an account structure consisting of one group account and at least two user accounts in the same group. All programs are loaded into the group account. Forms designers run the forms design program (SDEDM) and store screen description files under one user account. Terminal operators who record data must enter data using another user account. This account structure helps prevent accidental destruction of programs and files. Multiple users can run SDE/2000 programs at the same time since each user writes to a unique output data file.

Figure 1-4 illustrates SDE/2000 set up in accounts D900, D901 and D902.

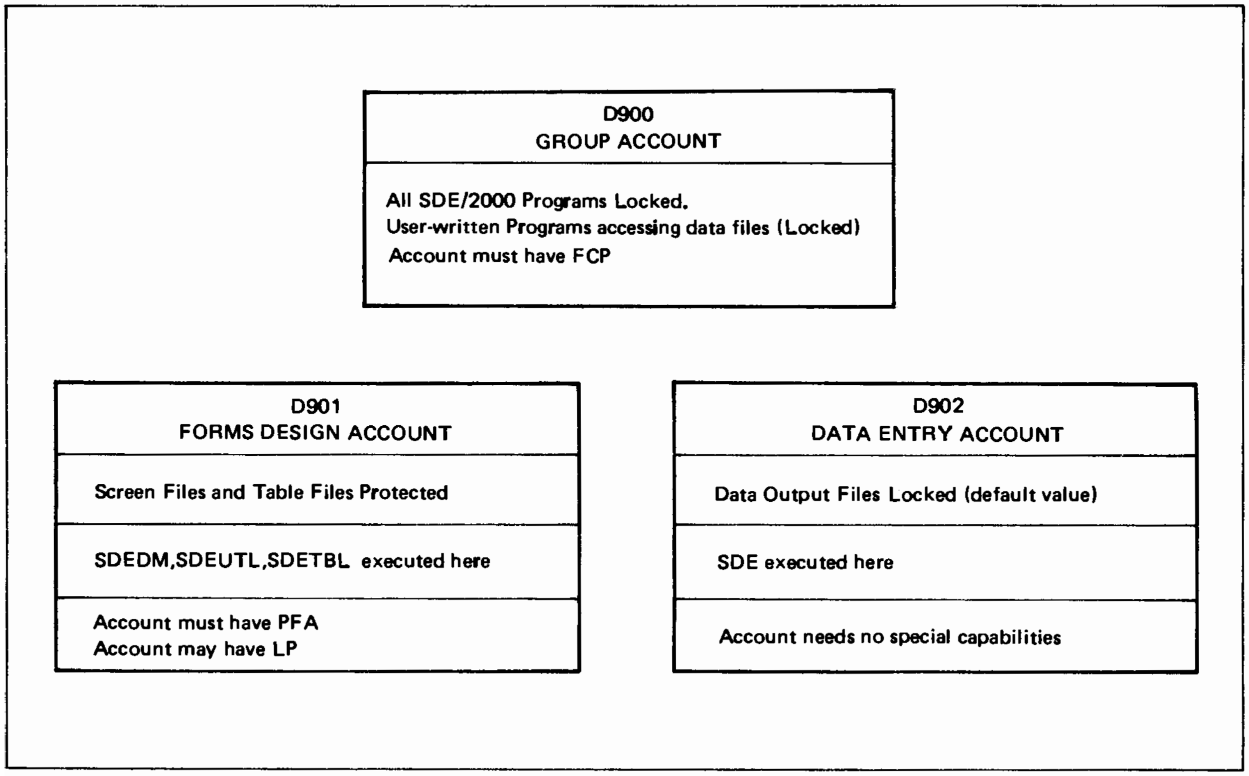

D900
GROUP ACCOUNT
All SDE/2000 Programs Locked.
User-written Programs accessing data files (Locked)
Account must have FCP
D901
FORMS DESIGN ACCOUNT
Screen Files and Table Files Protected
SDEDM,SDEUTL,SDETBL executed here
Account must have PFA
Account may have LP
D902
DATA ENTRY ACCOUNT
Data Output Files Locked (default value)
SDE executed here
Account needs no special capabilities

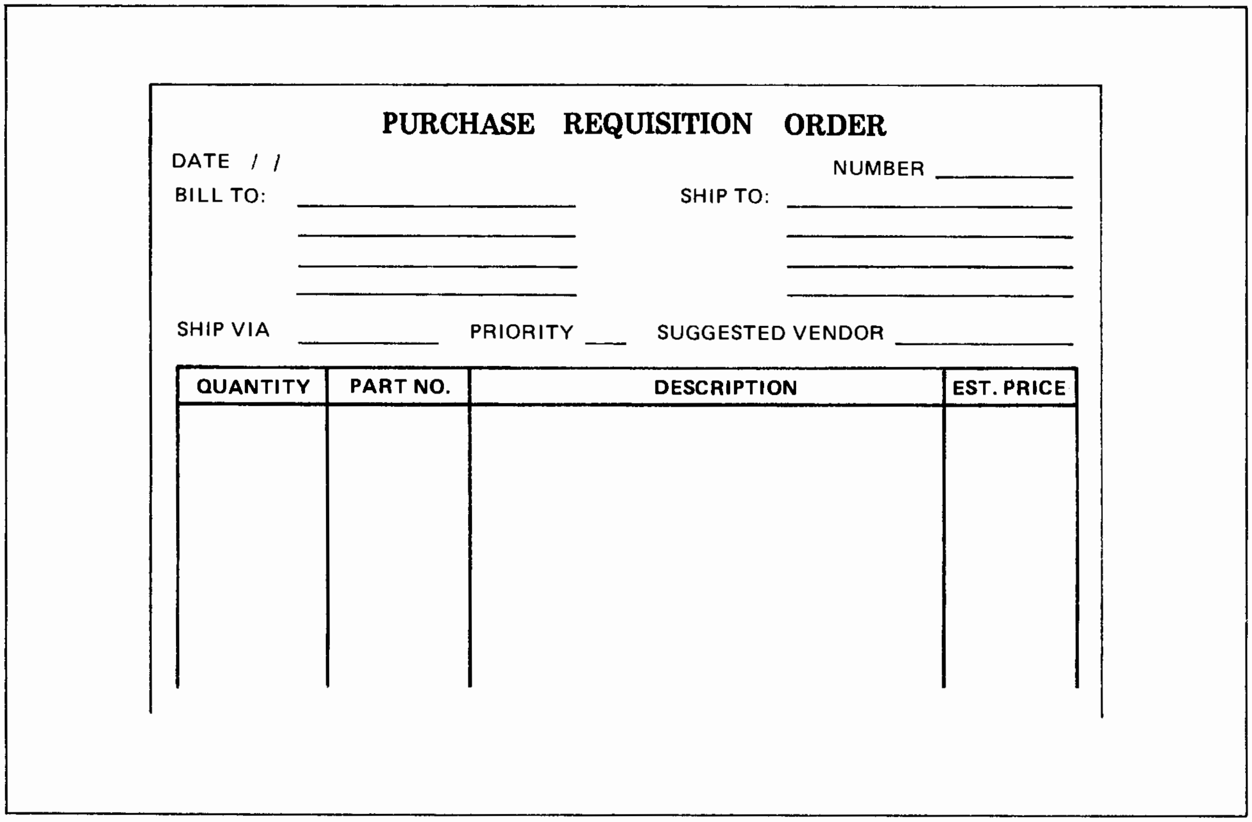
Figure 1-4. Account Structure

DESIGNING FORMS

SECTION II

2-1. SELECTING THE FORM

The function of SDE/2000 is to provide data entry from business forms displayed on the screen on an HP 2640A terminal. The first step in the design process is the selection of the form to be used. Figure 2-1 shows a typical purchase requisition order form.



PURCHASE REQUISITION ORDER

DATE / /

NUMBER ___________

BILL TO: ______________________

SHIP TO: ______________________

SHIP VIA ___________ PRIORITY ___ SUGGESTED VENDOR ____________

QUANTITY	PART NO.	DESCRIPTION	EST. PRICE

Figure 2-1. Typical Business Form

2-2. LAYING OUT THE FORM

The second step for the designer is to decide how to represent the chosen business form on the terminal. The information on the form falls into two broad categories: the *preprinted* areas and the input areas. The preprinted areas consist of company letterheads, instructions, labels, etc. - areas read by the user, but not changed. The *preprinted* areas are called *protected* fields. The input areas are designed to record data and will be filled in by the user. The *input* areas are *unprotected* fields.

In most cases the fields must be slightly rearranged to fit on the 79-column screen of the terminal. In some cases the form may be too large to fit on one screen of the terminal, even after rearrangement. SDE/2000 provides the designer with two possible solutions: *multi-page* capability and *multi-screen* capability. In both methods the form is divided into logical sections.

Multi-Page Capability

Multi-page capability takes advantage of the paging feature of the HP 2640A terminal. This capability allows a form larger than the size of the screen to be sent to the terminal and stored in its memory. The form is then accessed 24 lines at a time by using the NEXT PAGE and PREV PAGE keys on the terminal. SDE/2000 supports multi-page forms up to three pages in length.

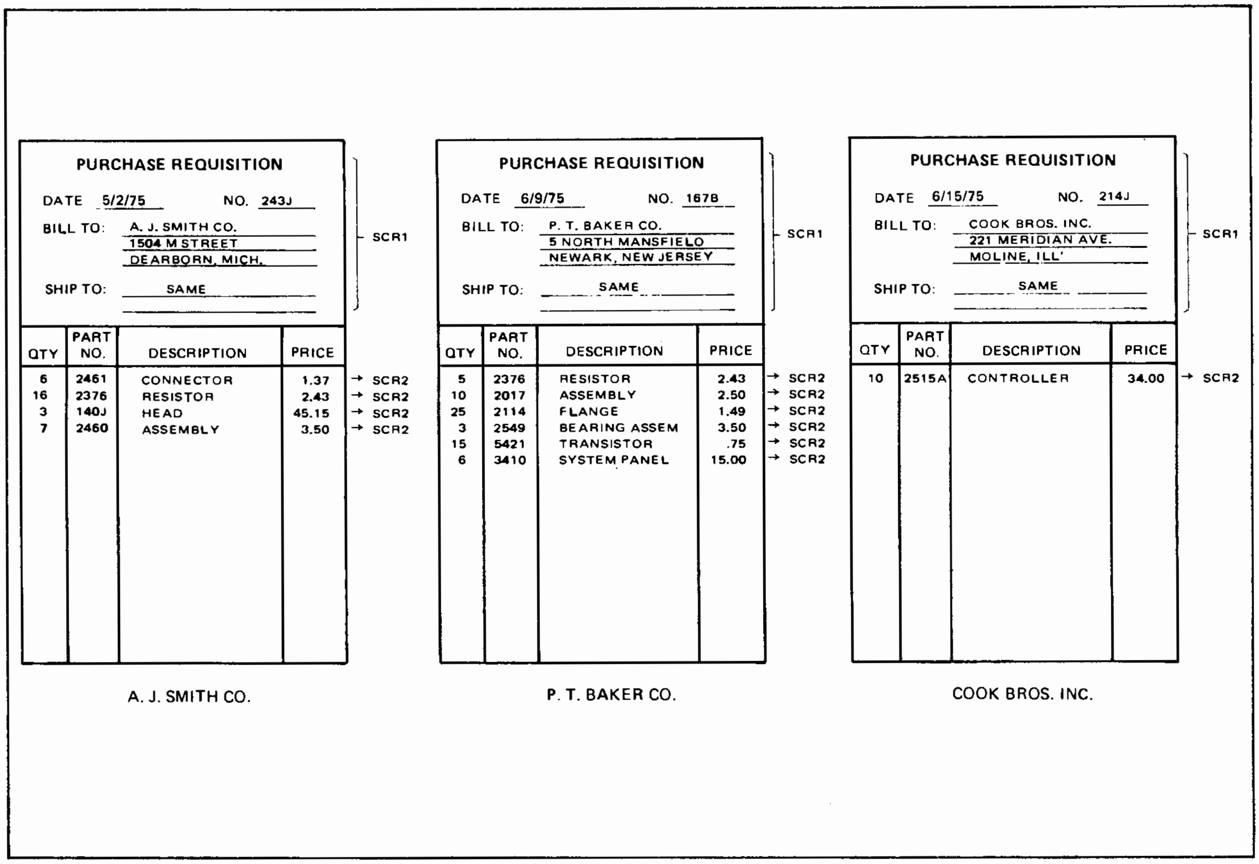
This method is useful for large forms with little or no repetition of the same type of data. Since the sample purchase requisition order form requests the same type of data for each part to be ordered, it is better suited for multi-screen capability.

Multi-Screen Capability

SDE/2000 gives the designer the capability to separate the logical sections of his form into separate screen files which are linked together. This is advantageous when one section of the form is highly repetitious as in the part data section of our example purchase requisition order.

Since an arbitrary number of parts can be ordered on the purchase requisition order, we need the ability to use Screen 2 repeatedly, until all items are entered. Splitting the form into two screens provides this ability. Place the shipping and billing information on one screen (call it SCR1) and the item description information on another screen (SCR2).

This will allow the terminal operator to fill in SCR2 as many times as necessary to enter all of the parts on the purchase requisition order. As shown in figure 2-2, Cook Bros., Inc. is ordering only one part, and SCR2 will be filled in only once. However, since A.J. Smith Co. and P.T. Baker Co. are ordering more parts, SCR2 may be repeatedly displayed until all parts are entered.



PURCHASE REQUISITION

DATE 5/2/75 NO. 243J

BILL TO: A. J. SMITH CO.
1504 M STREET
DEARBORN, MICH.

SHIP TO: SAME

QTY	PART NO.	DESCRIPTION	PRICE
6	2461	CONNECTOR	1.37
16	2376	RESISTOR	2.43
3	140J	HEAD	45.15
7	2460	ASSEMBLY	3.50

A. J. SMITH CO.

PURCHASE REQUISITION

DATE 6/9/75 NO. 167B

BILL TO: P. T. BAKER CO.
5 NORTH MANSFIELD
NEWARK, NEW JERSEY

SHIP TO: SAME

QTY	PART NO.	DESCRIPTION	PRICE
5	2376	RESISTOR	2.43
10	2017	ASSEMBLY	2.50
25	2114	FLANGE	1.49
3	2549	BEARING ASSEM	3.50
15	5421	TRANSISTOR	.75
6	3410	SYSTEM PANEL	15.00

P. T. BAKER CO.

PURCHASE REQUISITION

DATE 6/15/75 NO. 214J

BILL TO: COOK BROS. INC.
221 MERIDIAN AVE.
MOLINE, ILL'

SHIP TO: SAME

QTY	PART NO.	DESCRIPTION	PRICE
10	2515A	CONTROLLER	34.00

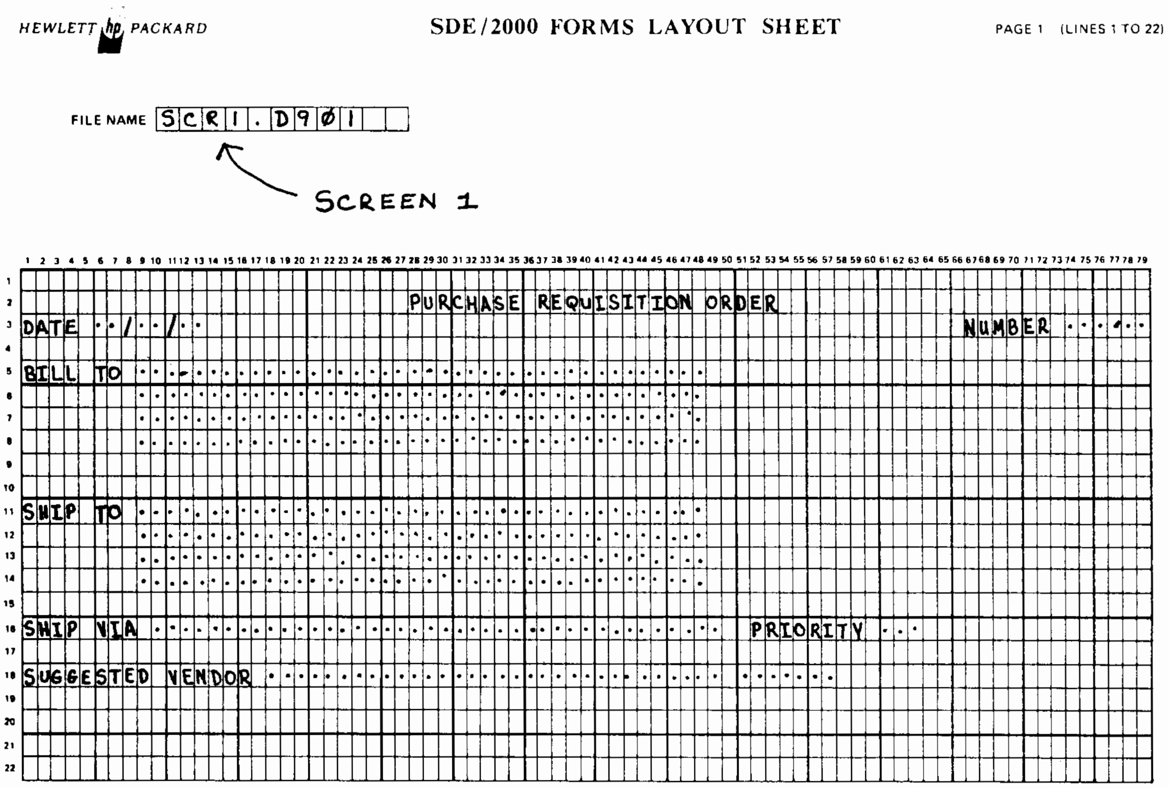
COOK BROS. INC.

Figure 2-2. Use of Multiple Screens

At data entry time the terminal operator will call up SCR1 and fill in the shipping and billing information. SCR2 will be linked to SCR1 and automatically displayed once the SCR1 data is accepted. SCR2 is continually displayed as many times as necessary to complete entry of the information.

After deciding what the form will look like and dividing it into screens (if necessary), the next step is to lay out each screen exactly as it will appear on the terminal. In laying out the form, keep in mind several considerations: (1) each page contains 22 lines and each line has 79 characters, (2) individual fields may have up to 40 characters, (3) the characters in a field may appear with inverse video, half bright, and underline display enhancements. Several layout forms are provided in Appendix D that can be used as worksheets. They are essentially grids that represent that part of the screen available to the designer.

Figures 2-3 and 2-4 are SDE layout sheets showing how the purchase requisition order might be done using two screens. In these examples, the designer chose to represent his unprotected fields (input areas) by dots.



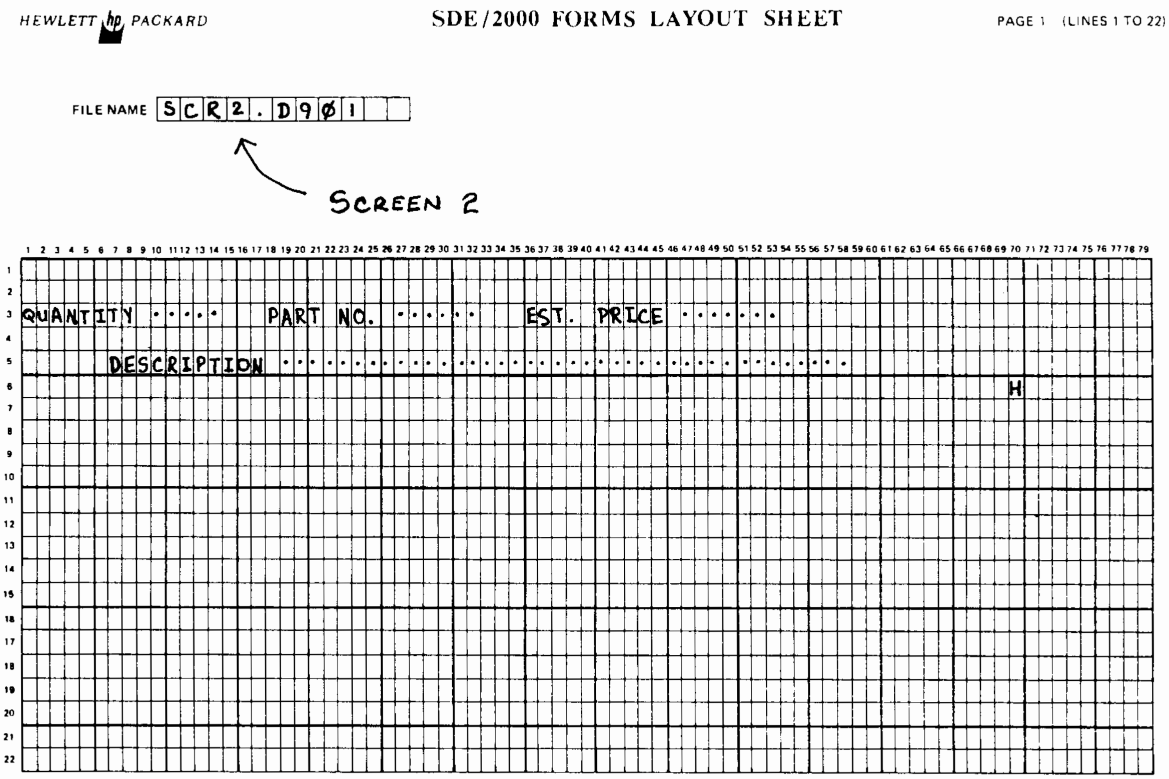
HEWLETT hp PACKARD   SDE/2000 FORMS LAYOUT SHEET   PAGE 1 (LINES 1 TO 22)

FILE NAME SCR1.D9Ø1

SCREEN 1

```
 2                            PURCHASE REQUISITION ORDER
 3 DATE ··/··/··                                                    NUMBER ······
 5 BILL TO   ······································
 6           ······································
 7           ······································
 8           ······································
11 SHIP TO   ······································
12           ······································
13           ······································
14           ······································
16 SHIP VIA  ·······································  PRIORITY ···
18 SUGGESTED VENDOR ·········································
```

Figure 2-3. Purchase Requisition Order, Screen 1



HEWLETT hp PACKARD   SDE/2000 FORMS LAYOUT SHEET   PAGE 1 (LINES 1 TO 22)

FILE NAME SCR2.D9Ø1

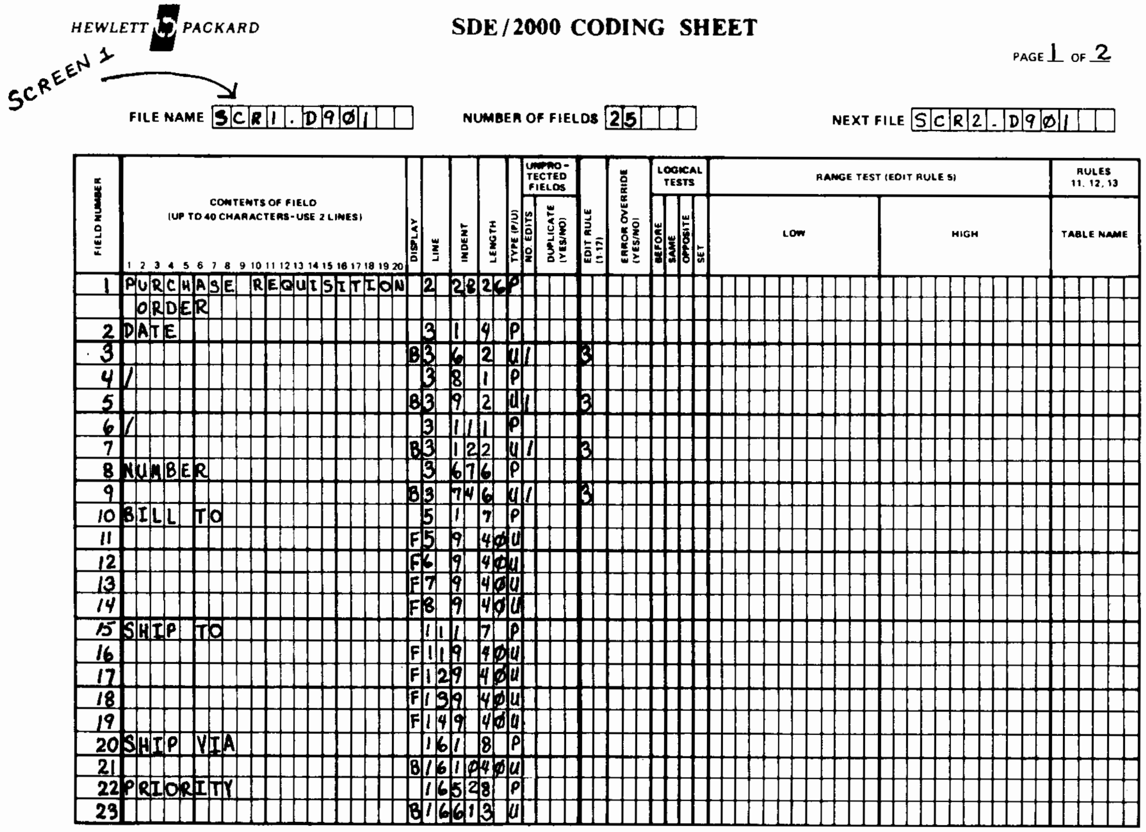
SCREEN 2

```
 3 QUANTITY ·····     PART NO. ······    EST. PRICE ······
 5       DESCRIPTION ··············································
 6                                                                      H
```

Figure 2-4. Purchase Requisition Order, Screen 2

2-3. CODING THE FORM

After producing a layout it is helpful to record information for each field on a coding sheet as in figures 2-5 through 2-7. (A blank coding sheet is included in Appendix D.) The coding sheet contains columns labeled appropriately for recording information as the designer will have to give it to the Forms Design Program (SDEDM). For each field you must specify the line and column where it begins, whether it is a protected or unprotected field, its length, how it is to be displayed, and its contents. If the field is unprotected, additional information (such as editing and logical tests) may be specified. Each of the entries on the coding sheet is explained in detail later in this section.



HEWLETT PACKARD

SDE/2000 CODING SHEET

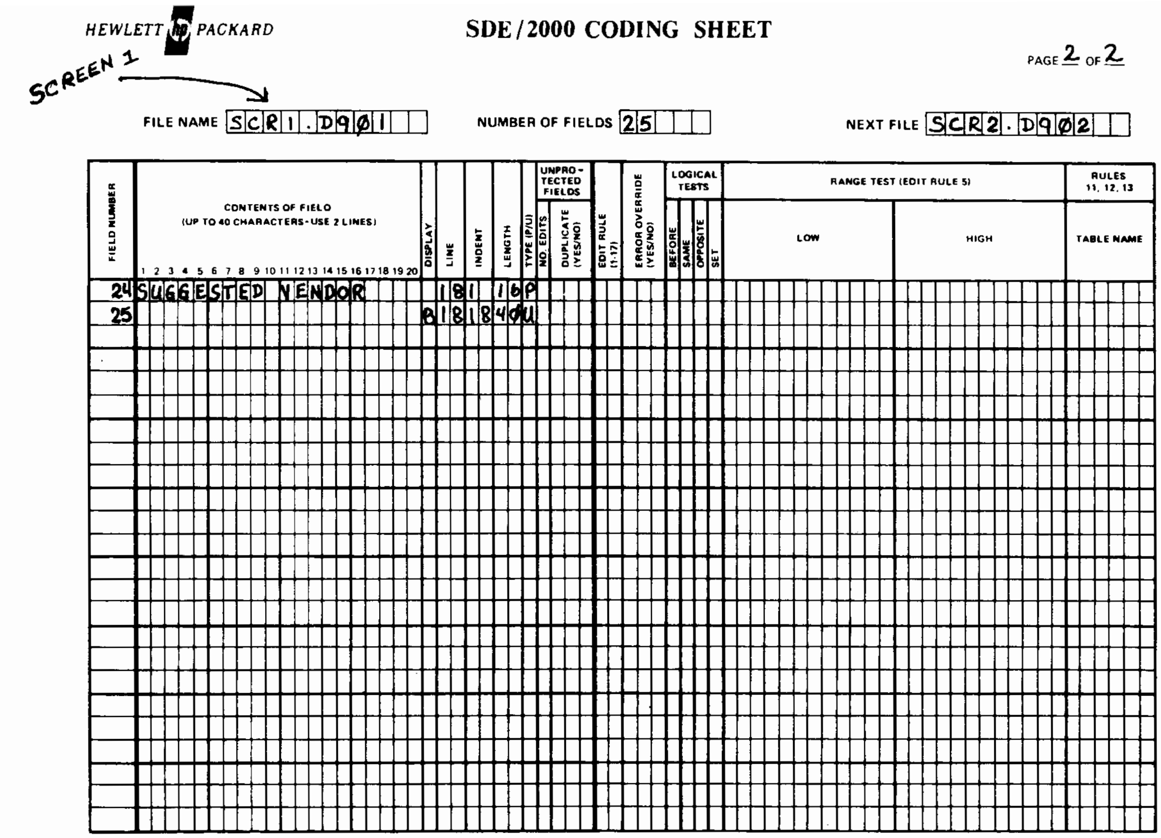
PAGE 1 OF 2

SCREEN 1

FILE NAME SCR1.D9Ø1 NUMBER OF FIELDS 25 NEXT FILE SCR2.D9Ø1

FIELD NUMBER	CONTENTS OF FIELD (UP TO 40 CHARACTERS-USE 2 LINES) 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20	DISPLAY	LINE	INDENT	LENGTH	TYPE (P/U)	UNPRO-TECTED FIELDS		EDIT RULE (1-17)	ERROR OVERRIDE (YES/NO)	LOGICAL TESTS				RANGE TEST (EDIT RULE 5)		RULES 11, 12, 13
							NO. EDITS	DUPLICATE (YES/NO)			BEFORE	SAME	OPPOSITE	SET	LOW	HIGH	TABLE NAME
1	PURCHASE REQUISITION		2	28	26	P											
	ORDER																
2	DATE		3	1	4	P											
3		B	3	6	2	U	/		3								
4	/		3	8	1	P											
5		B	3	9	2	U	/		3								
6	/		3	11	1	P											
7		B	3	12	2	U	/		3								
8	NUMBER		3	67	6	P											
9		B	3	74	6	U	/		3								
10	BILL TO		5	1	7	P											
11		F	5	9	4Ø	U											
12		F	6	9	4Ø	U											
13		F	7	9	4Ø	U											
14		F	8	9	4Ø	U											
15	SHIP TO		11	1	7	P											
16		F	11	9	4Ø	U											
17		F	12	9	4Ø	U											
18		F	13	9	4Ø	U											
19		F	14	9	4Ø	U											
20	SHIP VIA		16	1	8	P											
21		B	16	1Ø	4Ø	U											
22	PRIORITY		16	52	8	P											
23		B	16	61	3	U											

Figure 2-5. Coding Sheet, SCR1 Fields 1 through 23



HEWLETT PACKARD

SDE/2000 CODING SHEET

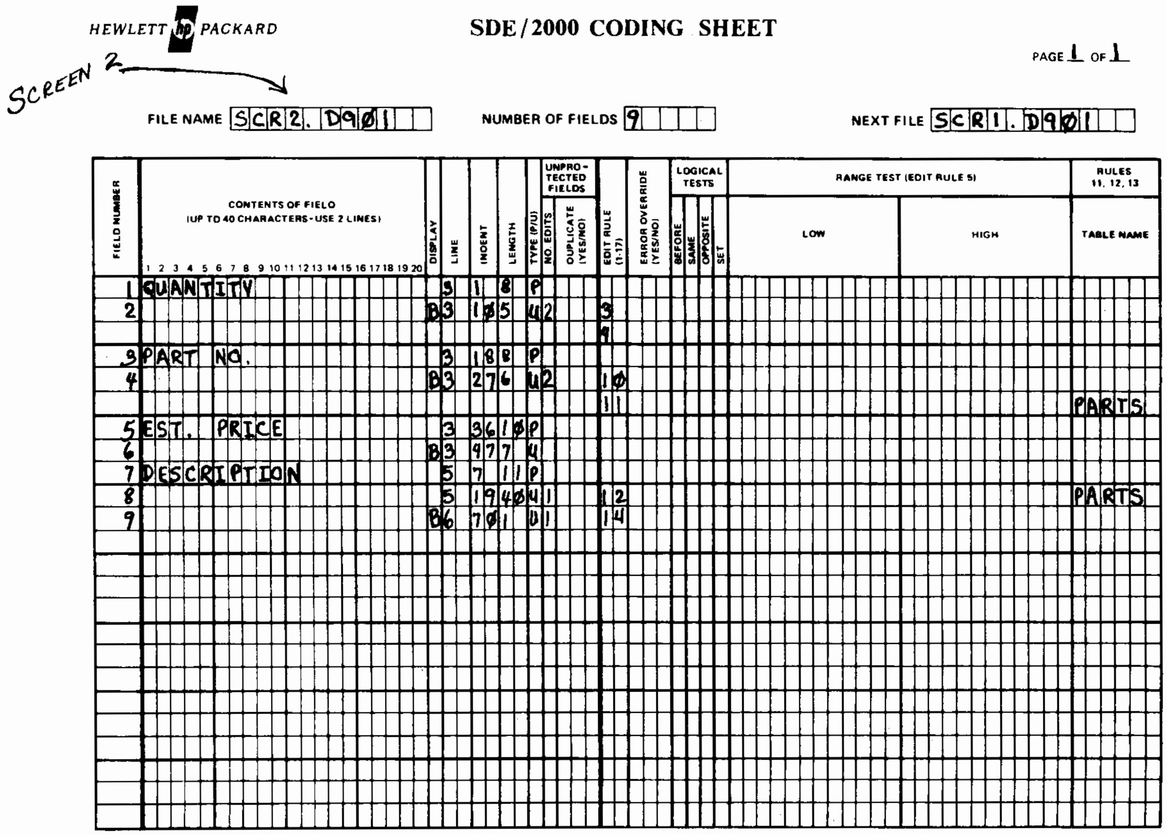
PAGE 2 OF 2

SCREEN 1

FILE NAME SCR1.D9Ø1 NUMBER OF FIELDS 25 NEXT FILE SCR2.D9Ø2

FIELD NUMBER	CONTENTS OF FIELD (UP TO 40 CHARACTERS-USE 2 LINES)	DISPLAY	LINE	INDENT	LENGTH	TYPE (P/U)	UNPROTECTED FIELDS		EDIT RULE (1-17)	ERROR OVERRIDE (YES/NO)	LOGICAL TESTS				RANGE TEST (EDIT RULE 5)		RULES 11, 12, 13
							NO. EDITS	DUPLICATE (YES/NO)			BEFORE	SAME	OPPOSITE	SET	LOW	HIGH	TABLE NAME
24	SUGGESTED VENDOR		18	1	16	P											
25		B	18	18	4Ø	U											

Figure 2-6. Coding Sheet, SCR1 Fields 24 through 25



HEWLETT PACKARD

SDE/2000 CODING SHEET

PAGE 1 OF 1

SCREEN 2

FILE NAME SCR2.D9Ø1 NUMBER OF FIELDS 9 NEXT FILE SCR1.D9Ø1

FIELD NUMBER	CONTENTS OF FIELD (UP TO 40 CHARACTERS-USE 2 LINES)	DISPLAY	LINE	INDENT	LENGTH	TYPE (P/U)	UNPROTECTED FIELDS		EDIT RULE (1-17)	ERROR OVERRIDE (YES/NO)	LOGICAL TESTS				RANGE TEST (EDIT RULE 5)		RULES 11, 12, 13
							NO. EDITS	DUPLICATE (YES/NO)			BEFORE	SAME	OPPOSITE	SET	LOW	HIGH	TABLE NAME
1	QUANTITY		3	1	8	P											
2		B	3	1Ø	5	U	2		3								
									4								
3	PART NO.		3	18	8	P											
4		B	3	27	6	U	2		1Ø								
									11								PARTS
5	EST. PRICE		3	36	1Ø	P											
6		B	3	47	7	U											
7	DESCRIPTION		5	7	11	P											
8			5	19	4Ø	U	1		12								PARTS
9		B	6	7Ø	1	U	1		14								

Figure 2-7. Coding Sheet, SCR2

2-4. GETTING ON THE TERMINAL

Once the form has been coded, the designer logs on to the system and begins defining the form to SDE/2000.

- If you are using the 2640A, the log-on process is as follows:

  Press RETURN and LINE FEED keys
  System will respond with the message, PLEASE LOG IN
  Type HELLO-account,password,2

  where "account" is a Forms Design user account and "password" is the password for that account. The "2" indicates that you are using a 2640A terminal in page mode.

- If you are using the 2644A, the log-on process is as follows:

  Depress the Auto LF button
  Press RETURN key
  System will respond with the message, PLEASE LOG IN
  Release the Auto LF button
  Type HELLO-account,password,2

  where "account" is a Forms Design user account and "password" is the password for that account. The "2" indicates that you are using a 2644A terminal in page mode.

Create any files that are to contain the form description. The size of these files is determined by counting the number of fields to be specified, dividing by 2, and adding 14.

Number of Records = (Fields/2) + 14

For the two screens of the purchase requisition order, you create the necessary files with the commands

```
CREATE-SCR1,26
CREATE-SCR2,18
```

Your screens may access table files containing fields that can be searched and displayed during data entry. You must create any such table files that are used by your form, and they must be at least two records long. Table functions include table searching, replacing screen data with table data and allowing the terminal operator to dynamically select the name of next screen to be displayed.

In our example the part numbers and descriptions are kept in a table file called PARTS. After you enter the part number during data entry, SDE will search PARTS to locate its description and display it on the screen.

The command to create this file is

```
CREATE-PARTS,25
```

The use of tables is explained in detail in Section III. The method for building tables is given at the end of this section.

2-5. EXECUTING SDEDM

After the preliminary work is completed, you are ready to execute the Forms Design program. Type the command

```
EXEC-*SDEDM
```

The first thing you see on the terminal are the words DEPRESS BLOCK MODE BUTTON. During execution of SDEDM the terminal communicates with the computer in block mode, so the button must remain depressed. You will then be presented with a series of screens.

An additional check is made to insure that the 2640A terminal is strapped for *page* block mode. If it is not, the message TERMINAL MUST BE STRAPPED FOR PAGE BLOCK MODE appears, and the program terminates. Refer to Appendix A to convert the terminal to page mode.

2-6. START-UP SCREEN

The first screen you see is shown in figure 2-8. The areas you must fill in are delimited by inverse video blanks (white blocks). The first piece of information you type is the file name that will contain the form description or /E if you want to terminate the program. The file name must be fully qualified (it must include the account number where the file is located) and it must have been previously created. For the purchase requisition order with a screen file in account D901 the entry is

```
SCR1.D901
```

The second entry is the number of field specifications. This is the total number of protected and unprotected fields. For SCR1 the number is 25.

The third entry is the name of the next logical file. In our example the item entries in SCR2 come after the shipping and billing information of SCR1, so the next logical file is

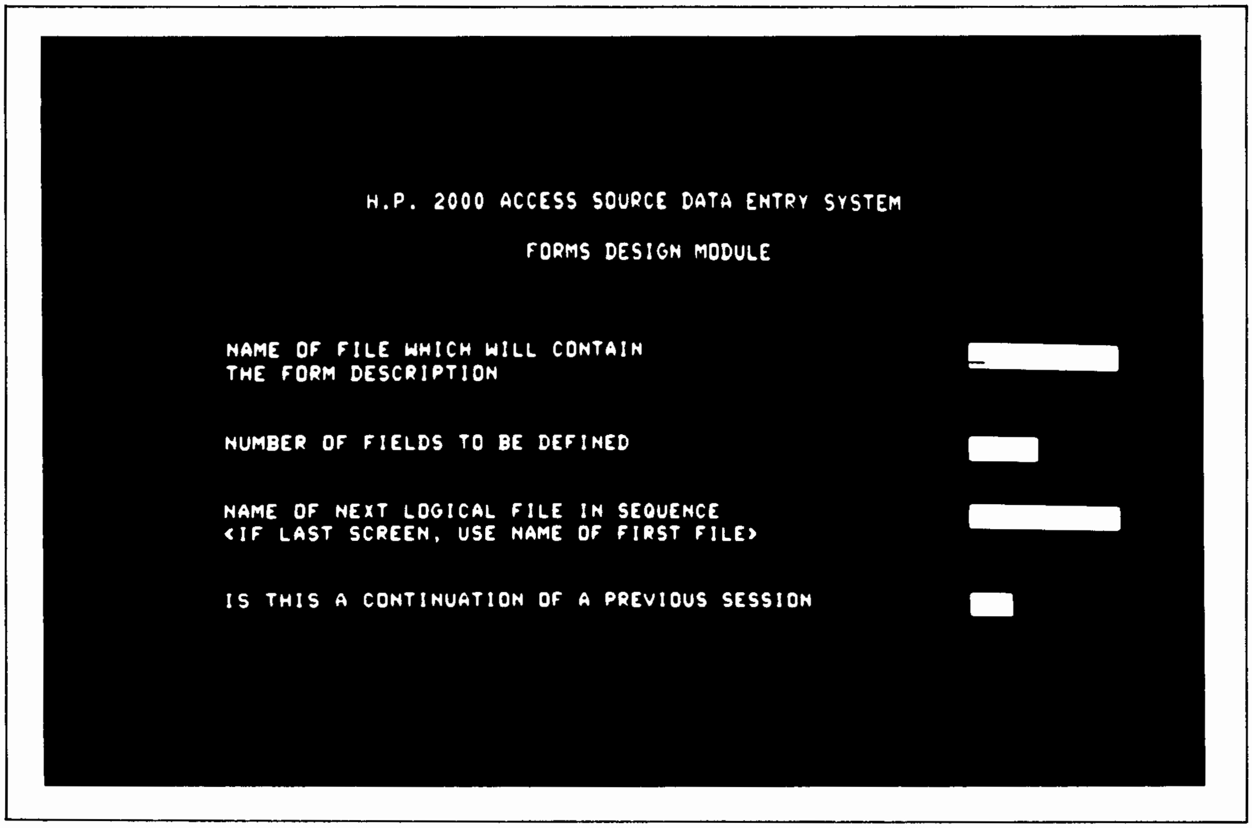
```
SCR2.D901
```

This file must also have been created in your account. When the last screen has been designed, enter the name of the first screen as the next logical file. In our example, after designing the second screen (SCR2), the next logical screen would be SCR1.D901. If there is only one screen in the form, the next logical file is the same as the current screen description file.

The last prompt asks: IS THIS A CONTINUATION OF A PREVIOUS SESSION.

Answer NO if you are just beginning to design the screen. The Field Description screen will appear, allowing you to define the *first* field.

Answer YES if you have previously terminated a session with a /E as the line number. The Field Description screen will appear with the *next* field to be defined, allowing you to start designing where you left off last time.



```
H.P. 2000 ACCESS SOURCE DATA ENTRY SYSTEM

FORMS DESIGN MODULE

NAME OF FILE WHICH WILL CONTAIN
THE FORM DESCRIPTION

NUMBER OF FIELDS TO BE DEFINED

NAME OF NEXT LOGICAL FILE IN SEQUENCE
<IF LAST SCREEN, USE NAME OF FIRST FILE>

IS THIS A CONTINUATION OF A PREVIOUS SESSION
```

Figure 2-8. Start-up Screen

2-7. ENTERING INFORMATION

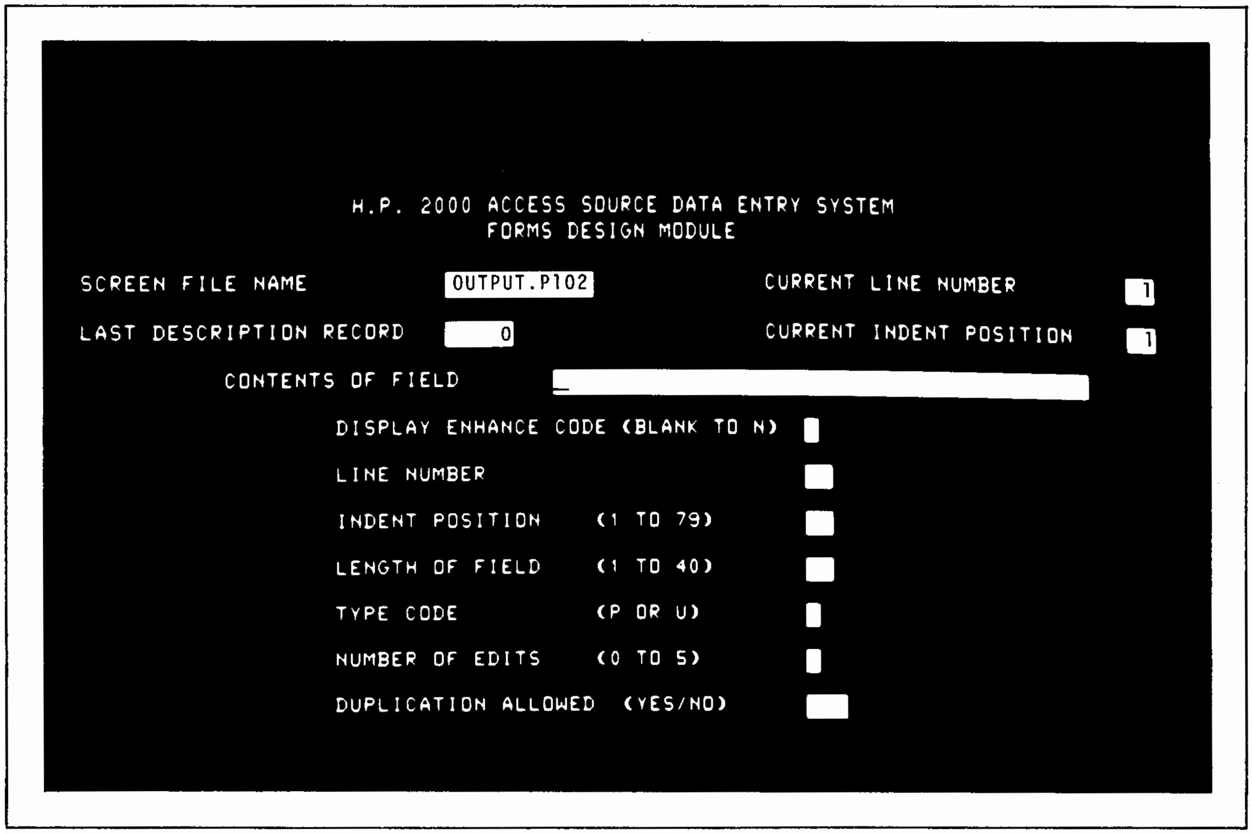
On all the screens, if an answer to a question fills up the input field, the terminal cursor will automatically move to the next field. Otherwise, you may use the TAB key to get from one field to the next. After you have answered all questions, check to make sure they are correct. They can be changed by using the DELETE CHAR key or by positioning the cursor to the field and typing over the old information. For multi-page forms, the NEXT PAGE key can be used to display the next 24 lines of memory and the PREV PAGE key can be used to display the previous 24 lines of memory. When you are satisfied with the screen, depress the ENTER key to send the information to the computer. If SDEDM catches any errors, a blinking message appears. You must retype the incorrect field and depress the ENTER key again before processing can continue.

2-8. FIELD DESCRIPTION SCREEN

After the Start-up screen has been entered, the Field Description screen (figure 2-9) appears on the terminal. This screen will reappear for each field.

The first specification is CONTENTS OF FIELD. Anything typed in this field will appear as text on the forms. An unprotected field usually consists of blanks, but can contain a value that may be overwritten when data is entered.

The second specification, DISPLAY ENHANCE CODE, refers to the 2640A's ability to present characters in Half-Bright, Underline, and Inverse Video display. A blank indicates the field will be displayed in the normal way. Other display code letters are B, D, F, H, J, L, and N, as shown in figure 2-10. Blinking is not allowed since it is used by the Data Entry program to indicate errors.



```
                  H.P. 2000 ACCESS SOURCE DATA ENTRY SYSTEM
                           FORMS DESIGN MODULE

SCREEN FILE NAME         OUTPUT.P102          CURRENT LINE NUMBER        1

LAST DESCRIPTION RECORD            0          CURRENT INDENT POSITION    1

        CONTENTS OF FIELD           _

               DISPLAY ENHANCE CODE (BLANK TO N)

               LINE NUMBER

               INDENT POSITION     (1 TO 79)

               LENGTH OF FIELD     (1 TO 40)

               TYPE CODE           (P OR U)

               NUMBER OF EDITS     (0 TO 5)

               DUPLICATION ALLOWED  (YES/NO)
```

Figure 2-9. Field Description Screen

ENHANCEMENTS	ENHANCEMENT CODES						
	B	D	F	H	J	L	N
HALF BRIGHT				X	X	X	X
UNDERLINE		X	X			X	X
INVERSE VIDEO	X		X		X		X

Figure 2-10. Display Enhancements

The third specification is the line number. The terminal page has 24 lines numbered 0 to 23. Lines 0 and 23 are reserved by the Data Entry program, so 22 lines can be used for each page on a business form. The terminal supports multi-page operation with the NEXT PAGE and PREV PAGE keys. SDEDM allows you to define a form that consists of up to three pages:

page 1 includes lines 1 to 22
page 2 includes lines 25 to 46
page 3 includes lines 49 to 70.

Any of these line numbers are valid for the line number specification, but once a line is passed you cannot go back to it. If you want to stop for a while, type /E as the line number. When you restart you can continue where you left off by answering YES to the continuation question on the Start-up screen.

The next field is the indent position: the column of the line where the field begins. This must be a number between 1 and 79 (inclusive). On any particular line, the indent positions of consecutive fields must be increasing.

The length of the field is typed next. A field may be up to 40 characters long. SDE/2000 does not allow overlapping fields, and the indent position and length must not sum to a number greater than 79.

The sixth specification indicates data type P (protected) or U (unprotected). If the field is for output only, it is protected. The unprotected fields are the ones that will be recorded on disc during data entry.

The NUMBER OF EDITS specification is for unprotected fields only. Placing a value between 0 and 5 (inclusive) will cause the Edit Rule screen to appear that number of times before the Field Description screen appears again.

The DUPLICATION ALLOWED specification is also for unprotected fields only. If duplication is allowed, the information in the field remains when the screen is repeated during data entry. Otherwise the cursor is moved to the beginning of each unprotected field and the erase command is executed before the form is ready for the entry of new data. For duplication, type YES. Otherwise type NO or leave the entry blank. Duplication is useful when the same information (such as the date) appears on many consecutive screens.

After you have answered all the questions, depress the ENTER key as before. If you have made any errors, a blinking message will appear at the bottom of the screen and the data in error will blink. If the first character of the bad input is a blank, a ? will appear. As before, you must correct and reenter the information.

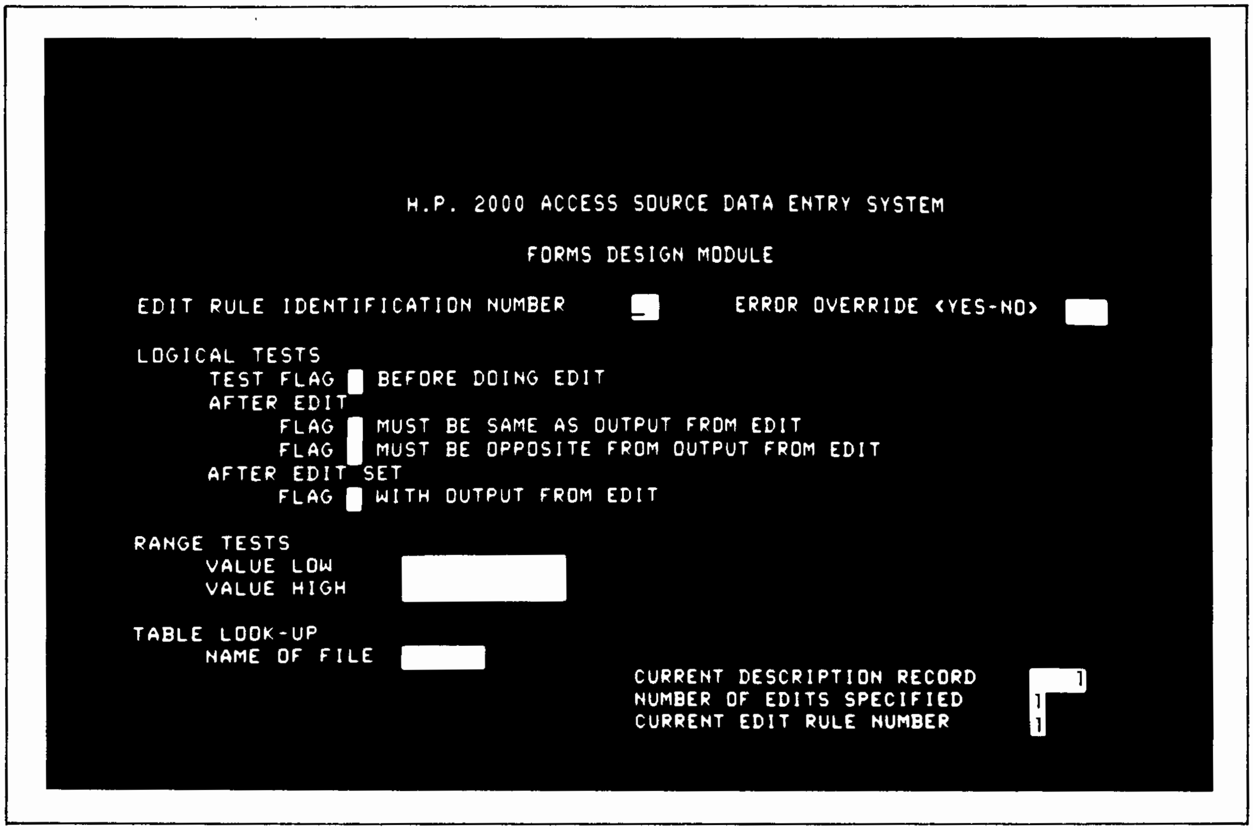
2-9. EDIT RULE SCREEN

You may specify up to five separate edit functions for each unprotected field. Included in the edit rules are routines to check data for errors, manipulate the contents of a 40 character save field, perform operations with external data tables, and perform operations on nine logical flags. Each rule sets an error indicator to 0 or 1. The seventeen edit rules are discussed in detail in Section III. The Edit Rule screen is shown in figure 2-11.

The first piece of information you must enter is the EDIT RULE IDENTIFICATION NUMBER, specifying the rule to be used. For example if you want a field to be checked to make sure it is numeric, you type a 3 or 03 in this field. If you type /E as the rule number you will terminate execution of SDEDM.

The ERROR OVERRIDE specification requires a YES or NO answer. A NO (or blank field) will cause an error message to appear if the indicator is set to 0 by this edit during data entry. A YES instructs the data entry error logic to remain inactive should the result of this edit set the error indicator to 0.

The next section of the screen, labeled LOGICAL TESTS, performs logical operations with the error indicator and the nine logical flags. (The logical flags can be set to a value of 0 or 1.) Each of the four input fields in this section specifies which of the nine logical flags to use in the associated operation. With the first specification you permit the edit only if the specified flag has a value of 1. The second and third specifications enable you to pass the edit only if the error indicator is set to a value that is the same/opposite as the value of the indicated flag. The final logical operation enables you to load the specified flag with the value of the error indicator after the edit is completed.



```
                    H.P. 2000 ACCESS SOURCE DATA ENTRY SYSTEM

                              FORMS DESIGN MODULE

     EDIT RULE IDENTIFICATION NUMBER            ERROR OVERRIDE <YES-NO>

     LOGICAL TESTS
          TEST FLAG   BEFORE DOING EDIT
          AFTER EDIT
               FLAG   MUST BE SAME AS OUTPUT FROM EDIT
               FLAG   MUST BE OPPOSITE FROM OUTPUT FROM EDIT
          AFTER EDIT SET
               FLAG   WITH OUTPUT FROM EDIT

     RANGE TESTS
          VALUE LOW
          VALUE HIGH

     TABLE LOOK-UP
          NAME OF FILE
                                   CURRENT DESCRIPTION RECORD        1
                                   NUMBER OF EDITS SPECIFIED       1
                                   CURRENT EDIT RULE NUMBER        1
```

Figure 2-11. Edit Rule Screen

An example of these logical tests is to check a field that has two possible values. Suppose a particular field must either be blank or contain the character N. You can 1) perform a range check for a blank with the error override on and set flag 1 with the result, then 2) perform a range check for N. If both range tests fail, the flag and the error indicator will be the same, so you should require the result of the second edit to be the opposite of flag 1. This example is coded in field 6 of figure C-15.

Figure 2-12 diagrams the logical tests in an edit call.

The RANGE TEST values are used with edit rule five, and are not used for any other edit. The final input field on the screen, NAME OF FILE, is used only with the table edit rules, numbers 11, 12 and 13. These entries will be explained in the next section.

After all the necessary information has been typed (in some cases only the first entry is needed) you enter it in the usual way by depressing the ENTER key.

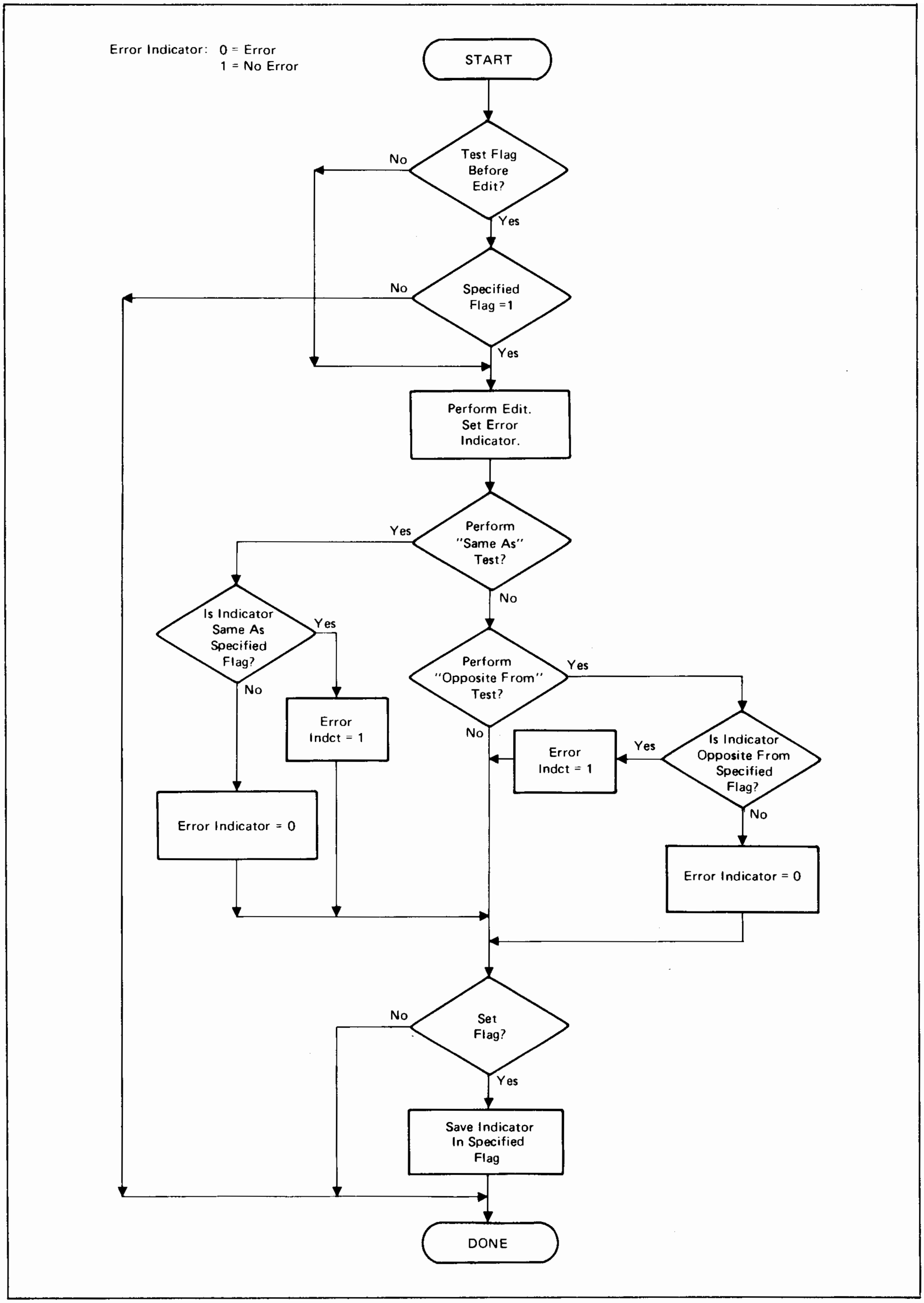

Error Indicator: 0 = Error
1 = No Error
START
Test Flag Before Edit?
No
Yes
Specified Flag =1
No
Yes
Perform Edit. Set Error Indicator.
Perform "Same As" Test?
Yes
No
Is Indicator Same As Specified Flag?
Yes
No
Error Indct = 1
Error Indicator = 0
Perform "Opposite From" Test?
Yes
No
Error Indct = 1
Yes
Is Indicator Opposite From Specified Flag?
No
Error Indicator = 0
Set Flag?
No
Yes
Save Indicator In Specified Flag
DONE

Figure 2-12. Logical Tests During Edit

2-10. CYCLE OF SDEDM SCREENS

The Edit Rule screen reappears until all the edits are specified or until you type a /E as the line number. The cycle of the screens is diagrammed in figure 2-13.

For multiple screen forms you execute SDEDM to design each screen. The Next File field on the Start-up screen allows you to link multiple screens together.

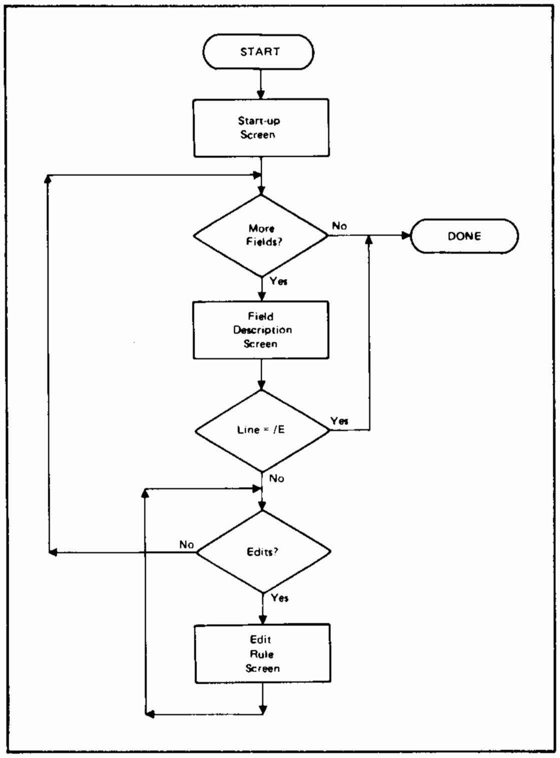


Figure 2-13. Cycle of SDEDM Screens

2-11. CONTROL OF SCREENS DURING DATA ENTRY

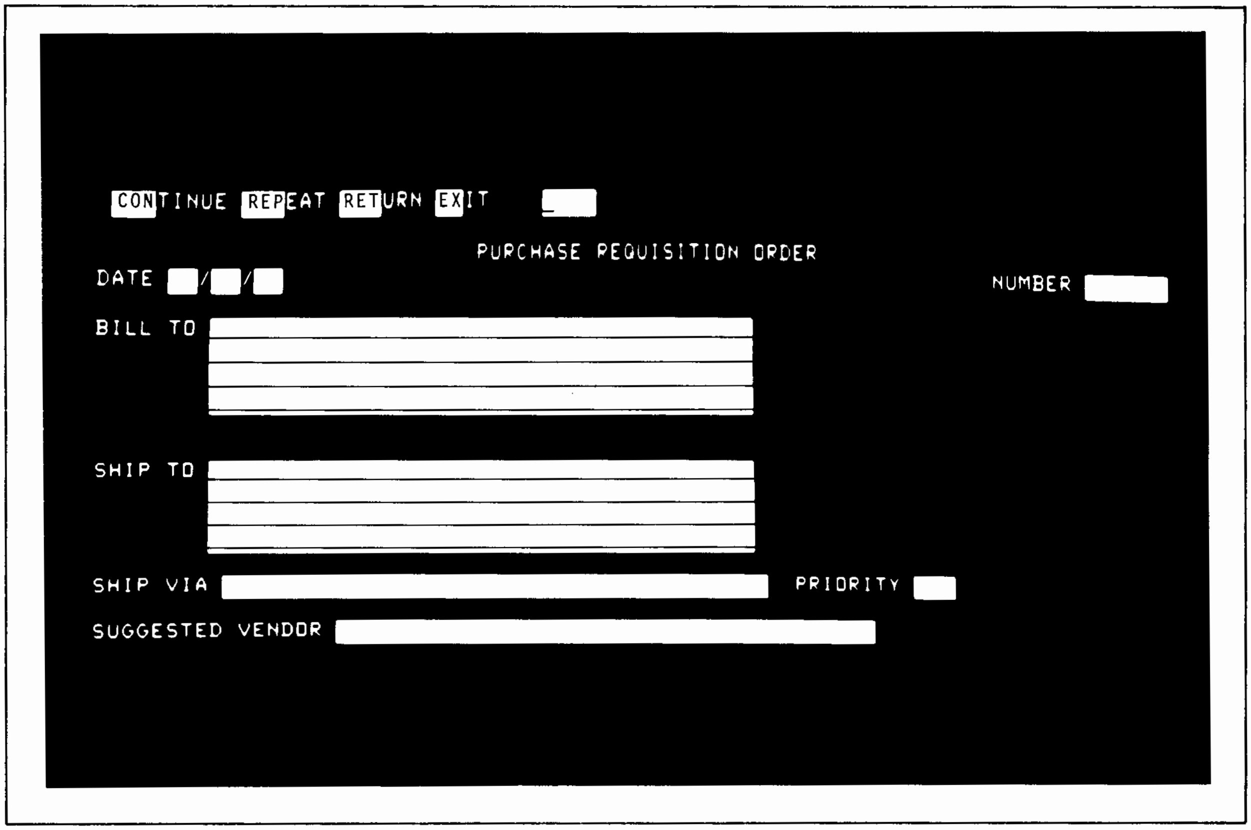
When your form is used during data entry, the person who enters the data controls the display of screens by the function options that appear on line 0. (The functions are discussed in subsection 4-5.) For example, after filling the first screen of the purchase requisition order (SCR1), he would CONTINUE to the item description screen (SCR2), then REPEAT this screen until all the items on the requisition were entered. Then he could CONTINUE with the next order until all were done, at which time he would EXIT.

2-12. EXPLANATION OF PURCHASE REQUISITION ORDER EDITS

After you understand the meaning of each field on the Field Description and Edit Rule screens, the reasons behind the entries on the SCR1 and SCR2 coding sheets (figures 2-5 through 2-7) should be clear. On SCR1, note that the date requires fields 2 through 7: one for the word DATE, two for the character /, and three for numeric input. The only edit rule specified, number 3, checks the date and number fields to insure they are numeric. Duplication is not allowed for any field because SCR1 will never be repeated; it is always followed by SCR2.

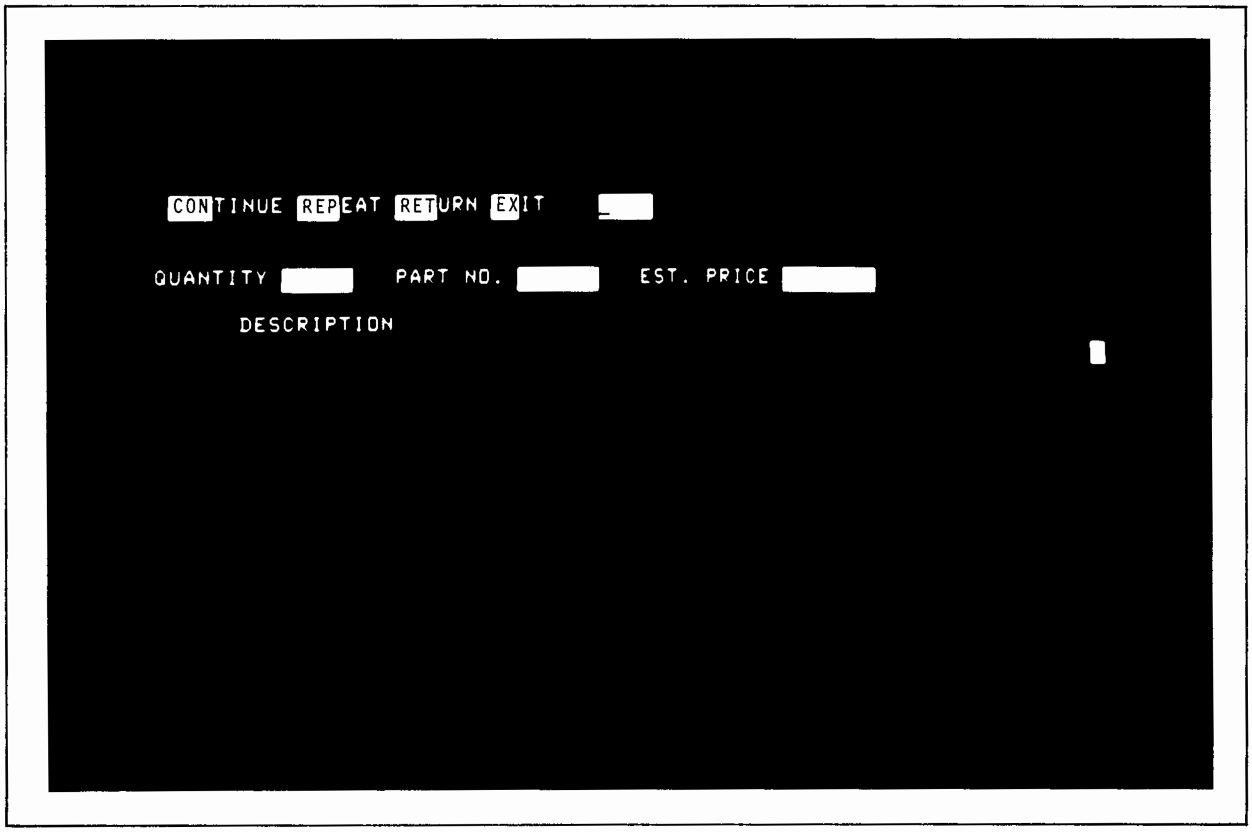
SCR2 requires more edits, including table functions with the part number table PARTS. The part number (field 4) is saved by edit rule 10, then checked by rule 11. The description is taken from the table and put in field 8 by rule 12. The last field uses rule 14 to hold the screen until it can be checked. When the user is sure the information is correct he types a blank in that field.

Most of the input fields use the B display enhancement (inverse video) because this delimits the field well. The shipping and billing entries use F because the underlining helps separate the four lines. The two screens, SCR1 and SCR2, are shown in figures 2-14 and 2-15.



```
CONTINUE REPEAT RETURN EXIT     _

                          PURCHASE REQUISITION ORDER
DATE   /  /                                                    NUMBER
BILL TO

SHIP TO

SHIP VIA                                          PRIORITY
SUGGESTED VENDOR
```

Figure 2-14. SCR1 Displayed on 2640A Terminal



```
CONTINUE REPEAT RETURN EXIT     _

QUANTITY        PART NO.        EST. PRICE
      DESCRIPTION
```

Figure 2-15. SCR2 Displayed on 2640A Terminal

2-13. COMPLETION OF SDEDM

After all the fields are specified or when you type /E for the line number, SDEDM will terminate. A slight pause occurs, after which the message, FORM IS NOW READY FOR USE will appear. If an error occurs, one of the following messages appear:

```
SCREEN FILE NOT ACCESSIBLE FOR PREPARATION
SCREEN FILE OVERFLOW - INCREASE SIZE OF THE SCREEN FILE
CAN'T PREPARE FILE - NUMERIC SPACE FULL
CAN'T PREPARE FILE - CHARACTER SPACE FULL
```

When the form is ready for use, you must take the 2640A out of block mode before any further commands can be issued to the computer. To do this, release the BLOCK MODE key. Upon completion of SDEDM the file you specified contains records describing each field of your form.

2-14. EDITING THE FORM AFTER IT IS SAVED

After you have described your business form to SDE/2000, you can display, list, or modify it on the terminal by executing the Screen File Utility program. To do this, type the command

```
EXEC-*SDEUTL
```

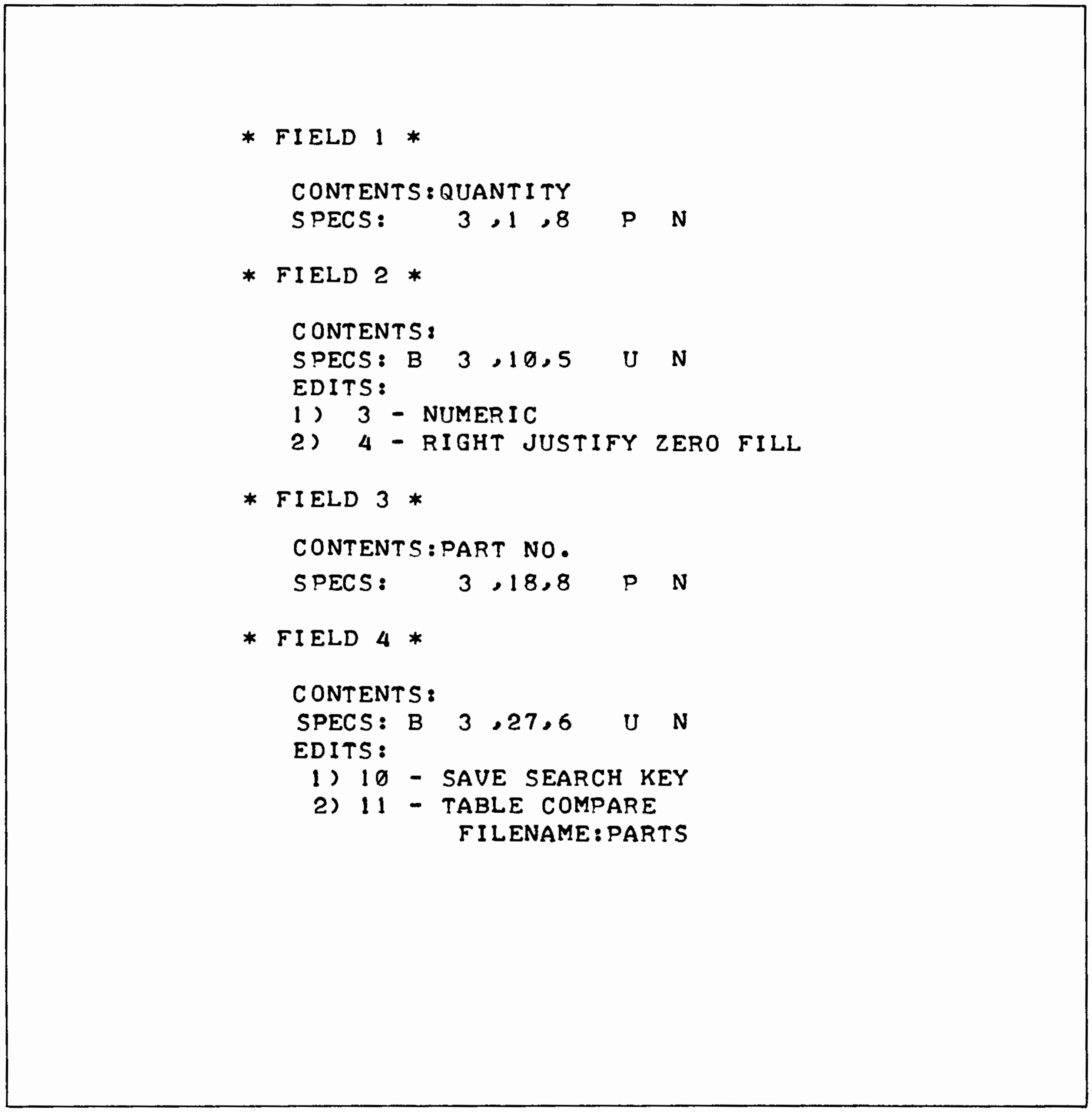
SDEUTL is an interactive program that does *not* run in page mode. The first question you must answer is the name of the file containing the screen description. SDEUTL will then ask whether you want to display, list, or modify the file. To specify one of these three functions, type a D, L, or M.

If you type D, SDEUTL will draw your form on the terminal, fill the unprotected fields with asterisks, and wait for you to depress the RETURN key. After you depress RETURN, the DISPLAY, LIST, OR MODIFY? prompt will reappear.

The LIST option gives you a listing of your screen file as it is recorded on disc. SDEUTL takes advantage of the fact that a 2000 Access user may channel his output to the line printer, and asks if you want the listing directed to the terminal or line printer. To get the listing on the terminal, simply respond TERM. To get the listing on the line printer, respond with the ASCII file name assigned to the line printer in your account. In order to use the line printer the designer's account must be given LP capability and an ASCII file must be created in his account for a line printer on the system.

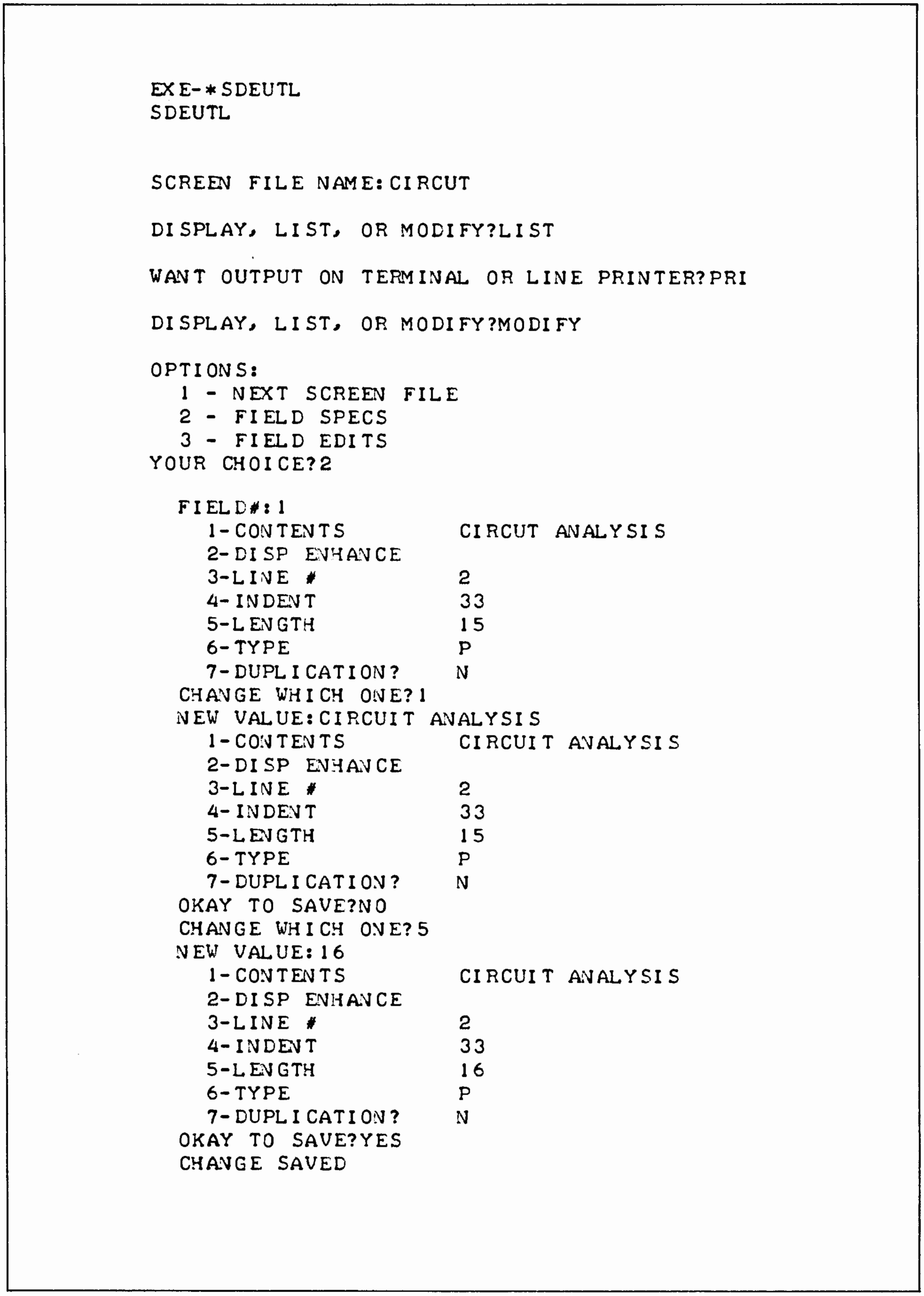
Figure 2-16 shows a partial listing of the file SCR2 which you can compare to the coding sheet in figure 2-7.

If you type M in response to the question, you must then choose what you want to modify: the name of the next screen file, field specification, or field edits. If you change the next screen file, you must type the fully qualified name of the new file. If you choose either of the last two options, you must specify the number of the field you want to change. Figure 2-17 shows how to modify field specifications. In that example the file contains a misspelled word in field 1. In addition to correcting the spelling error (from CIRCUT to CIRCUIT) the field must be lengthened from 15 to 16.



```
* FIELD 1 *

   CONTENTS:QUANTITY
   SPECS:     3 ,1 ,8    P   N

* FIELD 2 *

   CONTENTS:
   SPECS: B   3 ,1Ø,5    U   N
   EDITS:
   1)  3 - NUMERIC
   2)  4 - RIGHT JUSTIFY ZERO FILL

* FIELD 3 *

   CONTENTS:PART NO.
   SPECS:     3 ,18,8    P   N

* FIELD 4 *

   CONTENTS:
   SPECS: B   3 ,27,6    U   N
   EDITS:
    1) 1Ø - SAVE SEARCH KEY
    2) 11 - TABLE COMPARE
             FILENAME:PARTS
```

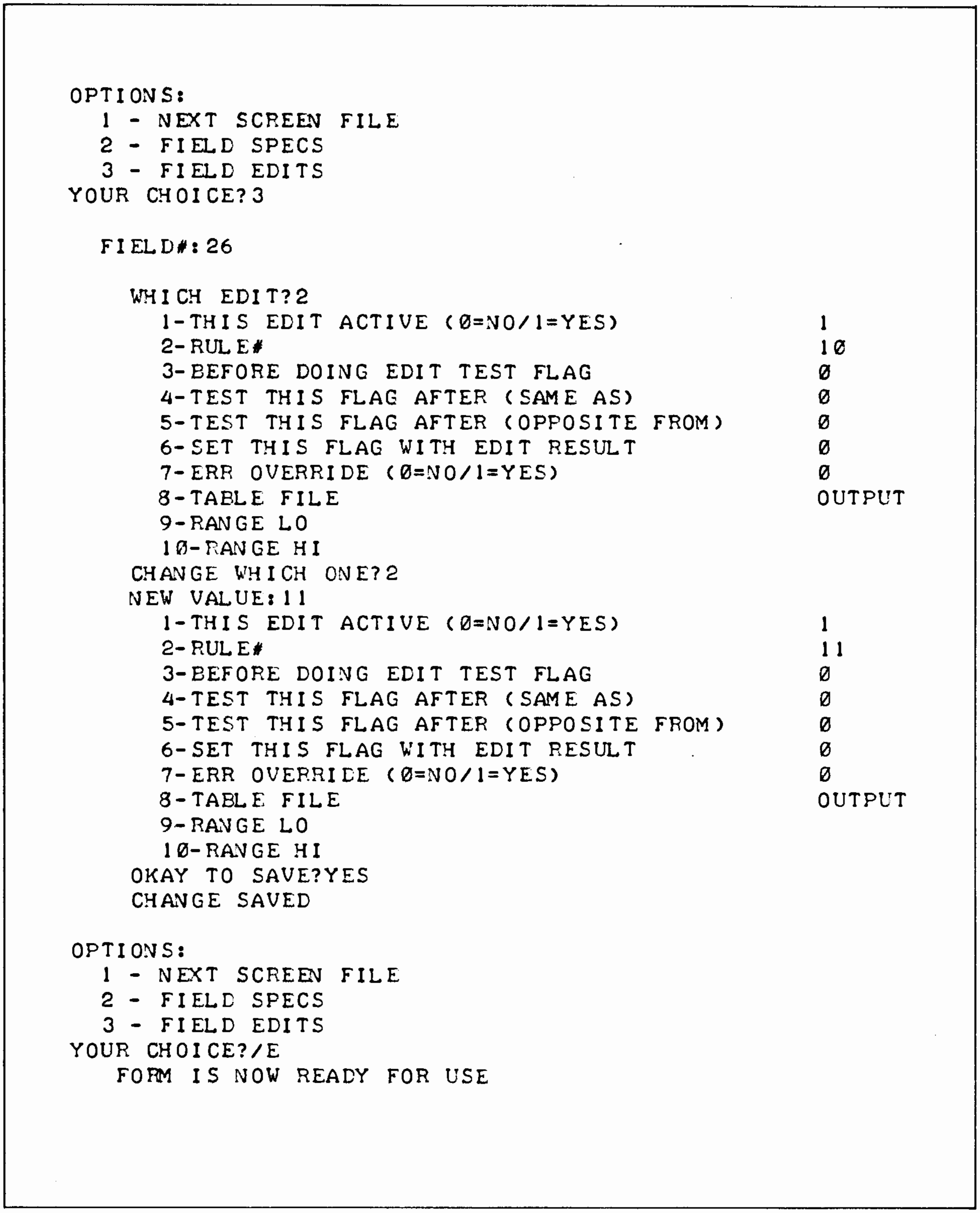
Figure 2-16. Partial Listing of SCR2



```
EXE-*SDEUTL
SDEUTL


SCREEN FILE NAME:CIRCUT

DISPLAY, LIST, OR MODIFY?LIST

WANT OUTPUT ON TERMINAL OR LINE PRINTER?PRI

DISPLAY, LIST, OR MODIFY?MODIFY

OPTIONS:
  1 - NEXT SCREEN FILE
  2 - FIELD SPECS
  3 - FIELD EDITS
YOUR CHOICE?2

  FIELD#:1
    1-CONTENTS          CIRCUT ANALYSIS
    2-DISP ENHANCE
    3-LINE #            2
    4-INDENT            33
    5-LENGTH            15
    6-TYPE              P
    7-DUPLICATION?      N
  CHANGE WHICH ONE?1
  NEW VALUE:CIRCUIT ANALYSIS
    1-CONTENTS          CIRCUIT ANALYSIS
    2-DISP ENHANCE
    3-LINE #            2
    4-INDENT            33
    5-LENGTH            15
    6-TYPE              P
    7-DUPLICATION?      N
  OKAY TO SAVE?NO
  CHANGE WHICH ONE?5
  NEW VALUE:16
    1-CONTENTS          CIRCUIT ANALYSIS
    2-DISP ENHANCE
    3-LINE #            2
    4-INDENT            33
    5-LENGTH            16
    6-TYPE              P
    7-DUPLICATION?      N
  OKAY TO SAVE?YES
  CHANGE SAVED
```

Figure 2-17. Changing Field Specifications

Figure 2-18 shows a continuation of the session in which the second edit in field 26 is changed. The original edit used rule 10 when it should have used rule 11.



```
OPTIONS:
  1 - NEXT SCREEN FILE
  2 - FIELD SPECS
  3 - FIELD EDITS
YOUR CHOICE?3

  FIELD#:26

    WHICH EDIT?2
      1-THIS EDIT ACTIVE (Ø=NO/1=YES)                    1
      2-RULE#                                            1Ø
      3-BEFORE DOING EDIT TEST FLAG                      Ø
      4-TEST THIS FLAG AFTER (SAME AS)                   Ø
      5-TEST THIS FLAG AFTER (OPPOSITE FROM)             Ø
      6-SET THIS FLAG WITH EDIT RESULT                   Ø
      7-ERR OVERRIDE (Ø=NO/1=YES)                        Ø
      8-TABLE FILE                                       OUTPUT
      9-RANGE LO
      1Ø-RANGE HI
    CHANGE WHICH ONE?2
    NEW VALUE:11
      1-THIS EDIT ACTIVE (Ø=NO/1=YES)                    1
      2-RULE#                                            11
      3-BEFORE DOING EDIT TEST FLAG                      Ø
      4-TEST THIS FLAG AFTER (SAME AS)                   Ø
      5-TEST THIS FLAG AFTER (OPPOSITE FROM)             Ø
      6-SET THIS FLAG WITH EDIT RESULT                   Ø
      7-ERR OVERRIDE (Ø=NO/1=YES)                        Ø
      8-TABLE FILE                                       OUTPUT
      9-RANGE LO
      1Ø-RANGE HI
    OKAY TO SAVE?YES
    CHANGE SAVED

OPTIONS:
  1 - NEXT SCREEN FILE
  2 - FIELD SPECS
  3 - FIELD EDITS
YOUR CHOICE?/E
   FORM IS NOW READY FOR USE
```

Figure 2-18. Changing Field Edit

For both of these options you must answer YES to the OKAY TO SAVE? question when all the modifications for the field specifications or edits are complete. If you press the RETURN key without typing anything, the changes will not be recorded and you will be presented with the OPTIONS question again. If you press RETURN without typing anything in response to the OPTIONS question, SDEUTL will print the DISPLAY, LIST, OR MODIFY? question.

When you modify your screen description file, it is your responsibility to enter proper corrections. SDEUTL does not check your new entries to see if they are acceptable. For example, if you type a number greater than 17 as an edit rule number it will be recorded on disc and a fatal error will occur during data entry.

You should also note that new fields cannot be inserted by SDEUTL. Only existing ones can be modified.

When you are satisfied with the form and have exited the program, use the 2000 Access PROTECT command to protect the file. In the purchase requisition order example,

```
PROTECT-SCR1
```

protects the file containing the first screen.

2-15. BUILDING TABLE FILES

If your form uses table files, you must enter the key and data fields before execution of the Data Entry program. The format and use of table files are discussed in section 3-14. You can make entries (or deletions) in your tables by executing the Table Utility program. To do this, type

```
EXEC-*SDETBL
```

SDETBL is an interactive program that does *not* run in block mode. The first question you must answer is

```
TABLE FILE NAME?
```

Type the name of a table file you have already created in your account. SDETBL will then print

```
ADD, REMOVE, OR LIST?
```

You answer by typing one of the three function names.

If you type ADD, you will be prompted for key and data values until you terminate the program by typing /E. SDETBL will not permit you to add duplicate keys, nor will it accept key or data values longer than 40 characters. If you try to add more entries than your file can hold, SDETBL will print FILE FULL and stop.

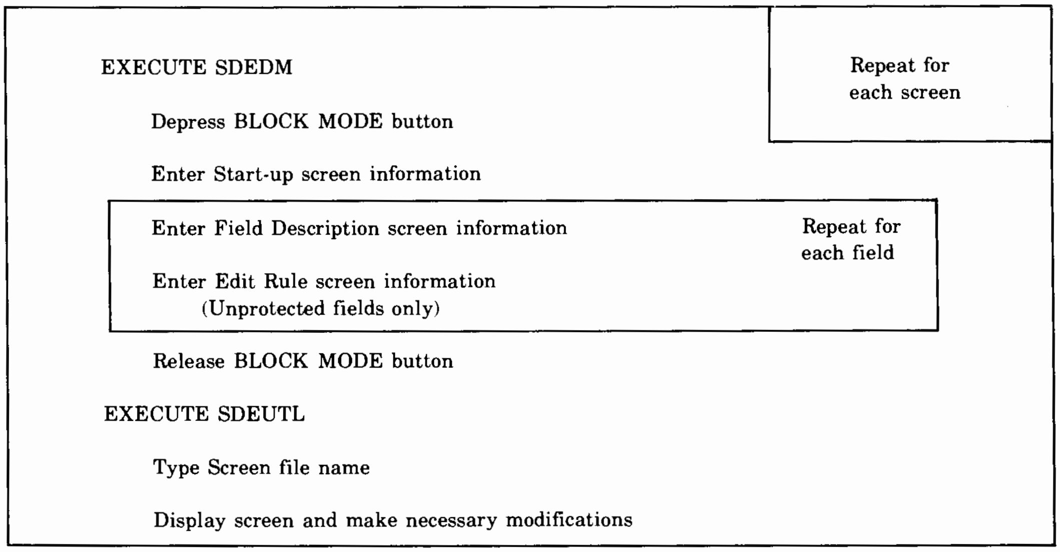
If you choose the REMOVE function, SDETBL will prompt you for key values of table entries you want removed from the table. You can remove as many as needed, terminating the program with a response of /E.

If you choose the third function, LIST, SDETBL will print a list of the key and data fields in your table.

Table files must be protected before Data Entry and must be at least two records long.

2-16. QUICK REFERENCE TO FORMS DESIGN AND DISPLAY

Log on to 2000 Access with 2640A terminal strapped for page mode. CREATE any files that will store forms or tables.



EXECUTE SDEDM

Depress BLOCK MODE button

Enter Start-up screen information

Enter Field Description screen information

Enter Edit Rule screen information
(Unprotected fields only)

Repeat for each field

Release BLOCK MODE button

EXECUTE SDEUTL

Type Screen file name

Display screen and make necessary modifications

Repeat for each screen

PROTECT Screen files

If any Table files were used:

EXECUTE SDETBL

Enter Key and Data fields

PROTECT Table files

Log off

EXAMPLE:

```
HELLO-D901,PASWRD,2
CREATE-SCR1,13
CREATE-SCR2,6
CREATE-PARTS,25
EXEC-*SDEDM
     (Design SCR1)
EXEC-*SDEUTL
     (Display SCR1)
EXEC-*SDEDM
     (Design SCR2)
EXEC-*SDEUTL
     (Display SCR2)
PROTECT-SCR1
PROTECT-SCR2
EXEC-*SDETBL
     (Enter information for PARTS)
PROTECT-PARTS
BYE
```

2-17. VALID INPUT VALUES

START-UP SCREEN

NAME OF FILE WHICH WILL CONTAIN THE FORM DESCRIPTION	Fully qualified name of any file in youraccount or /E.
NUMBER OF FIELDS	0 to 99999
NAME OF NEXT LOGICAL FILE IN SEQUENCE	Fully qualified name of any file in your account or /E.
CONTINUATION	YES, NO

FIELD DESCRIPTION SCREEN

CONTENTS OF FIELD	Any 40 characters	
DISPLAY ENHANCE CODE	B,D,F,H,J,L,N (figure 2-10) or blank	
LINE	1-22 (Page 1), 25-46 (Page 2), 49-70 (Page 3), /E (to stop)	
INDENT POSITION	1 to 79	
LENGTH	1 to 40 (Indent + Length must be less than 80)	
TYPE CODE	P (Protected) or U (Unprotected)	
NUMBER OF EDITS	0 to 5 or blank	Unprotected Fields Only
DUPLICATION ALLOWED	YES, NO, or blank	

EDIT RULE SCREEN

EDIT RULE IDENTIFICATION NO.	1 to 17, /E	
ERROR OVERRIDE	YES, NO, or blank	
TEST FLAG BEFORE EDIT	1 to 9	
FLAG MUST BE SAME	1 to 9	
FLAG MUST BE OPPOSITE	1 to 9	
AFTER EDIT SET FLAG	1 to 9	
VALUE LOW	Any 12 Characters, Left Justified	Edit 5
VALUE HIGH	Any 12 Characters, Left Justified	
NAME OF FILE	Any File in Your Account (Edits 11, 12, 13)	

SDE/2000 EDIT RULES

SECTION III

This section describes SDE/2000 edit rules in detail and explains the relationships between them. Each rule performs its function on the associated uprotected field before it is written to disc.

3-1. RULE 1: ALPHANUMERIC FIELD CHECK

Check each character in the field and accept only the letters A to Z, the numbers 0 to 9, and the space character.

Error Indicator: 0 - Unacceptable character found
1 - All characters alphanumeric

3-2. RULE 2: ALPHABETIC FIELD CHECK

Check each character in the field and accept only the letters A to Z and the space character.

Error Indicator: 0 - Unacceptable character found
1 - All characters alphabetic

3-3. RULE 3: NUMERIC FIELD CHECK

Check each character in the field and accept only the numbers 0 to 9 and the space character. Embedded blanks are not allowed. The number may include a decimal point.

Error Indicator: 0 - Unacceptable character found
1 - All characters numeric

3-4. RULE 4: RIGHT JUSTIFY ZERO FILL

Move the contents of the field to the right end, then replace leading blanks with zeroes.

Error Indicator: Always set to 1

3-5. RULE 5: RANGE CHECK

Check the field to see if it falls within a given range. This edit rule uses the two RANGE TEST values from the Edit Rule screen (figure 2-11). The field must be greater than or equal to VALUE LOW and less than or equal to VALUE HIGH. The twelve character values must be left justified. They are compared as ASCII strings.

Error Indicator: 0 - Unprotected field out of range
1 - Unprotected field within range

3-6. RELATIONSHIP BETWEEN RULES 4 AND 5

Many times you will want to check numeric fields to see if they fall within a certain range. If a field is three characters long and accepts values from 1 to 999, you will want to use rule 4 (Right Justify Zero Fill) before doing the range check with rule 5. Then there is no difference between 1, 01, and 001.

3-7. MODULO 11 CHECK DIGIT

During the entry of numeric data, transcription errors can occur. Digits of a number may be transposed or typed incorrectly. A check digit is used with numeric fields to assure accuracy whenever the field is transcribed. Check digits can be generated by several methods, but SDE/2000 uses a series of calculations that produces what is known as the Modulo 11 Check Digit. The result is a number from 0 to 10. Input fields requiring a check digit of 10 cannot be used and will not be passed by edit rules 6 and 7. If the digits in a number are transposed or incorrectly typed, the Modulo 11 Check Digit will change.

3-8. RULE 6: MODULO 11 CREATE

Calculate the Modulo 11 Check Digit for the numeric input field and append it to the end. There must be room for the check digit, so the size of the field must be one larger than the maximum number of digits. All characters appearing in the field must be digits except the rightmost character, which must be a blank.

3-9. RULE 7: MODULO 11 VERIFY

Verify that the last character in the field is the Modulo 11 Check Digit. All characters in the field must be digits. This edit will usually be preceded by rules 3 (Numeric Field Check) and 4 (Right Justify Zero Fill).

One use for this edit is to verify that a part number has been typed correctly. If a digit is left out, or if two digits are transposed, the Modulo 11 Check Digit will not pass.

3-10. SDE/2000 SAVE AREA

The Save Area is a 40 character string variable which is given a value by rule 8 and displayed on the screen by rule 9. This area provides a way to transfer information between screens. For example, an account number may be saved from the first screen with rule 8 and displayed on the second screen with rule 9. You must be sure the source field is no larger than the destination, or an SDE/2000 system error will occur.

3-11. RULE 8: SAVE CONTENTS OF INPUT FIELD

Copy the field into the 40 character save area.

Error Indicator: Always set to 1.

3-12. RULE 9: DISPLAY CONTENTS OF SAVE AREA

Move the Save Area to the specified field.

Error Indicator: Always set to 1

If the save field is larger than the destination field, it is truncated. Since this edit is performed when the form is drawn, no testing or setting of the 9 available indicators is done.

3-13. RULE 10: SAVE SEARCH KEY

Copy the input field into the Search Key save area. This must be done before a Table Compare, Table Replace, or Screen Replace.

Error Indicator: Always set to 1

3-14. FORMAT OF TABLE FILES

Edit rules 11, 12, and 13 use the file specified by the last entry on the Edit Rule screen (TABLE LOOK-UP NAME OF FILE). Files containing tables for SDE/2000 have the following format: a number followed by two 40-character strings. The first string contains the key and the second contains the data field. The table key values may be searched and compared to the Search Key, and the associated table data field may be displayed on the screen by edit rule 12. Table files must be set up and protected before data entry time. They must be at least two records long, and they must be in the same account as Screen Description files.

3-15. RULE 11: TABLE COMPARE

Sequentially search the file to find a table key matching the Search Key.

Error Indicator: 0 - Either
a. Requested file not on system
b. No match found within file
1 - Match found in file

3-16. RULE 12: TABLE REPLACE

Sequentially read the specified file to find a table key matching the Search Key. When the match is found, replace the unprotected field with the table data field.

Error Indicator: 0 - Either
    a. Requested file not on system
    b. No match found within file
  1 - Match found in file

If an error occurs with this edit during data entry, the Search Key cannot be changed and processing cannot continue. Specify edit rule 11 on the field which contained the Search Key so that it will be checked while it can be changed.

3-17. USING TABLE RULES

In the purchase requisition order example of Section II, we used a table with part numbers as keys and descriptions as data. The number field is first placed in the Search Key area with edit rule 10. Then its validity is checked by rule 11. Finally, edit rule 12 is performed on the description field, placing the description on the screen.

The order of these rules is very important. The Search Key must be saved first, then it should be checked to see if it is in the file. Only then is it safe to perform a Table Replace.

You can speed the sequential searches by putting the most used items at the beginning of a table.

3-18. RULE 13: SCREEN NAME REPLACE

Sequentially read the specified table to find a record key matching the Search Key. When the match is found, replace the name of the file containing the next screen with the data field of the disc record. The file names in the table must be fully qualified.

This rule provides a way to dynamically change the order in which screens appear. For example, suppose the table DEPTS has the following format:

KEY	DATA
1	DEPT1.D901
2	DEPT2.D901
3	DEPT3.D901
⋮	⋮

where DEPT1, DEPT2 . . . are screen descriptions held by account D901. Then you could have a terminal operator type a number indicating the department for which he is entering data. That department's screen would appear on the terminal next. After using edit rules 10 and 11 to save the number in the Search Key and verify that it is in the table, you can use rule 13 to bring the appropriate screen to the terminal.

Error Indicator: 0 - Either
a. Requested file not on system
b. No match found within file
1 - Match found within file

3-19. RULE 14: SCREEN HOLD/REJECT

Examine the first character of the field for a blank, H, or R. Take the following actions:

H (HOLD): Wait for another ENTER from the terminal.

R (REJECT): Position the cursor to the first input field and allow any data on the screen to be reentered.

∧ (Blank - No Action): Return with no action.

This rule allows the user to hold the form on the screen so it can be checked before data is recorded. Processing cannot continue until a blank is typed in the field.

Error Indicator: Always set to 1

3-20. RULE 15: AND FUNCTION

Perform the logical AND function on logical flags 1 and 2. Place the result in the Error Indicator. The Error Indicator is set to 1 only when both Flag 1 and Flag 2 are 1.

Flag 1	1	1	0	0
Flag 2	1	0	1	0
Error Indicator	1	0	0	0

3-21. RULE 16: OR FUNCTION

Perform the logical OR function on logical flags 1 and 2. Place the result in the Error Indicator. The Error Indicator is set to 1 whenever Flag 1 or Flag 2 is 1.

Flag 1	1	1	0	0
Flag 2	1	0	1	0
Error Indicator	1	1	1	0

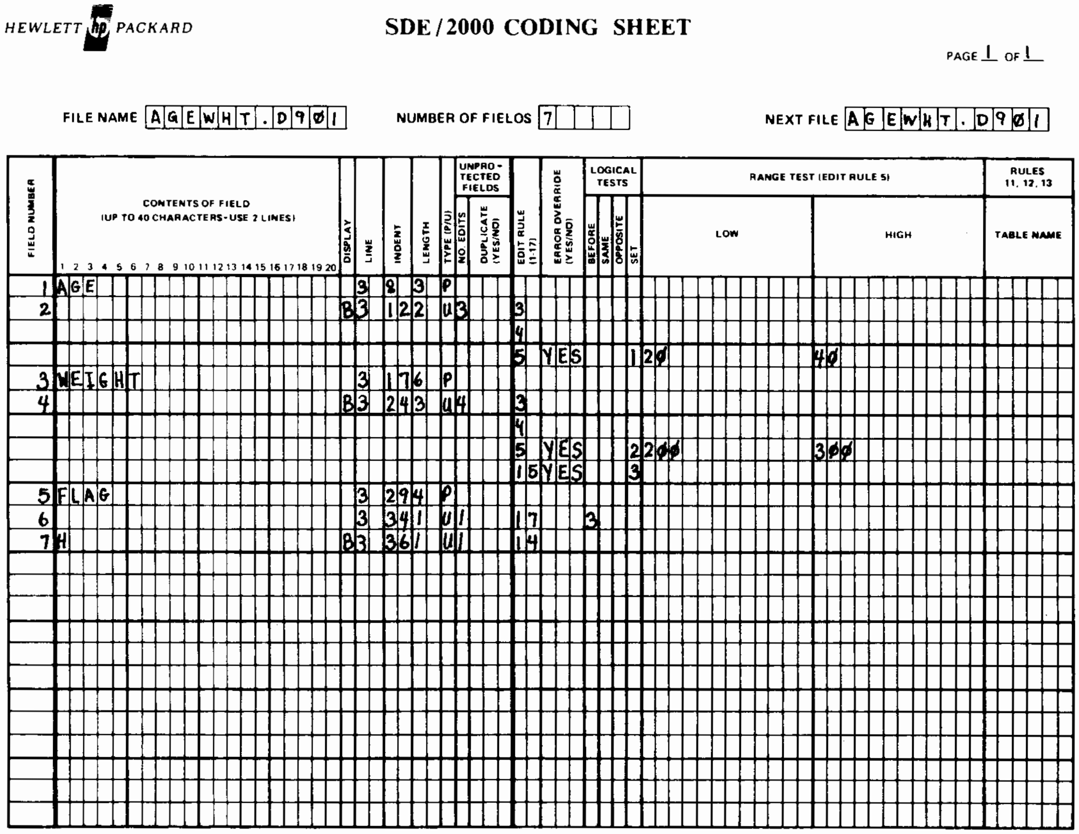
3-22. RULE 17: MOVE FLAG

Move the character # to the unprotected field. This rule can be used with other edit rules to flag data that falls into specific categories. An example is given in the next subsection.

Error Indicator: Always 1

3-23. USE OF LOGICAL EDITS

Rules 15 and 16 can be combined with the range test to check for several conditions. Suppose a form has two input fields, weight and age, and you want to flag everyone between the ages of 20 and 40 that weighs 200 to 300 pounds. You specify the two range tests on the data, and set error flags 1 and 2 with the results. Then you perform the AND operation and set another flag with the result. This third flag indicates whether the # character should be moved to the data field. This example is coded in figure 3-1. (The final HOLD/REJECT field is included so the screen will stay long enough for you to see the #.)



HEWLETT PACKARD

SDE/2000 CODING SHEET

PAGE 1 OF 1

FILE NAME AGEWHT.D9Ø1 NUMBER OF FIELDS 7 NEXT FILE AGEWHT.D9Ø1

FIELD NUMBER	CONTENTS OF FIELD (UP TO 40 CHARACTERS-USE 2 LINES) 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20	DISPLAY	LINE	INDENT	LENGTH	TYPE (P/U)	UNPRO-TECTED FIELDS		EDIT RULE (1-17)	ERROR OVERRIDE (YES/NO)	LOGICAL TESTS				RANGE TEST (EDIT RULE 5)		RULES 11, 12, 13
							NO. EDITS	DUPLICATE (YES/NO)			BEFORE	SAME	OPPOSITE	SET	LOW	HIGH	TABLE NAME
1	AGE		3	8	3	P											
2		B	3	12	2	U	3		3								
									4								
									5	YES				1	2Ø	4Ø	
3	WEIGHT		3	17	6	P											
4		B	3	24	3	U	4		3								
									4								
									5	YES				2	2ØØ	3ØØ	
									15	YES				3			
5	FLAG		3	29	4	P											
6			3	34	1	U	1		17		3						
7	H	B	3	36	1	U	1		14								

Figure 3-1. Coding Sheet, AGEWHT

3-24. SUMMARY OF EDIT RULES

RULE	NAME	ERROR CONDITION SET TO	
		0 = ERROR FOUND	1 = NO ERROR
1	Alphanumeric Field Check	Non-Alphanumeric character found	All characters alphanumeric
2	Alphabetic Field Check	Non-Alphabetic character found	All characters alphabetic
3	Numeric Field Check	Non-Numeric character found	All characters numeric
4	Right Justify Zero Fill	Never	Always
5	Range Check	Value out of range	Value within range
6	Modulo 11 Create	Check digit not created	Check digit created
7	Modulo 11 Verify	Check digit did not verify	Check digit passed
8	Save Contents of Input Field	Never	Field moved to save area
9	Display Contents of Save Area	Source bigger than destination	Save area moved to field
10	Save Search Key	Never	Field placed in search key
11	Table Compare	Requested file not on system or No match within file	Match is found
12	Table Replace	Requested file not on system or No match within file	Table data placed in field
13	Screen Name Replace	Requested file not on system or No match within file	Next Screen name replaced
14	Screen Hold/Reject	Never	Always
15	AND Function	Flags 1 and 2 not both 1	Flags 1 and 2 both 1
16	OR Function	Flags 1 and 2 both 0	Either Flag 1 or 2 is 1
17	Move Flag	Never	Always

HOW TO ENTER DATA

SECTION IV

This section describes how to use SDE/2000 to enter data into disc files.

4-1. GETTING ON THE TERMINAL

- If you are using the 2640A, the log-on process is as follows:

  Press RETURN and LINE FEED keys
  System will respond with the message, PLEASE LOG IN
  Type HELLO-account,password,2

  where "account" is a Data Entry user account (as described in figure 1-4) and "password" is the password for that account. The "2" indicates that you are using a 2640A terminal in page mode.

- If you are using the 2644A, the log-on process is as follows:

  Depress the Auto LF button
  Press RETURN key
  System will respond with the message, PLEASE LOG IN
  Release the Auto LF button
  Type HELLO-account,password,2

  where "account" is a Data Entry user account (as described in figure 1-4) and "password" is the password for that account. The "2" indicates that you are using a 2644A terminal in page mode.

4-2. PRELIMINARIES

Before using the Data Entry program (SDE), the Screen Description files must be created and protected by the forms designer as described in Sections II and III.

The Data Entry Supervisor should make certain that a file has been created to hold the data to be entered. The amount of space required in a data file can be estimated as follows:

> One record holds approximately 500 characters.
>
> Count the number of input characters in the form and multiply by the number of times you expect to use the form. This gives you the total number of characters you expect to be entering into the system.
>
> Divide by 500 to determine the number of records necessary to hold the data for that form.

If you are using different forms, calculate the amount of space needed for each form and use the total for your data file size.

Example:

For a data file named OUTPUT requiring 100 records, use the command:

```
CREATE-OUTPUT,100
```

It is important to remember that from session to session, SDE appends data to existing data files. To reinitialize a data file so that SDE starts entering data at the beginning of the file, PURGE the file and then CREATE it again. Since all existing data in a file is lost when the file is purged, make sure that the data is no longer needed, or has been copied to magnetic tape.

It is possible to allow automatic execution of the data entry program (SDE) whenever someone logs on to the Data Entry user account. In this way, terminal operators will not need to type the EXEC-*SDE command. In order to have the 2000 Access system start execution of the Data Entry program, you must change the HELLO program that is stored in account A000. This program, which is executed whenever anyone logs on the system, normally prints a system message and terminates by printing READY. In the following example, the statements 100 to 150 have been added to the HELLO program to start execution of SDE whenever account D902 logs on:

```
 10 PRINT "SYSTEM MESSAGE"
      .
      .
      .

100 DIM A$(80)
110 SYSTEM A$, "TIME"
120 IF A$(1,4)#"D902" THEN 140
130 CHAIN "*SDE"
140 PRINT "READY"
150 END
```

4-3. EXECUTING THE PROGRAM

After the preliminary work is done, you are ready to execute the Data Entry program. Type the command

```
EXEC-*SDE
```

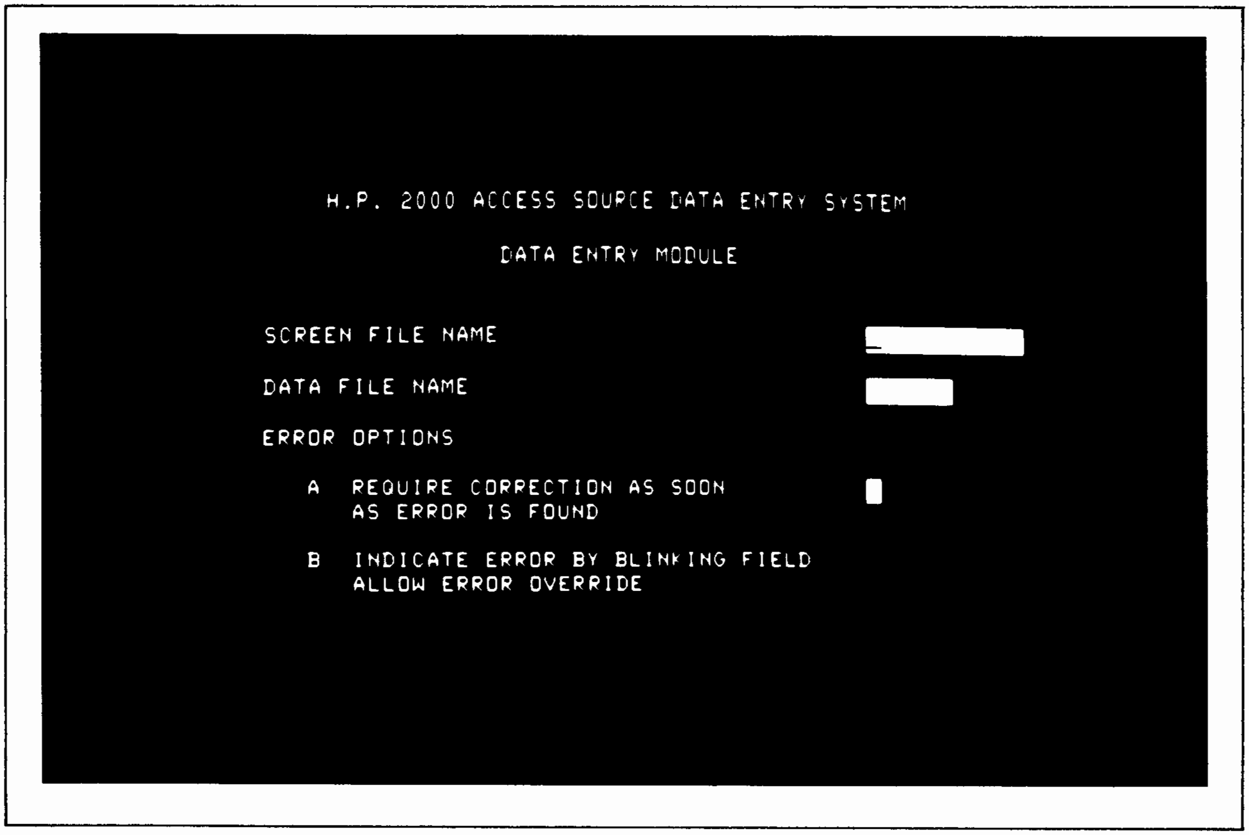
The first message you will see on the terminal is DEPRESS BLOCK MODE BUTTON. During execution of SDE the terminal communicates with the computer in block mode, so the button must remain depressed. You will then be ready to use the program.

An additional check is made to insure that the terminal is strapped for page block mode. If it is not, the message TERMINAL MUST BE STRAPPED FOR PAGE BLOCK MODE appears, and the program terminates. Refer to Appendix A to convert the terminal to page mode.

On all the screens, if an answer to a question fills up the input field, the terminal cursor will automatically move to the next field. Otherwise, you may use the TAB key to get from one field to the next. After you have answered all questions, check to make sure they are correct. They can be changed by using the DELETE CHAR key or by positioning the cursor to the field and typing over the old information. For multi-page forms, the NEXT PAGE key can be used to display the next 24 lines of memory and the PREV PAGE key can be used to display the previous 24 lines of memory. When you are satisfied with the screen, depress the ENTER key to send the information to the computer. If SDE catches any errors, a blinking message appears. You must retype the incorrect field and depress the ENTER key again before processing can continue.

4-4. THE SELECTION SCREEN

The Selection screen (figure 4-1) gives you the ability to select your form, data file, and error options.



```
        H.P. 2000 ACCESS SOURCE DATA ENTRY SYSTEM

                    DATA ENTRY MODULE

    SCREEN FILE NAME

    DATA FILE NAME

    ERROR OPTIONS

       A   REQUIRE CORRECTION AS SOON
           AS ERROR IS FOUND

       B   INDICATE ERROR BY BLINKING FIELD
           ALLOW ERROR OVERRIDE
```

Figure 4-1. Selection Screen

The first piece of information you must type is the fully qualified name of the file containing the desired form's description, or the word EXIT if you want to terminate the program. Since the screen description file for the form resides in the forms design account (as described in figure 1-4), you must qualify the file name by indicating the account number. For example, if you want to use the Screen Description file SCR1 that is saved under account D901, you make the entry

SCR1.D901

The second entry is the name of the file that will contain the output data. An entry of EXIT will terminate the program. When a data file is selected, SDE/2000 sets the file pointer to the end of the data in the file; previous information is retained.

For both of the first two entries, the terminal cursor will move to the next field when the name fills up the entire input area. Otherwise you may use the TAB key to get from one field to the next.

The third and final entry on this screen is the error option. By selecting option A, SDE/2000 will require that any error it finds in an input field be corrected before the data is recorded. (This is the usual option.) When bad data is entered, an error message appears at the bottom of the screen and the field must be retyped. When option B is selected, if a field is in error SDE blinks the contents and moves on to the next field. After all the fields are read, if there were errors the message DATA ERROR OVERRIDE (YES-NO) appears at the top of the form. If you answer YES, SDE writes the data to the Output Data file and flags it as being in error. Otherwise the cursor is positioned to the beginning of the form and you must correct the errors and retransmit the data.

After you have answered the three questions on the Selection screen, check to make sure they are correct. They can be changed by positioning the cursor to the field and typing over the old information. When you are satisfied, depress the ENTER key to send the information to the computer. If there are any errors, you will be informed by a blinking message. The incorrect field must be retyped and the entire screen reentered before processing can continue.

4-5. THE FORM SCREEN

After you select the form, SDE displays it on the 2640A. Four functions are presented at the top of the screen. After entering the data on the screen, and before depressing the ENTER key, position the cursor to the first field by using the TAB key or Cursor Home key (↖). Select one of the four functions:

CONTINUE
The data is written into the Output Data file and the next screen appears on the terminal.

REPEAT
The data is written into the Output Data file. Unprotected fields of the form that don't allow duplication are cleared, permitting additional data to be entered using the same form.

RETURN
The data is written into the Output Data file and the Selection screen appears on the terminal.

EXIT
No data is written to the Ouput Data file and the Selection screen appears on the terminal.

The functions are abbreviated as CON, REP, RET, and EX. They are typed in the first field (line 0). If the field is left blank the default is CONTINUE.

Individual screens may be up to three pages in length. As soon as one page is full the next one will appear. You may tab between pages or use the NEXT PAGE key on the terminal.

As you did with the Selection screen, check the data before you depress the ENTER key. If any fields are in error and you specified option A on the Selection screen, you must make corrections.

4-6. COMPLETION OF SDE

After you type EXIT for one of the file names on the Selection screen, SDE will terminate and print DONE on the terminal. You must take the 2640A out of block mode before any further commands can be issued to the computer. To do this, release the BLOCK MODE key. Upon completion of SDE your Output Data file contains the information entered from the forms. The file holds the information for each field in a string variable.

The exact format of the data in the file is given in Section V.

4-7. QUICK REFERENCE TO DATA ENTRY

Log on to 2000 Access with 2640A terminal strapped for page mode. CREATE a file to hold the data, if needed.

EXECUTE SDE

Depress BLOCK MODE key
Enter Selection screen information
Enter data for each form
EXIT from form
EXIT from Selection screen
Release BLOCK MODE key

Log off

Example:

```
cr linefeed
PLEASE LOG IN
HELLO-D902,PASWRD,2
CREATE-OUTPUT,100
EXEC-*SDE
        (Enter data to OUTPUT)
BYE
```

4-8. VALID INPUT VALUES

SELECTION SCREEN

Screen file name	Fully qualified name of any protected file on the system, or EXIT
Data file name	Any file in your account or EXIT
Error Option	A (require correction) or B (allow errors)

USER PROGRAMS

SECTION V

This section describes how SDE/2000 data files can be used. In order to use the data output files you should be familiar with 2000 Access/BASIC, particularly the statements and functions for file manipulation.

5-1. USING SDE/2000 OUTPUT DATA FILES

Broadly speaking, there are two ways you can use SDE/2000 Output Data files: send them to a larger remote computer or work with them on the HP 2000 Access system. In the first application, the data is presented to the remote machine as a series of card images. You can use existing programs to accomplish this or you can write your own. The second application is to write your own BASIC programs to manipulate the data, using the files stored on the 2000 Access system.

5-2. FORMAT OF OUTPUT DATA FILES

The data for a particular screen is preceded by a header record with the following format:

Number	Value set to zero
Character String	
Characters 1 - 11	Name of the screen file, fully qualified. (Fully qualified means that the file name must include the account number where the file resides)
Character 12	Error Flag: 1 = SDE detected no errors in data 0 = SDE has detected errors in data

This record is followed by the data records received from unprotected fields. There is one record for the data from each unprotected field on the screen. The size and order of the records match the size and order of the unprotected fields on the screen.

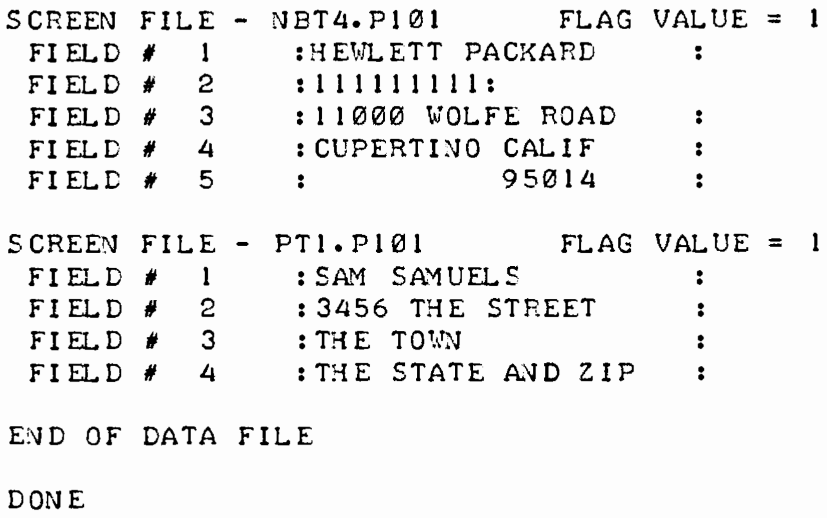
Each data record is written as a number followed by a string. The numeric variable is the number of the input field on the screen. It therefore varies from one to the number of input fields on the screen. The string is the edited data from the unprotected field of the form.

The following sample program, named DATEST, lists the contents of an output data file, taking advantage of the fact that the number is a field count to detect when a new screen header has been read.

```
DATEST

1Ø   DIM F$[6],S$[4Ø]
2Ø   FILES *
3Ø   PRINT "DATA FILE NAME:";
4Ø   LINPUT F$
5Ø   ASSIGN F$,1,T1
6Ø   IF T1<3 THEN 9Ø
7Ø   PRINT "FILE NOT ACCESSIBLE."
8Ø   GOTO 3Ø
9Ø   IF  END #1 THEN 16Ø
1ØØ   READ #1;S,S$
11Ø   IF S#Ø THEN 14Ø
12Ø   PRINT LIN(1)"SCREEN FILE - "S$[1,11]"     FLAG VALUE = "S$[12]
13Ø   GOTO 1ØØ
14Ø   PRINT " FIELD # "S":"S$":"
15Ø   GOTO 1ØØ
16Ø   PRINT LIN(1)"END OF DATA FILE"
17Ø   END
```

Executing the program DATEST produces the following printout.

```
DATEST

DATA FILE NAME:BOUT

SCREEN FILE - NBT3.P1Ø1       FLAG VALUE = 1
 FIELD #   1     :MR CUSTOMER           :
 FIELD #   2     :987654321:
 FIELD #   3     :1234 STREET FIVE      :
 FIELD #   4     :CUPERTINO             :
 FIELD #   5     :CALIFORNIA  95Ø14     :
 FIELD #   6     :2112:
 FIELD #   7     :BIG MAMA OF 21MX FAMILY:
 FIELD #   8     :2116:
 FIELD #   9     :PREDECESSOR TO 21ØØ:
 FIELD #   1Ø    :MED:

SCREEN FILE - NBT4.P1Ø1       FLAG VALUE = 1
 FIELD #   1     :MS CUSTOMER           :
 FIELD #   2     :555555555:
 FIELD #   3     :212223 SOME ROAD      :
 FIELD #   4     :POTTSTOWN   PA        :
 FIELD #   5     :          19464       :
```



```
SCREEN FILE - NBT4.P1Ø1        FLAG VALUE = 1
 FIELD #  1     :HEWLETT PACKARD      :
 FIELD #  2     :111111111:
 FIELD #  3     :11ØØØ WOLFE ROAD     :
 FIELD #  4     :CUPERTINO CALIF      :
 FIELD #  5     :          95Ø14      :

SCREEN FILE - PT1.P1Ø1         FLAG VALUE = 1
 FIELD #  1     :SAM SAMUELS          :
 FIELD #  2     :3456 THE STREET      :
 FIELD #  3     :THE TOWN             :
 FIELD #  4     :THE STATE AND ZIP    :

END OF DATA FILE

DONE
```

As shown in figure 1-4, data files reside in the user account designated for data entry, (such as D902). This account does not normally have PFA (program/file access) capability and data files are normally locked. Therefore, any programs accessing the data files should be locked programs placed in the group master account (such as D900) and executed in the data entry account (D902). Since the program is being executed in the account where the file resides, it is not necessary for the input file name to be fully qualified; that is, include the account number.

5-3. FORMAT OF TABLE FILES

SDE/2000 contains a utility program for table files (SDETBL), but if you want to modify it or write your own utility program, you will need to know the format of the tables.

Each entry in the table has the following format:

number (can be ignored)
40 character ASCII string (key)
40 character ASCII string (data)

If the table file is used only for look-up, the data field string will still be in the file but will be a null string.

The following sample program named TLIST lists the contents of a table file.

```
TLIST

10  DIM F$[11],K$[40],D$[40]
20  FILES *
30  PRINT LIN(1);"TABLE FILE NAME:";
40  LINPUT F$
50  IF (POS(F$,".")) THEN 80
60  PRINT "FILE NAME MUST BE FULLY QUALIFIED"
70  GOTO 30
80  ASSIGN F$,1,A
90  IF A<3 THEN 120
100  PRINT "FILE NOT ACCESSIBLE"
110  GOTO 30
120  I=1
130  PRINT
140  IF  END #1 THEN 190
150  READ #1;N,K$,D$
160  PRINT I;TAB(7);K$"  :  "D$
170  I=I+1
180  GOTO 150
190  END
```

Executing the program TLIST produces the following printout.

```
RUN
TLIST


TABLE FILE NAME:BTAB1.P101

 1     2100  :  THE FAMOUS ONE
 2     2105  :  SMALL 21MX
 3     2108  :  MEDIUM 21MX
 4     2112  :  LARGE 21MX
 5     2114  :  IT HAD NEAT SWITCHES
 6     2115  :  WHO EVER HEARD OF IT
 7     2116  :  IT HAD LOTS OF LITES

DONE
```

As shown in figure 1-4, table files reside in the forms design account and must be protected. Since the forms design account has PFA (program/file access), and the table files are protected, they are available on a read-only basis to other user accounts. This means the sample program TLIST (which only reads and lists the table files) can be executed in any user account and therefore the input file name must be fully qualified by including the account number.

If you write a program to *modify* table files, it should be run while logged on to the forms design account, since the files are protected and modification requires write access. In this case you do not need to fully qualify the file name because you are logged on to the account where the file resides.

5-4. FORMAT OF SDE/2000 SCREEN DESCRIPTION FILES

The following information will be of use if you want to access Screen Description files. The structure of each record on the Screen Description file is

Description	Length	Start Character
Screen Control	49	1
Edit Number 1	38	50
Edit Number 2	38	88
Edit Number 3	38	126
Edit Number 4	38	164
Edit Number 5	38	202
	239 Characters	

The Screen Control has the following fields:

Character Position	Length	Description
1	2	Line Number: a value from 1 to 70
3	2	Indent Position: a value from 1 to 79
5	1	Type Code: P or U
6	2	Length of field: a value from 1 to 40
8	1	Display Enhance code: see figure 2-10
9	1	Duplication Allowed: N (no) or Y (yes)
10	40	Field contents: ASCII string

The five edits all have the same format. The first one is:

50	1	Active Indicator: 0 = no edit 1 = edit assigned	
51	2	Edit Rule number: a Value from 1 to 17	
53	1	Test Flag indicator	A value from 1 to 9 indicating which logical flag to use. A value of 0 indicates no action.
54	1	Same As indicator	
55	1	Opposite From indicator	
56	1	Set Flag indicator	
57	1	Error Override: 0 = NO, 1 = YES	
58	6	Table Name: file used for Edit 11, 12, or 13	
64	12	Range Low	Used for Edit 5
76	12	Range High	

5-5. USER WRITTEN INTERFACE

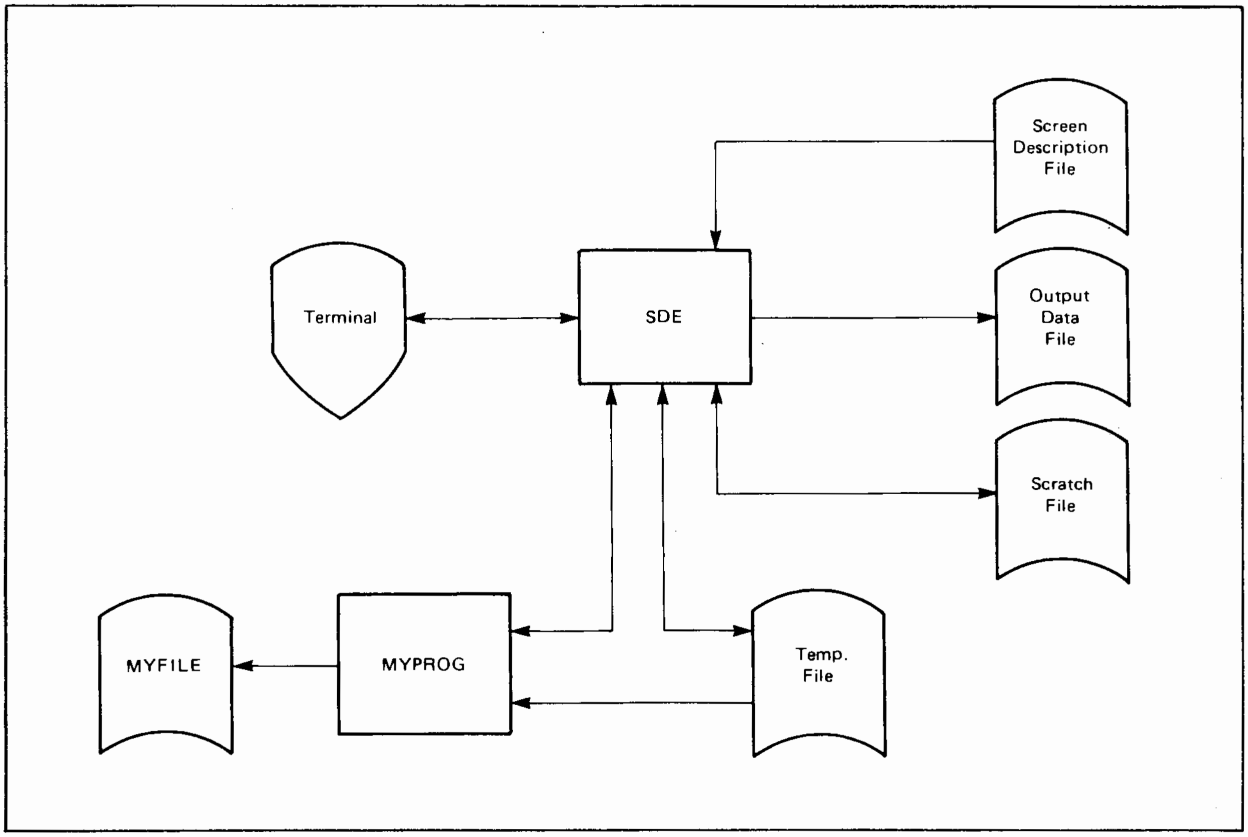


Figure 5-1. Interaction Between Programs and Files

The Data Entry program can be used as a "front end" to your own program. That is, you can have SDE collect information from the terminal, then transfer to a program you have written to manipulate the data.

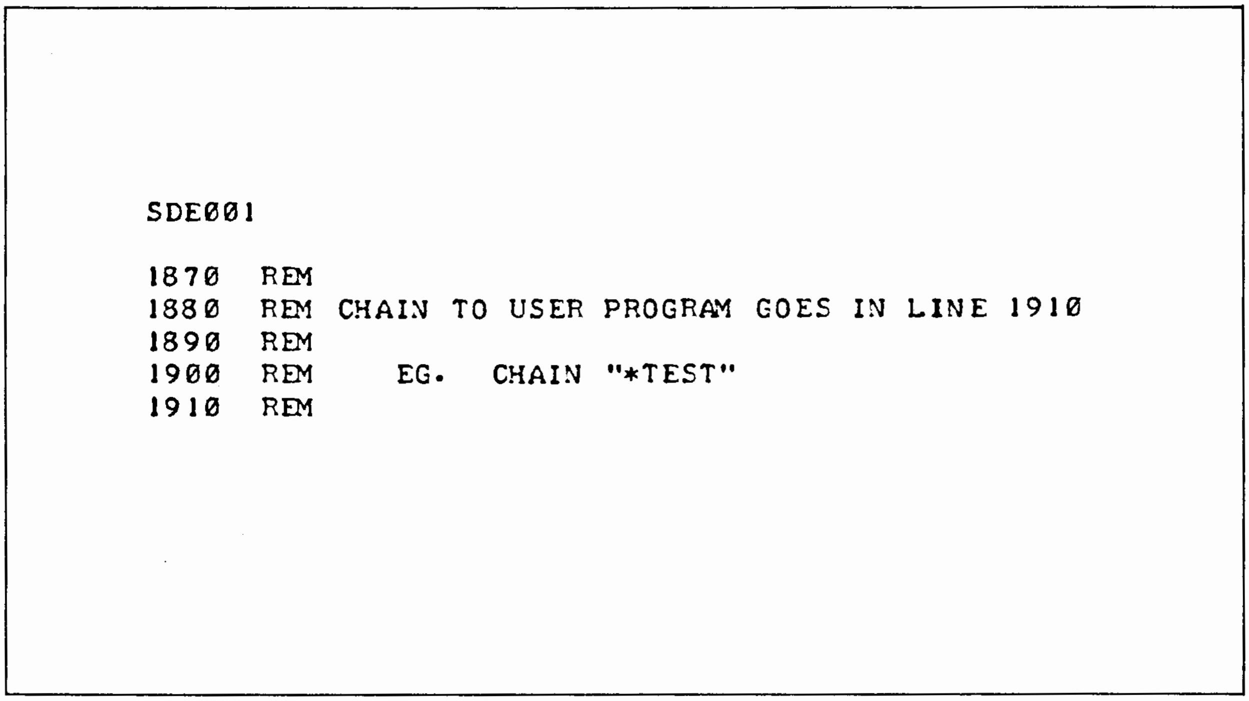
Before writing your program, you need to understand the Data Entry program. This program consists of two modules which chain to each other: SDE and SDE001. Together they perform the following:

FORMS DISPLAY - SDE draws the Selection screen and gets the names of the Screen Description and Output Data files.

DATA ENTRY AND EDITING - SDE001 draws the form from the Screen Description file, reads the contents of unprotected fields on the terminal into a scratch file, edits them based on information contained in the Description file, and writes them in a temporary file.

DATA STORAGE - SDE001 transfers the contents of the temporary file to the Output Data file.

You must chain to your program from SDE001 after Data Entry and Editing, manipulate the data, then chain back. For SDE001 to continue processing, you must preserve global and local variables. SDE001 will not perform Data Storage when you chain back. Figure 5-2 shows the section of SDE001 where you will chain to your own program.



```
SDE001

1870  REM
1880  REM  CHAIN TO USER PROGRAM GOES IN LINE 1910
1890  REM
1900  REM     EG.   CHAIN "*TEST"
1910  REM
```

Figure 5-2. Code from SDE001

To aid you in writing interface programs, SDECOM contains the common block for SDE. Log on to the group account and type the command

GET-SDECOM

to bring the COM statements into your Work Space. Then start writing your program, beginning with a statement number greater than 4. The common block contains several variables that may be of use to you:

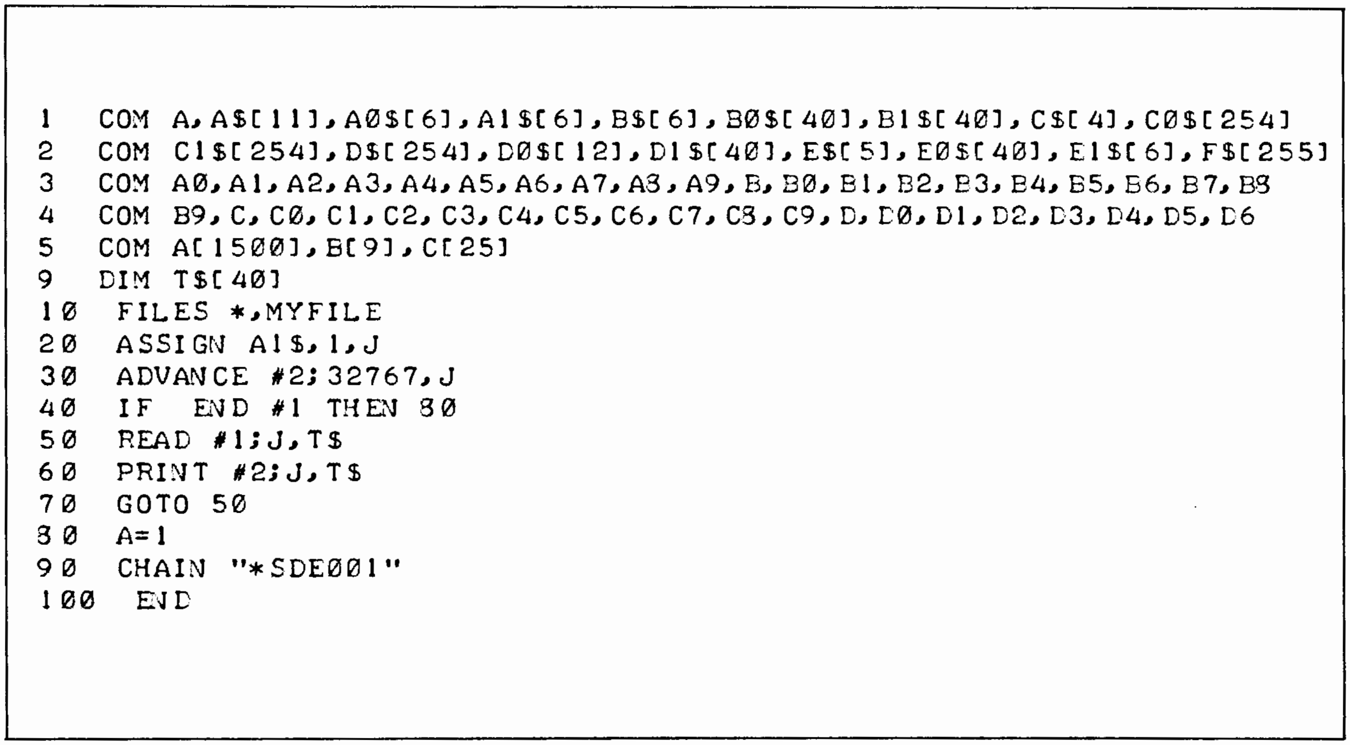
A$	Screen File name
A0$	Output Data File name
B$	Scratch File name
A1$	Temporary File name
A	SDE001 Entry Switch

Somewhere in your program you must set A to 1 to indicate that program execution is being returned to SDE001 by your program. The last statement executed by your program must be

CHAIN "*SDE001"

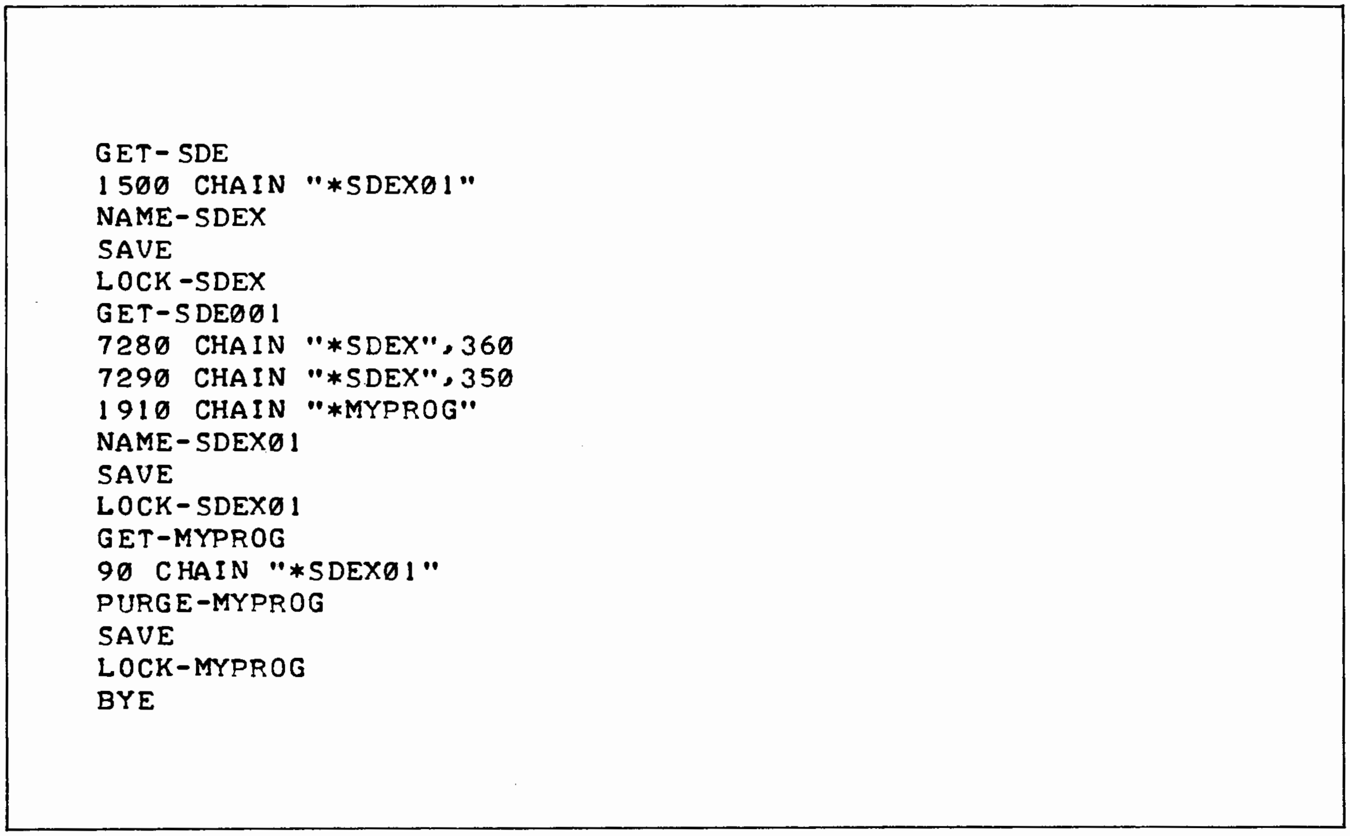
As an example, suppose you want to chain to MYPROG and copy the contents of the temporary file into a file named MYFILE. Figure 5-3 lists such a program.

You may not want to destroy the original copy of SDE001 with a new version that chains to your program. To avoid this situation you can save the modified version under another name, such as SDEX01. If you do this, you must also rename SDE and change the statements which chain to SDE and SDE001. Figure 5-4 lists these additional changes.



```
1   COM A,A$[11],AØ$[6],A1$[6],B$[6],BØ$[40],B1$[40],C$[4],CØ$[254]
2   COM C1$[254],D$[254],DØ$[12],D1$[40],E$[5],EØ$[40],E1$[6],F$[255]
3   COM AØ,A1,A2,A3,A4,A5,A6,A7,A8,A9,B,BØ,B1,B2,B3,B4,B5,B6,B7,B8
4   COM B9,C,CØ,C1,C2,C3,C4,C5,C6,C7,C8,C9,D,DØ,D1,D2,D3,D4,D5,D6
5   COM A[15ØØ],B[9],C[25]
9   DIM T$[4Ø]
1Ø  FILES *,MYFILE
2Ø  ASSIGN A1$,1,J
3Ø  ADVANCE #2;32767,J
4Ø  IF  END #1 THEN 8Ø
5Ø  READ #1;J,T$
6Ø  PRINT #2;J,T$
7Ø  GOTO 5Ø
8Ø  A=1
9Ø  CHAIN "*SDEØØ1"
1ØØ  END
```

Figure 5-3. Sample Interface Program (MYPROG)



```
GET-SDE
15ØØ CHAIN "*SDEXØ1"
NAME-SDEX
SAVE
LOCK-SDEX
GET-SDEØØ1
728Ø CHAIN "*SDEX",36Ø
729Ø CHAIN "*SDEX",35Ø
191Ø CHAIN "*MYPROG"
NAME-SDEXØ1
SAVE
LOCK-SDEXØ1
GET-MYPROG
9Ø CHAIN "*SDEXØ1"
PURGE-MYPROG
SAVE
LOCK-MYPROG
BYE
```

Figure 5-4. Changing Name of SDE

APPENDIX A

LOADING SDE/2000

This section describes the method of loading SDE/2000 on the 2000 Access system. SDE/2000 uses the file and program security features of 2000 Access to prevent accidental destruction of programs and screen description files. Refer to figure 1-4 for an example of account structure and capabilities.

A-1. LIST OF PROGRAMS

SDE/2000 consists of these programs:

SDEDM draws the screen pictured in figure 2-8 and chains to SDEDM1.

SDEDM1 draws the screens pictured in figures 2-9 and 2-11, and writes the Screen Description file.

SDEPRE prepares Screen Description files. It is chained to by SDEDM1 and is not apparent to the user.

SDEUTL performs utilities on Screen Description files.

SDETBL performs utilities on Table files.

SDE draws the selection screen, gets file names, and chains to SDE001.

SDE001 draws the form, collects data from the terminal, writes the Output Data file, and chains to SDE when the EXIT command is entered.

SDECOM contains the SDE common block.

A-2. HP 2640A TERMINAL

HP 2640A terminals that are used to run SDEDM and SDE must be strapped for page mode. Normally this option is disabled and the terminal is set for line-field operation. To change from line-field to page mode:

a. Unplug the 2640A and open it to its half-open position.

b. Locate and remove the Keyboard Interface PCA, part no. 02640-60019, from the Backplane Assembly connector closest to the power supply.

c. Remove jumper D from the board.

d. Reinstall the Keyboard Interface PCA.

e. Close the terminal cover and reconnect the power.

f. Perform the terminal self-test.

For further information, consult pages 1-11 and 1-12 of *Installation and Service Manual, Model 2640A Interactive Display Terminal* (02640-90012).

A-3. HP 2644A TERMINAL

HP 2644A terminals that are used to run SDEDM and SDE must be strapped for page mode. Normally this option is disabled and the terminal is set for line-field operation. To change from line-field to page mode:

a. Unplug the 2644A and open it to its half-open position.

b. Locate and remove the Keyboard Interface PCA, part no. 02640-60019 from the Backplane Assembly connector.

c. Remove jumper D from the board.

d. Reinstall the Keyboard Interface PCA.

e. Close the terminal cover and reconnect the power.

f. Perform the terminal self-test.

For further information, consult pages 1-10 and 1-11 of *Installation and Service Manual, Model 2644A Mini DataStation* (02644-90002).

A-4. SETTING BUFFER SIZE ON SDE PORTS

The default buffer size for ports attached to the 2000 Access system is 64 words. Each of the ports that use SDE/2000 must have a buffer size of 128 words. The following console listing shows how to set the buffer size on all ports to 128 words during I/O Processor Configuration.

```
HP 2ØØØ ACCESS I/O PROCESSOR CONFIGURATOR
RELOAD? NO
DATE? 1Ø.17.75:GDK
MEMORY SIZE?  32
NUMBER OF PORTS?  32
BUFFER LENGTH OPTION? YES
ENTER PORT NUMBER, BUFFER LENGTH LIST; TERMINATE WITH 'END'
ALL,128
END
TIME BASE GENERATOR SELECT CODE?
```

The next listing shows how to set the buffer size for four ports (0, 1, 20, and 30).

```
HP2ØØØ ACCESS I/O PROCESSOR CONFIGURATOR
RELOAD? NO
DATE? D
MEMORY SIZE?  32
NUMBER OF PORTS?  32
BUFFER LENGTH OPTION?  YES
ENTER PORT NUMBER, BUFFER LENGTH LIST; TERMINATE WITH 'END'
Ø,128
1,128
2Ø,128
3Ø,128
END
TIME BASE GENERATOR SELECT CODE?
```

ERRORS

APPENDIX B

SDE/2000 provides error messages on the bottom line of the 2640 terminal. Non-fatal errors are listed in table B-1. You must correct the indicated error before processing can continue.

Table B-1. Non-Fatal Errors

	FORMS DESIGN SCREEN					DATA ENTRY	
ERROR MESSAGE	START-UP	FIELD DESCRIPTION	EDIT RULE	TABLE FILE UTILITY	SCREEN FILE UTILITY	SELECTION	FORM
TRANSMISSION ERROR — TRY AGAIN	X	X	X		X	X	X
FILE NAME MUST BE FULLY QUALIFIED	X				X	X	
READ ONLY FILE	X						
FILE NOT ACCESSIBLE	X					X	
INPUT NOT NUMERIC	X	X	X				X
INPUT VALUE IS NEGATIVE OR ZERO		X					
INPUT LINE NUMBER RESERVED BY SYSTEM		X					
LINE NUMBER GREATER THAN 71		X					
LINE NUMBER NOT ASCENDING		X					
INDENT POSITION GREATER THAN 79		X					
INDENT VALUE NOT ASCENDING		X					
LENGTH GREATER THAN 79		X					
MORE THAN FIVE EDITS REQUESTED		X					
TYPE CODE NOT P OR U		X					
DISPLAY CODE INCORRECT		X					
ANSWER NOT VALID		X					
INVALID EDIT RULE NUMBER			X				
INVALID EDIT FLAG NUMBER			X				
FILE NAMED NOT ON SYSTEM			X				X
FILE IN USE BY SOMEONE ELSE				X			
NO SUCH FILE				X			
TYPE 'ADD', 'REMOVE', OR 'LIST'				X			
TOO LONG				X			
DUPLICATE KEY				X			
NO SUCH KEY				X			
TABLE EMPTY				X			
TABLE FILES MUST BE AT LEAST 2 RECORDS LONG				X			
FILE IS NOT A FILLED SCREEN FILE					X		
UNABLE TO ACCESS *(filename)*					X		
REACHED END BEFORE FINDING THIS FIELD					X		
PLEASE RECORD: 'TERM' FOR TERMINAL LP ASCII FILE NAME FOR LINE PRINTER					X		
ENTER 'A' OR 'B' ONLY						X	
INPUT NOT ALPHANUMERIC							X
INPUT NOT ALPHABETIC							X
INPUT NOT WITHIN REQUIRED RANGE							X
CHECK DIGIT ERROR							X
INPUT NOT FOUND IN VERIFICATION FILE							X
INVALID INPUT							X

If the message FORM DESIGN OR SYSTEM ERROR appears on the terminal, a fatal error has occurred because the forms designer made a mistake, someone tampered with the Data Entry program, or something is wrong with the disc files. Your form will not work without being redesigned. Further fatal errors are listed in table B-2.

If the message TRANSMISSION ERROR-TRY AGAIN appears, depress the ENTER key again.

If a program termination occurs, and you cannot get the terminal to respond, take the following action:

Contact the system operator to reenable the BREAK key for your port.
Release the BLOCK MODE key.
Depress the RESET TERMINAL key.
Depress the BREAK key.
Type ECHO-ON.

Table B-2. Fatal Errors

```
CAN'T PREPARE FILE - CHARACTER SPACE FULL
CAN'T PREPARE FILE - NUMERIC SPACE FULL
FILE FULL. ENTRY NOT ADDED
FORM DESIGN OR SYSTEM ERROR - BAD EDIT RULE
FORM DESIGN OR SYSTEM ERROR - DATA FILE FULL
FORM DESIGN OR SYSTEM ERROR - DATA FILE PROB
FORM DESIGN OR SYSTEM ERROR - EMPTY FORM FILE
FORM DESIGN OR SYSTEM ERROR - FORM FILE PROB
FORM DESIGN OR SYSTEM ERROR - IMPOSSIBLE EDIT ERROR
FORM DESIGN OR SYSTEM ERROR - NO SAVE FIELD
FORM DESIGN OR SYSTEM ERROR - SCRATCH FILE PROB
FORM DESIGN OR SYSTEM ERROR - TABLE FILE PROB
FORM DESIGN OR SYSTEM ERROR - TEMP FILE PROB
FORM DESIGN OR SYSTEM ERROR - UNPREPARED FORM FILE
SCREEN FILE NOT ACCESSIBLE FOR PREPARATION
SCREEN FILE OVERFLOW - INCREASE SIZE OF THE SCREEN FILE
STOPPING DUE TO LACK OF RESPONSE
TERMINAL MUST BE STRAPPED FOR PAGE BLOCK MODE
UNABLE TO ASSIGN SDE SCRATCH FILE
UNABLE TO ASSIGN SDE TEMPORARY FILE
UNABLE TO CREATE SDE SCRATCH FILE
UNABLE TO CREATE SDE TEMPORARY FILE
```

EXAMPLES

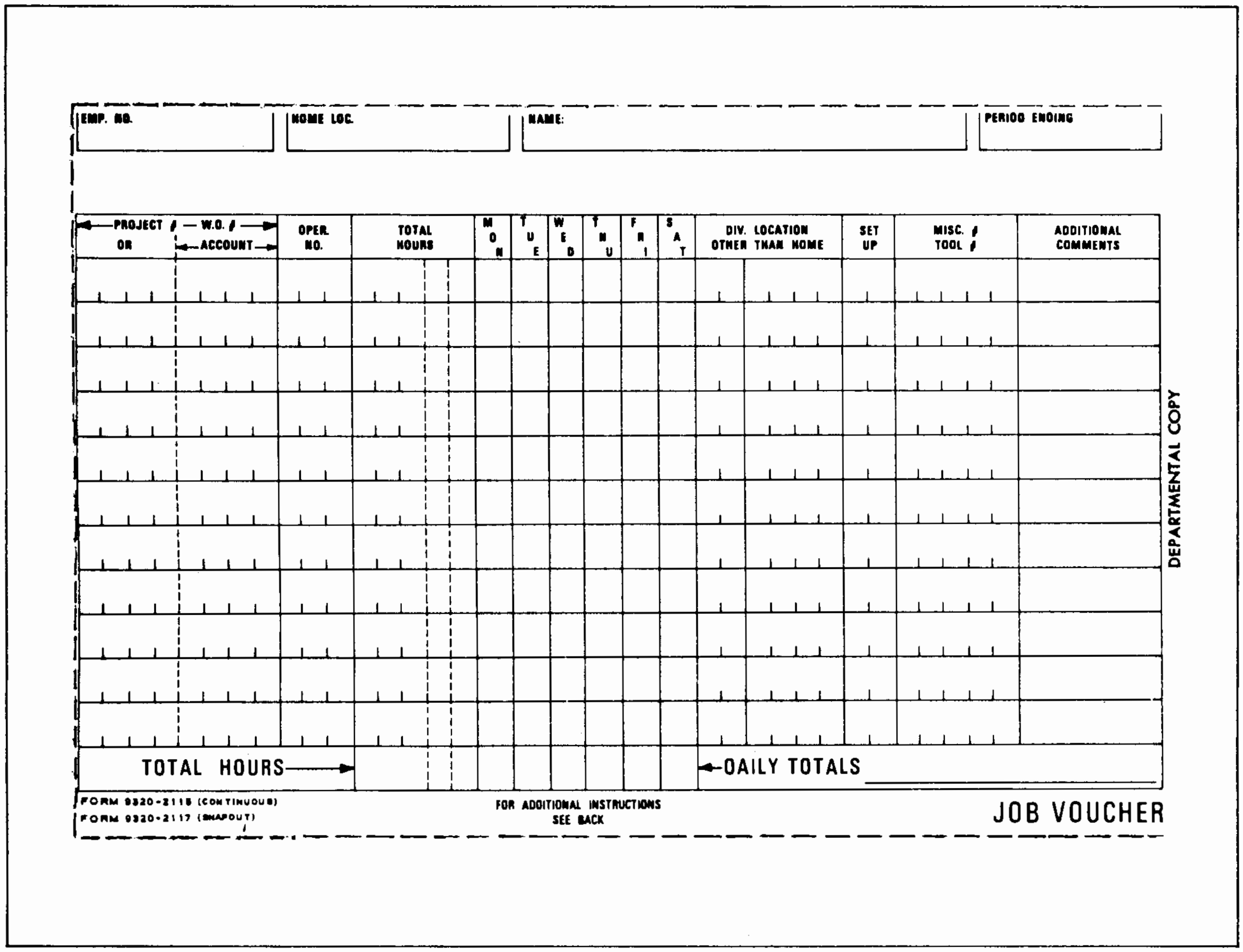
APPENDIX C

This section provides further examples of SDE/2000 screen forms.

C-1. JOB VOUCHER

The manufacturing area of a company uses many forms that are suitable for SDE/2000 design: inspection results, tooling information, machine utilization, master schedule, work order, inventory ticket, etc. One such form, the job voucher, is shown in figure C-1. Since this form can have an arbitrary number of projects entered we will split it into multiple logical screens. The first screen contains the employee information, the second contains information for a project, and the third contains the daily and weekly totals. The only edit checks are for numeric input.

The steps used in developing the form are illustrated in figures C-2 through C-11.



EMP. NO.	HOME LOC.	NAME:	PERIOD ENDING

←PROJECT # — OR	W.O. # → ←ACCOUNT→	OPER. NO.	TOTAL HOURS	MON	TUE	WED	THU	FRI	SAT	DIV. LOCATION OTHER THAN HOME	SET UP	MISC. # TOOL #	ADDITIONAL COMMENTS
TOTAL HOURS→										←DAILY TOTALS			

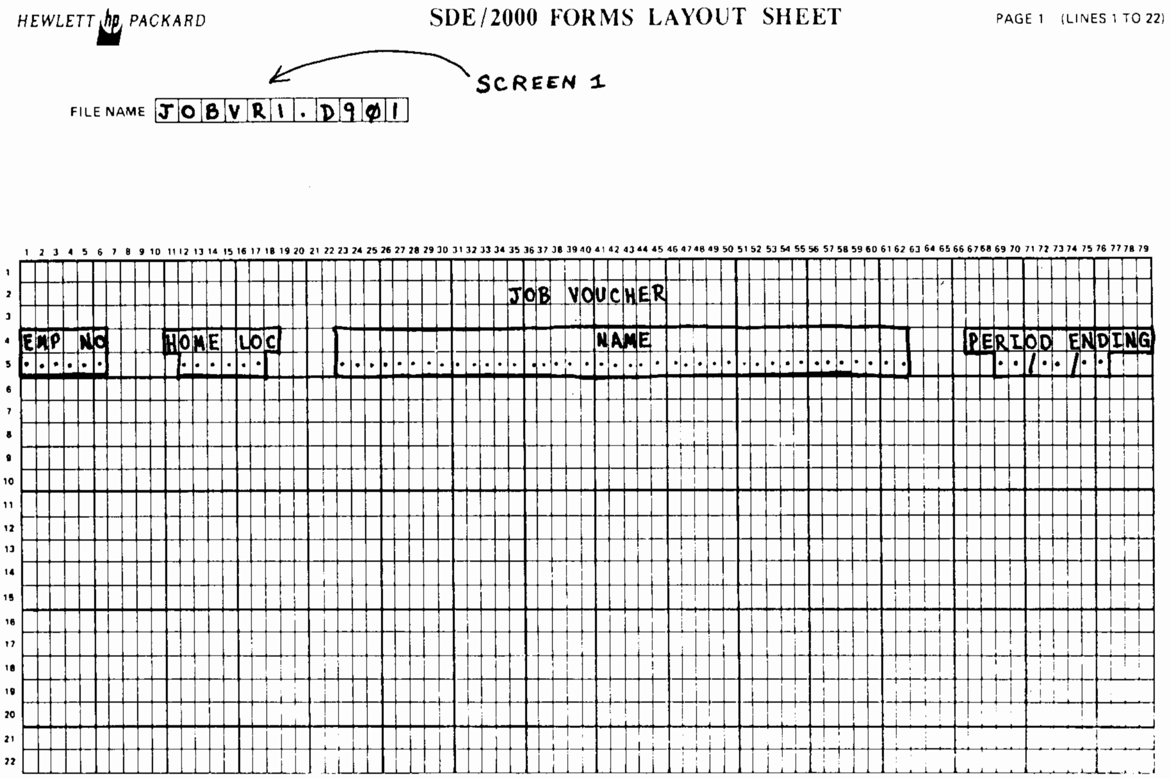
DEPARTMENTAL COPY

FORM 9320-2115 (CONTINUOUS)
FORM 9320-2117 (SNAPOUT)

FOR ADDITIONAL INSTRUCTIONS SEE BACK

JOB VOUCHER

Figure C-1. Job Voucher Form



HEWLETT hp PACKARD

SDE/2000 FORMS LAYOUT SHEET

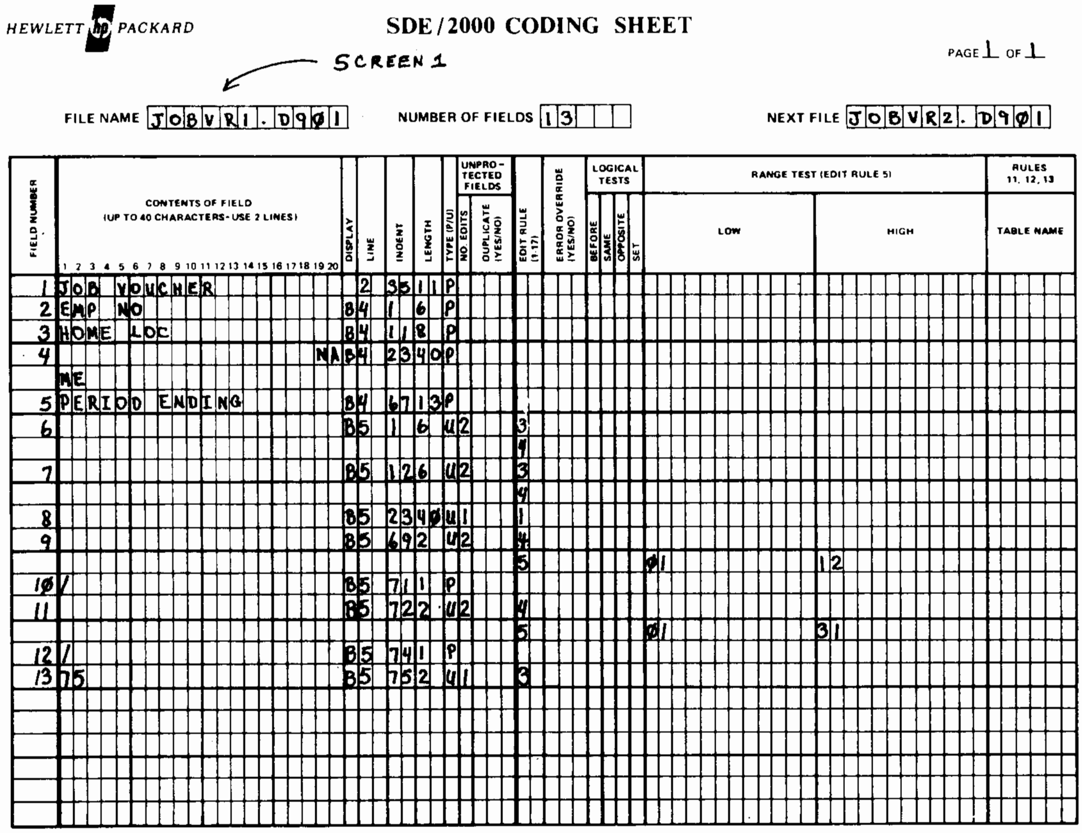
PAGE 1 (LINES 1 TO 22)

SCREEN 1

FILE NAME JOBVR1.D9Ø1

```
Line 2:                                    JOB VOUCHER
Line 4: EMP NO    HOME LOC                       NAME                                  PERIOD ENDING
Line 5: ......     ......     ........................................                    ../../..
```

Figure C-2. Forms Layout Sheet for Job Voucher, Screen 1



HEWLETT hp PACKARD

SDE/2000 CODING SHEET

PAGE 1 OF 1

SCREEN 1

FILE NAME JOBVR1.D9Ø1 NUMBER OF FIELDS 13 NEXT FILE JOBVR2.D9Ø1

FIELD NUMBER	CONTENTS OF FIELD (UP TO 40 CHARACTERS - USE 2 LINES)	DISPLAY	LINE	INDENT	LENGTH	TYPE (P/U)	UNPROTECTED FIELDS		EDIT RULE (1-17)	ERROR OVERRIDE (YES/NO)	LOGICAL TESTS				RANGE TEST (EDIT RULE 5)		RULES 11, 12, 13
							NO. EDITS	DUPLICATE (YES/NO)			BEFORE	SAME	OPPOSITE	SET	LOW	HIGH	TABLE NAME
1	JOB VOUCHER		2	35	11	P											
2	EMP NO	B	4	1	6	P											
3	HOME LOC	B	4	11	8	P											
4	NA	B	4	23	40	P											
	ME																
5	PERIOD ENDING	B	4	67	13	P											
6		B	5	1	6	U	2		3								
									4								
7		B	5	12	6	U	2		3								
									4								
8		B	5	23	4Ø	U	1		1								
9		B	5	69	2	U	2		4								
									5						Ø1	12	
1Ø	/	B	5	71	1	P											
11		B	5	72	2	U	2		4								
									5						Ø1	31	
12	/	B	5	74	1	P											
13	75	B	5	75	2	U	1		3								

Figure C-3. Coding Sheet for Job Voucher, Screen 1

HEWLETT PACKARD

SDE/2000 FORMS LAYOUT SHEET

PAGE 1 (LINES 1 TO 22)

SCREEN 2

FILE NAME JOBVR2.D9Ø1

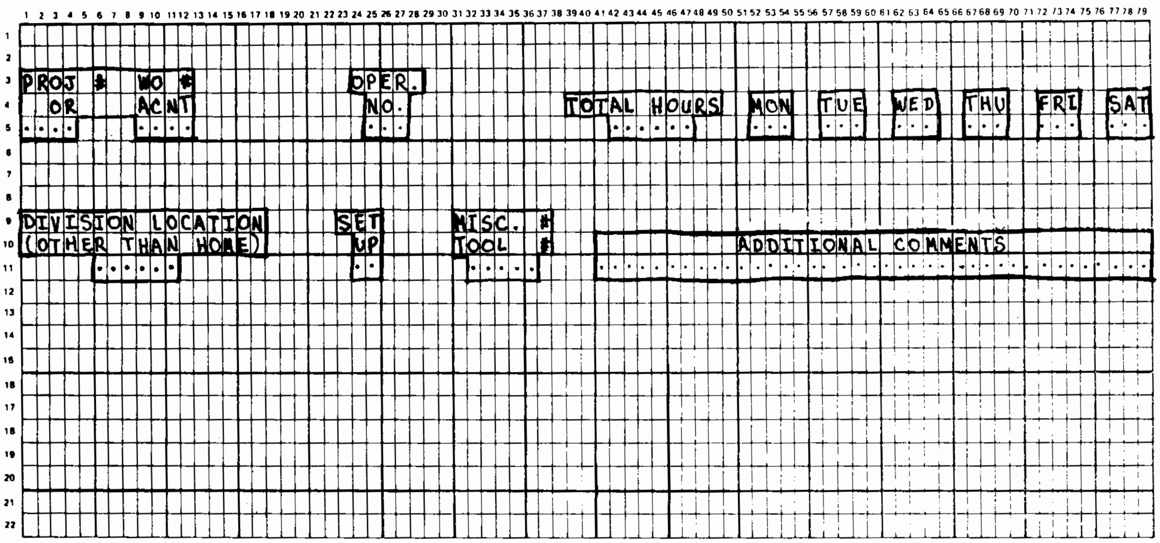


Figure C-4. Forms Layout Sheet for Job Voucher, Screen 2

HEWLETT PACKARD

SDE/2000 CODING SHEET

PAGE 1 OF 2

SCREEN 2

FILE NAME JOBVR2.D9Ø1 NUMBER OF FIELDS 32 NEXT FILE JOBVR3.D9Ø1

FIELD NUMBER	CONTENTS OF FIELD (UP TO 40 CHARACTERS - USE 2 LINES)	DISPLAY	LINE	INDENT	LENGTH	TYPE (P/U)	UNPROTECTED FIELDS NO. EDITS	UNPROTECTED FIELDS DUPLICATE (YES/NO)	EDIT RULE (1-17)	ERROR OVERRIDE (YES/NO)	LOGICAL TESTS BEFORE	LOGICAL TESTS SAME	LOGICAL TESTS OPPOSITE	LOGICAL TESTS SET	RANGE TEST (EDIT RULE 5) LOW	RANGE TEST (EDIT RULE 5) HIGH	RULES 11, 12, 13 TABLE NAME
1	PROJ # WO #	B	3	1	12	P											
2	OPER.	B	3	24	5	P											
3	OR ACNT	B	4	1	12	P											
4	NO.	B	4	25	3	P											
5	TOTAL HOURS	B	4	39	11	P											
6	MON	B	4	52	3	P											
7	TUE	B	4	57	3	P											
8	WED	B	4	62	3	P											
9	THU	B	4	67	3	P											
10	FRI	B	4	72	3	P											
11	SAT	B	4	77	3	P											
12		B	5	1	4	U	1		3								
13		B	5	9	4	U	1		3								
14		B	5	25	3	U	1		3								
15		B	5	42	6	U	1		3								
16		J	5	52	3	U	1		3								
17		J	5	57	3	U	1		3								
18		J	5	62	3	U	1		3								
19		J	5	67	3	U	1		3								
2Ø		J	5	72	3	U	1		3								
21		J	5	77	3	U	1		3								
22	DIVISION LOCATION	B	9	1	17	P											
23	SET	B	9	23	3	P											
24	MISC. #	B	9	31	7	P											

Figure C-5. Coding Sheet for Job Voucher, Screen 2, Fields 1 through 24

HEWLETT hp PACKARD

SDE/2000 CODING SHEET

PAGE 2 OF 2

SCREEN 2

FILE NAME JOBVR2.D9Ø1 NUMBER OF FIELDS 32 NEXT FILE JOBVR3.D9Ø1

FIELD NUMBER	CONTENTS OF FIELD (UP TO 40 CHARACTERS - USE 2 LINES)	DISPLAY	LINE	INDENT	LENGTH	TYPE (P/U)	UNPROTECTED FIELDS		EDIT RULE (1-17)	ERROR OVERRIDE (YES/NO)	LOGICAL TESTS				RANGE TEST (EDIT RULE 5)		RULES 11, 12, 13
							NO. EDITS	DUPLICATE (YES/NO)			BEFORE	SAME	OPPOSITE	SET	LOW	HIGH	TABLE NAME
25	(OTHER THAN HOME)	B	1Ø	1	17	P											
26	UP	B	1Ø	24	2	P											
27	TOOL #	B	1Ø	31	7	P											
28	ADDITIONAL	B	1Ø	41	39	P											
	COMMENTS																
29		B	11	6	6	U	1		3								
30		B	11	29	2	U	Ø										
31		B	11	32	5	U	1		3								
32		J	11	41	39	U	Ø										

Figure C-6. Coding Sheet for Job Voucher, Screen 2, Fields 25 through 32

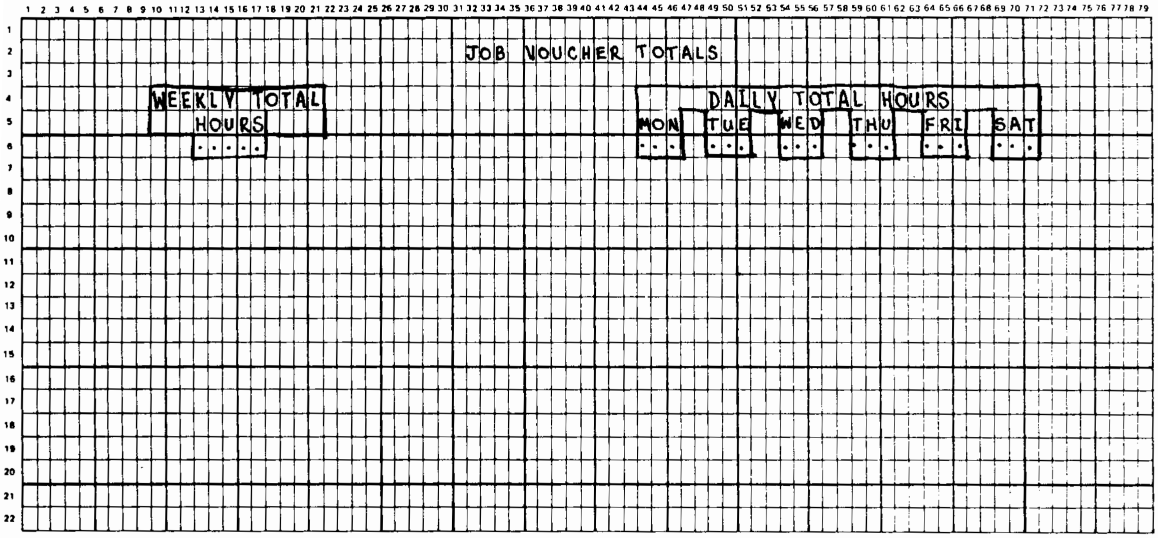
HEWLETT hp PACKARD

SDE/2000 FORMS LAYOUT SHEET

PAGE 1 (LINES 1 TO 22)

SCREEN 3

FILE NAME JOBVR3.D9Ø1



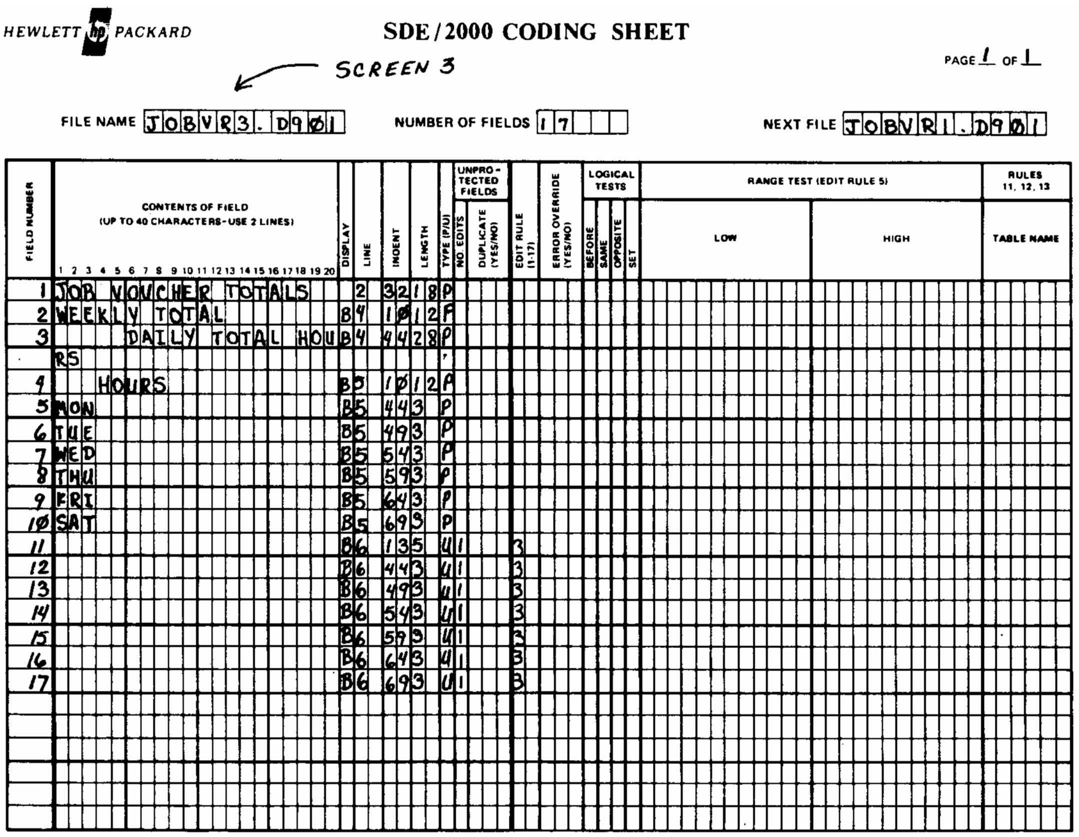
JOB VOUCHER TOTALS

WEEKLY TOTAL HOURS

DAILY TOTAL HOURS

MON TUE WED THU FRI SAT

Figure C-7. Forms Layout Sheet for Job Voucher, Screen 3



HEWLETT PACKARD

SDE/2000 CODING SHEET

PAGE 1 OF 1

SCREEN 3

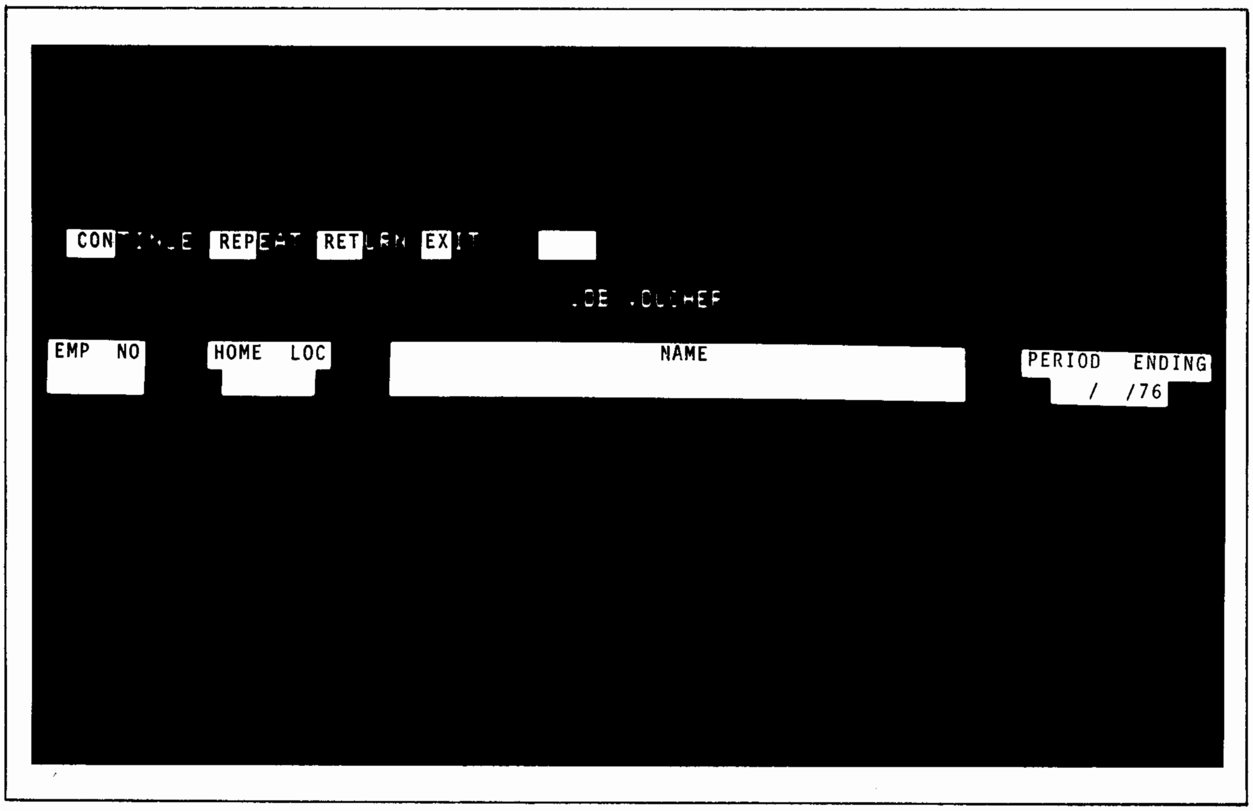
FILE NAME JOBVR3.D9Ø1

NUMBER OF FIELDS 17

NEXT FILE JOBVR1.D9Ø1

FIELD NUMBER	CONTENTS OF FIELD (UP TO 40 CHARACTERS-USE 2 LINES)	DISPLAY	LINE	INDENT	LENGTH	TYPE (P/U)	UNPROTECTED FIELDS		EDIT RULE (1-17)	ERROR OVERRIDE (YES/NO)	LOGICAL TESTS				RANGE TEST (EDIT RULE 5)		RULES 11, 12, 13
							NO. EDITS	DUPLICATE (YES/NO)			BEFORE	SAME	OPPOSITE	SET	LOW	HIGH	TABLE NAME
1	JOB VOUCHER TOTALS		2	32	18	P											
2	WEEKLY TOTAL	B	4	1Ø	12	F											
3	DAILY TOTAL HOU	B	4	44	28	P											
	RS																
4	HOURS	B	5	1Ø	12	P											
5	MON	B	5	44	3	P											
6	TUE	B	5	49	3	P											
7	WED	B	5	54	3	P											
8	THU	B	5	59	3	P											
9	FRI	B	5	64	3	P											
1Ø	SAT	B	5	69	3	P											
11		B	6	13	5	U	1		3								
12		B	6	44	3	U	1		3								
13		B	6	49	3	U	1		3								
14		B	6	54	3	U	1		3								
15		B	6	59	3	U	1		3								
16		B	6	64	3	U	1		3								
17		B	6	69	3	U	1		3								

Figure C-8. Coding Sheet for Job Voucher, Screen 3



JOB VOUCHER

EMP NO

HOME LOC

NAME

PERIOD ENDING
/ /76

Figure C-9. JOBVR1

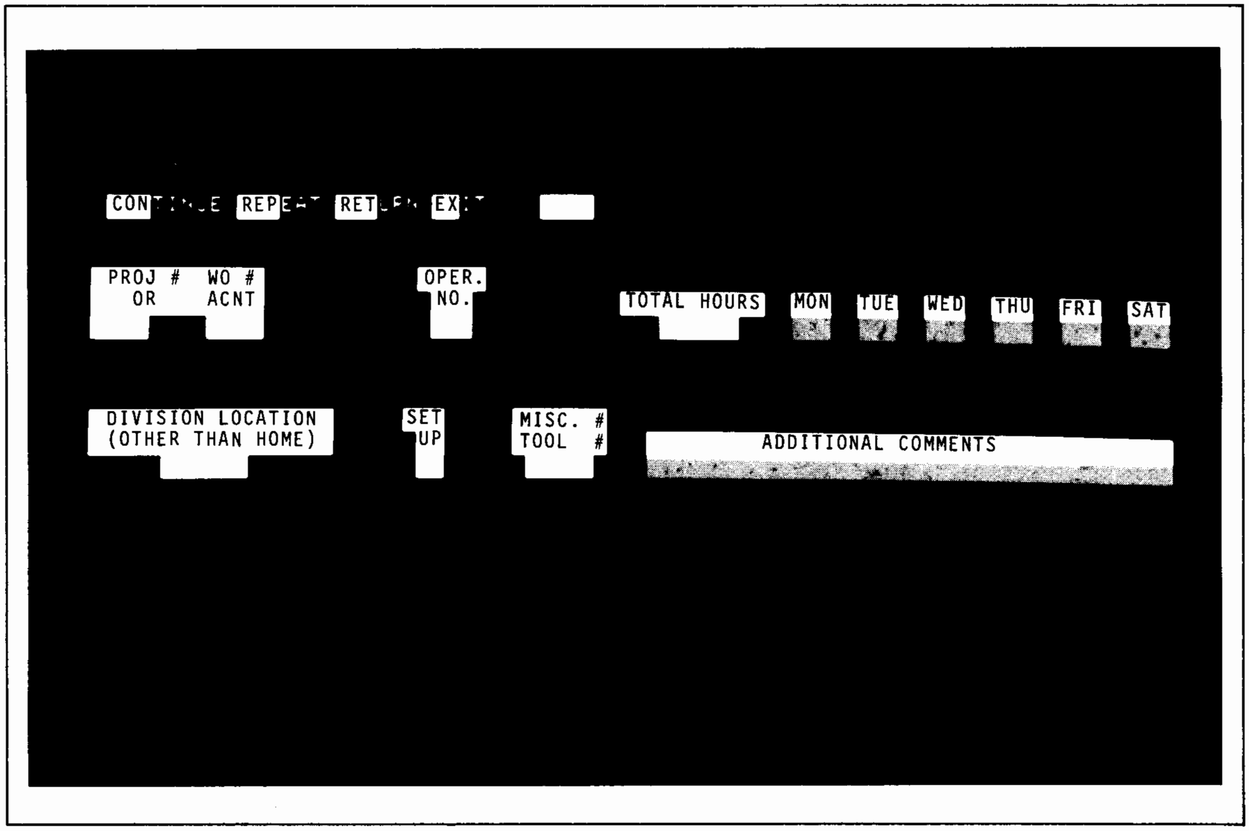


Figure C-10. JOBVR2

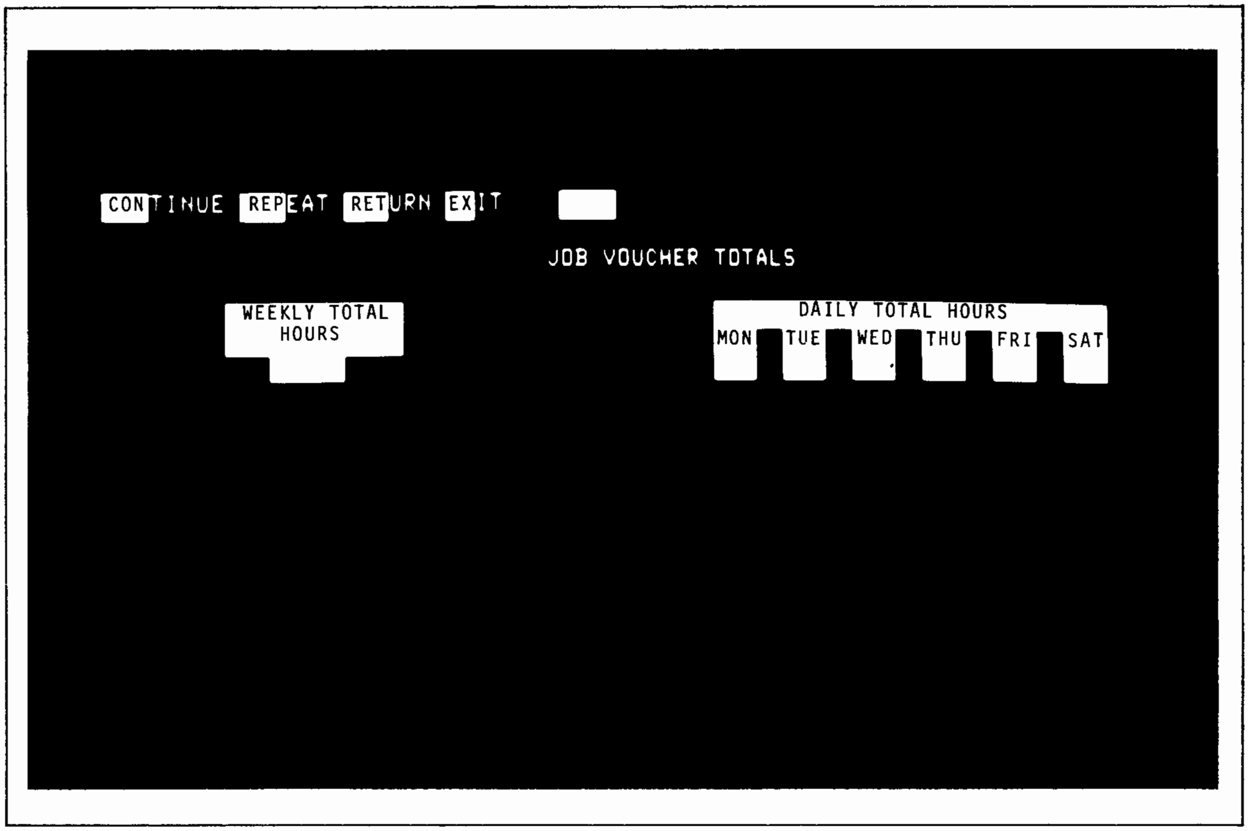
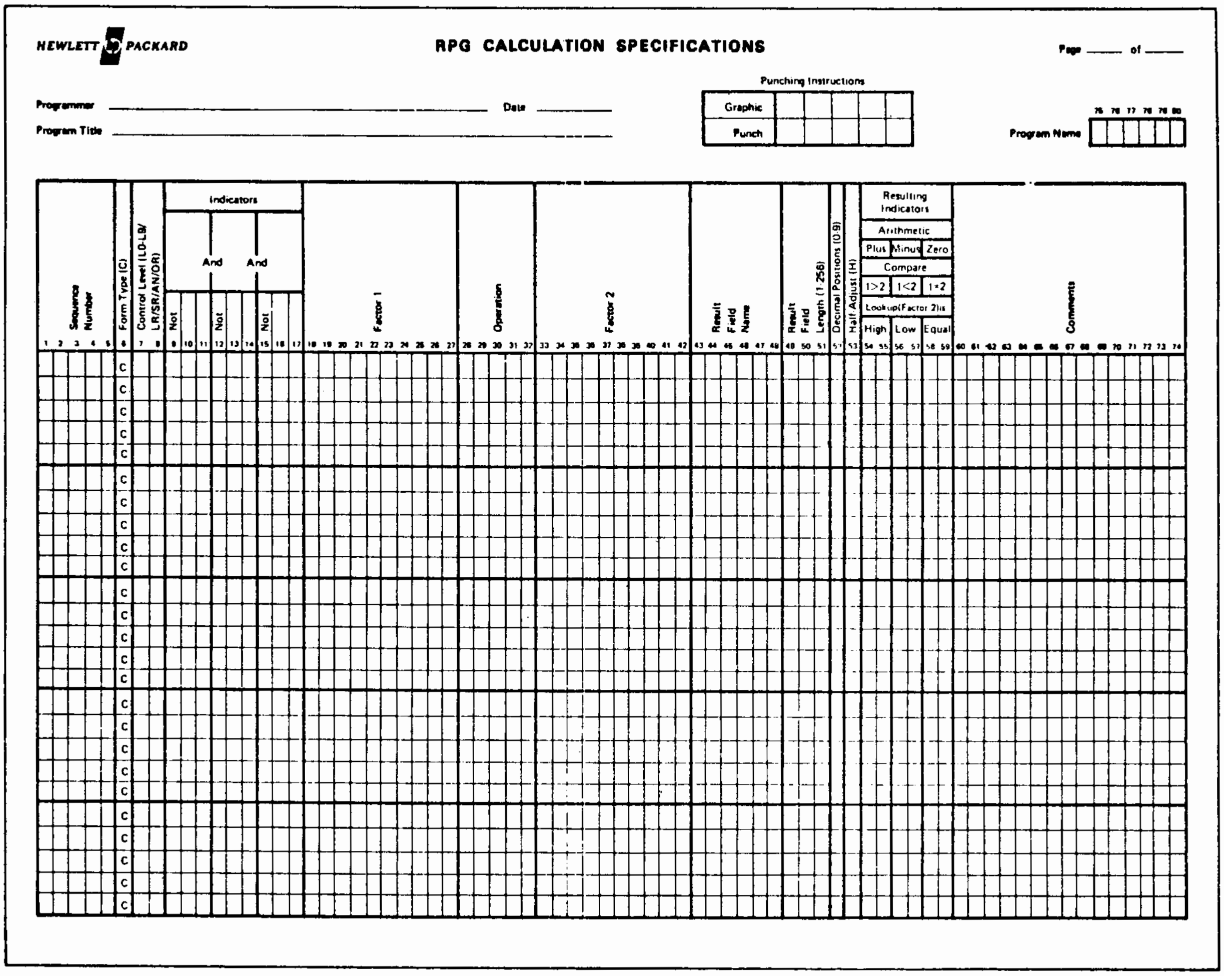


Figure C-11. JOBVR3

C-2. RPG CODING FORM

Figure C-12 shows the coding form for RPG Calculation Specifications. SDE/2000 can be used to accept data for this form, check it for errors, then write it to a disc file for Remote Job Entry to a large computer.

One possible way to arrange the fields is shown by the layout sheet, figure C-13. Many fields have a limited number of valid input values. Figure C-14 summarizes the acceptable range for each field.



HEWLETT PACKARD

RPG CALCULATION SPECIFICATIONS

Page ____ of ____

Programmer ____________________ Date ________

Program Title ____________________

Punching Instructions

Graphic					
Punch					

Program Name □□□□□□ (75 76 77 78 79 80)

Sequence Number	Form Type (C)	Control Level (L0-L9/ LR/SR/AN/OR)	Indicators: Not	And Not	And Not	Factor 1	Operation	Factor 2	Result Field Name	Result Field Length (1-256)	Decimal Positions (0-9)	Half Adjust (H)	Resulting Indicators: Arithmetic Plus / Compare 1>2 / Lookup(Factor 2)is High	Minus / 1<2 / Low	Zero / 1=2 / Equal	Comments
1-5	6	7-8	9-11	12-14	15-17	18-27	28-32	33-42	43-48	49-51	52	53	54-55	56-57	58-59	60-74
	C															
	C															
	C															
	C															
	C															
	C															
	C															
	C															
	C															
	C															
	C															
	C															
	C															
	C															
	C															
	C															
	C															
	C															
	C															
	C															
	C															
	C															
	C															
	C															
	C															

Figure C-12. Coding Form for RPG Calculation Specification

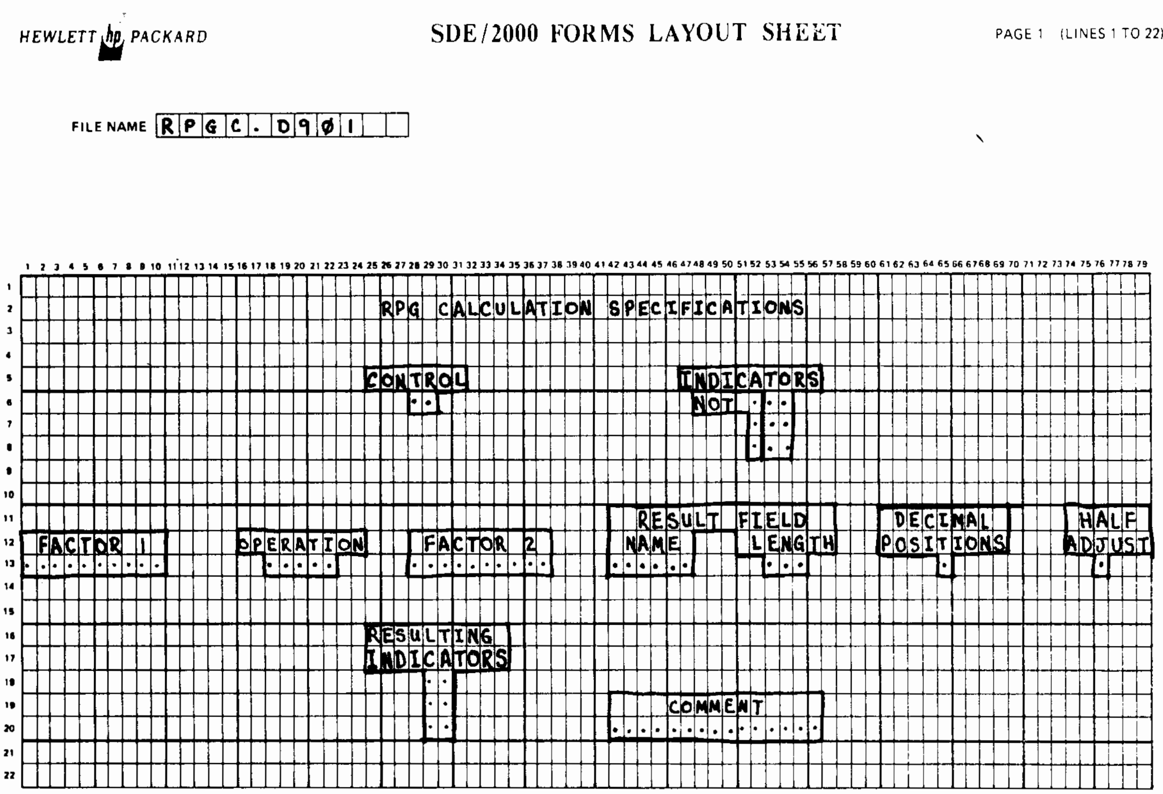


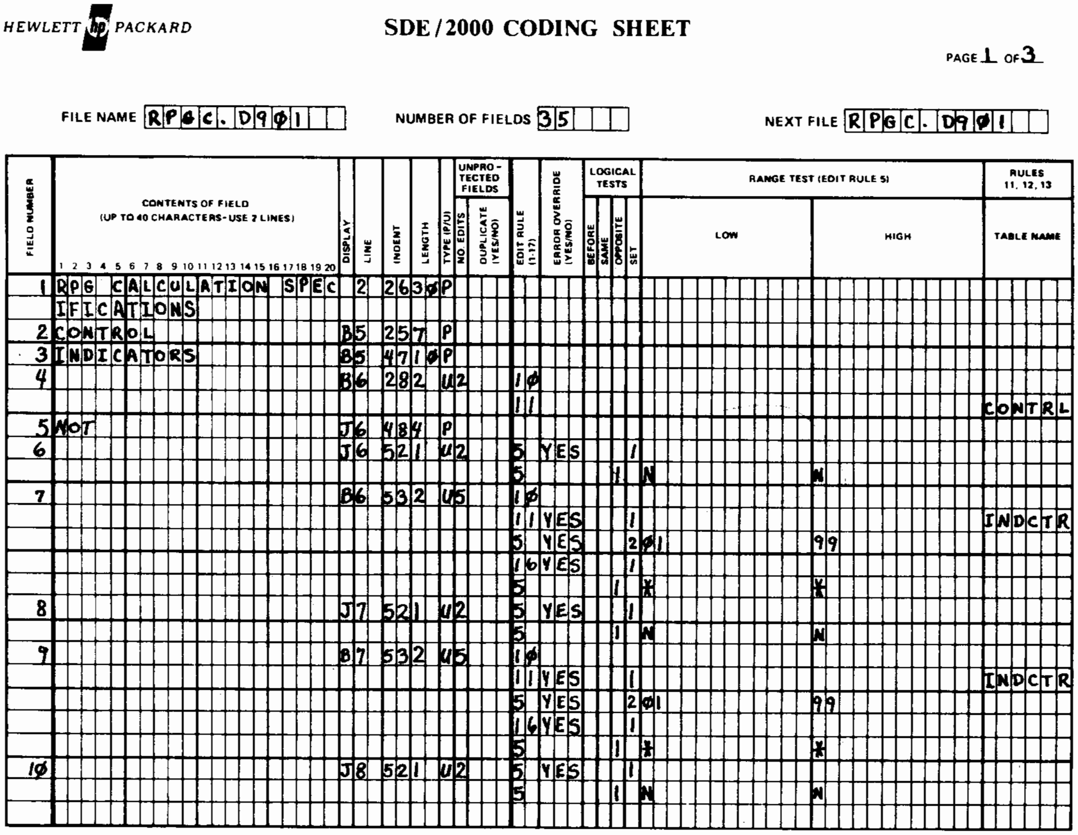
Figure C-13. Forms Layout Sheet for RPG Calculation Specifications

COLUMN(S)	FIELD NAME	ACCEPTED VALUE (any field may be left blank)
7-8	Control Level	L0-L9, LR, SR, AN, OR
9, 12, 15	Not	N
10-11, 13-14, 16-17	Indicators	01-99, L0-L9, LR, MR, H0-H9, U1-U8, OA-OG, 1P, *
18-27	Factor 1	(any value)
28-32	Operation	(any RPG calculation)
33-42	Factor 2	(any value)
43-48	Result Field Name	(any value)
49-51	Result Field Length	1-256, Right Justified
52	Decimal Positions	0-9
53	Half Adjust	H
54-55, 56-57	Resulting Indicators	01-99, L0-L9, LR, MR, H0-H9, U1-U8, OA-OG, 1P
60-74	Comment	(any value)

Figure C-14. RPG Calculation Specification Fields

The methods used to check each field are shown by the coding sheets in figures C-15 through C-17. To aid in checking, three tables have been set up to hold all the valid codes. These three tables, CONTRL, INDCTR, and OPERTN, are listed in figure C-18. (Because only the key values in the tables are used, the data fields are set to blanks.)

Look at the coding sheets and see how the fields are checked. Field 4 is checked by placing the input value in the search key, then searching for it in CONTRL. The same type of check is used in field 23 on the table OPERTN. However, fields 7, 9, and 11 must do more than a check on the table INDCTR: the field is valid if its value is in the table, if it is between 01 and 99, or if it is the character "*". The table is checked first, and the result of the search goes into flag 1. Then flag 2 is set with a check on the range 01-99. The OR function is performed on these two flags and the result placed in flag 1 (so that flag 1 is set to 1 if either condition was met). The final edit on the field performs a range check for the character "*". Because the "opposite from flag 1" option is specified, this edit produces an error message only when the range check fails and flag 1 is 0. The tests in fields 31 to 33 are similar, but the "*" character is not checked.



HEWLETT PACKARD

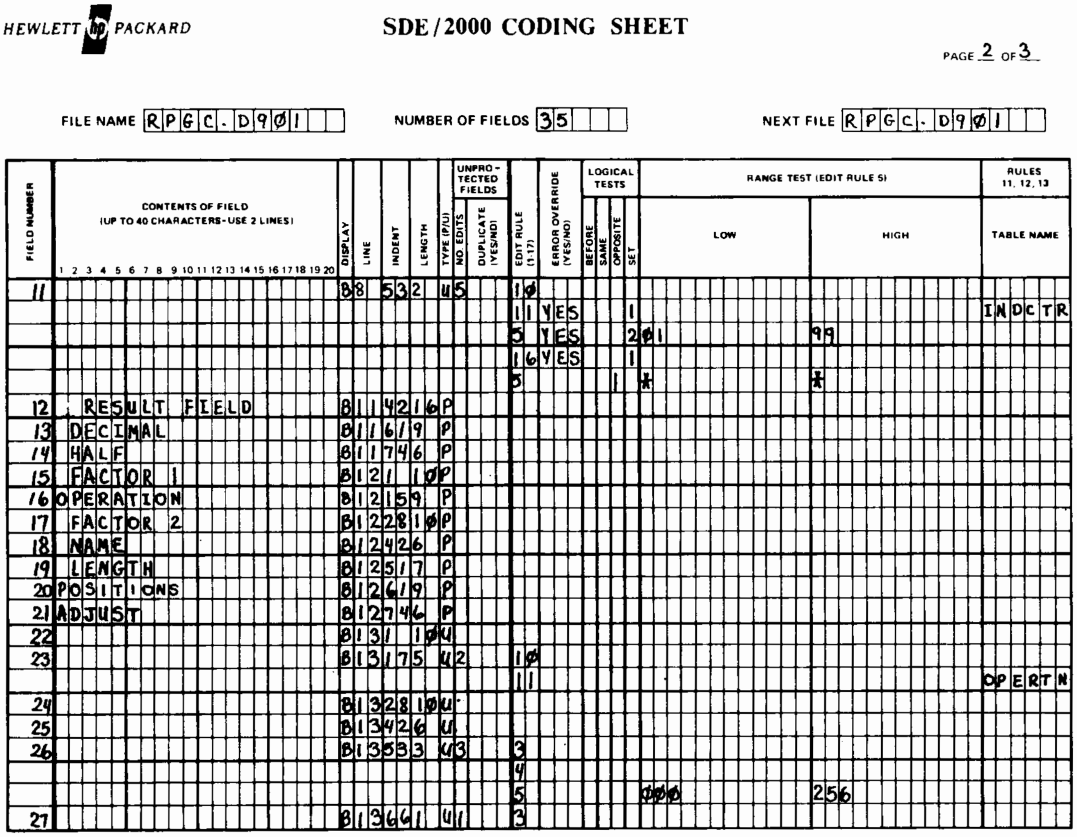
SDE/2000 CODING SHEET

PAGE 1 OF 3

FILE NAME RPGC. D9ø1 NUMBER OF FIELDS 35 NEXT FILE RPGC. D9ø1

FIELD NUMBER	CONTENTS OF FIELD (UP TO 40 CHARACTERS-USE 2 LINES) 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20	DISPLAY	LINE	INDENT	LENGTH	TYPE (P/U)	UNPROTECTED FIELDS		EDIT RULE (1-17)	ERROR OVERRIDE (YES/NO)	LOGICAL TESTS				RANGE TEST (EDIT RULE 5)		RULES 11, 12, 13
							NO. EDITS	DUPLICATE (YES/NO)			BEFORE	SAME	OPPOSITE	SET	LOW	HIGH	TABLE NAME
1	RPG CALCULATION SPEC		2	26	3ø	P											
	IFICATIONS																
2	CONTROL	B	5	25	7	P											
3	INDICATORS	B	5	47	1ø	P											
4		B	6	28	2	U	2		1ø								
									11								CONTRL
5	NOT	J	6	48	4	P											
6		J	6	52	1	U	2		5	YES				1			
									5				1		N	N	
7		B	6	53	2	U	5		1ø								
									11	YES				1			INDCTR
									5	YES				2	ø1	99	
									16	YES				1			
									5				1		*	*	
8		J	7	52	1	U	2		5	YES				1			
									5				1		N	N	
9		B	7	53	2	U	5		1ø								
									11	YES				1			INDCTR
									5	YES				2	ø1	99	
									16	YES				1			
									5				1		*	*	
1ø		J	8	52	1	U	2		5	YES				1			
									5				1		N	N	

Figure C-15. Coding Sheet, RPGC.D901, Fields 1 through 10



HEWLETT PACKARD

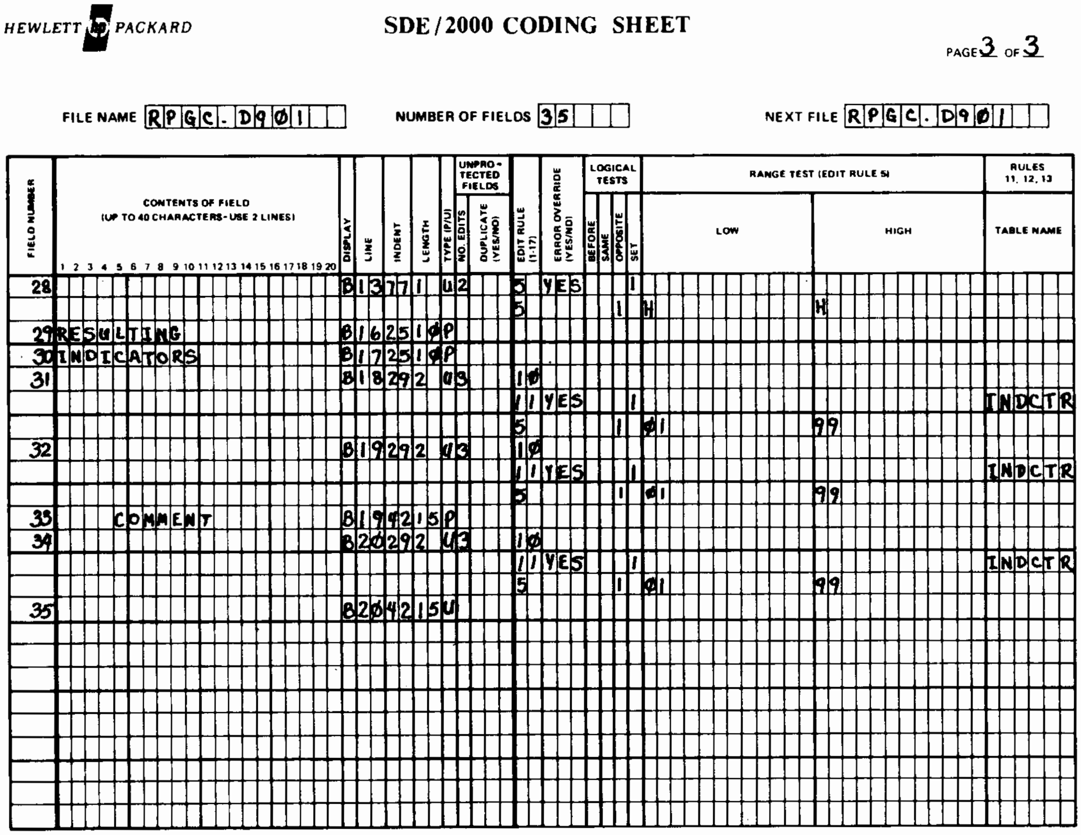
SDE/2000 CODING SHEET

PAGE 2 OF 3

FILE NAME RPGC.D9Ø1 NUMBER OF FIELDS 35 NEXT FILE RPGC.D9Ø1

							Unprotected fields				Logical tests				Range test (edit rule 5)		Rules 11, 12, 13
FIELD NUMBER	CONTENTS OF FIELD (UP TO 40 CHARACTERS-USE 2 LINES)	DISPLAY	LINE	INDENT	LENGTH	TYPE (P/U)	NO. EDITS	DUPLICATE (YES/NO)	EDIT RULE (1-17)	ERROR OVERRIDE (YES/NO)	BEFORE	SAME	OPPOSITE	SET	LOW	HIGH	TABLE NAME
11		B	8	53	2	U	5		1Ø								
									11	YES				1			INDCTR
									5	YES				2	Ø1	99	
									16	YES				1			
									5				1		*	*	
12	RESULT FIELD	B	11	42	16	P											
13	DECIMAL	B	11	61	9	P											
14	HALF	B	11	74	6	P											
15	FACTOR 1	B	12	1	1Ø	P											
16	OPERATION	B	12	15	9	P											
17	FACTOR 2	B	12	28	1Ø	P											
18	NAME	B	12	42	6	P											
19	LENGTH	B	12	51	7	P											
20	POSITIONS	B	12	61	9	P											
21	ADJUST	B	12	74	6	P											
22		B	13	1	1Ø	U											
23		B	13	17	5	U	2		1Ø								
									11								OPERTN
24		B	13	28	1Ø	U											
25		B	13	42	6	U											
26		B	13	53	3	U	3		3								
									4								
									5						ØØØ	256	
27		B	13	66	1	U	1		3								

Figure C-16. Coding Sheet, RPGC.D901, Fields 11 through 27



HEWLETT PACKARD

SDE/2000 CODING SHEET

PAGE 3 OF 3

FILE NAME RPGC.D9Ø1 NUMBER OF FIELDS 35 NEXT FILE RPGC.D9Ø1

							Unprotected fields				Logical tests				Range test (edit rule 5)		Rules 11, 12, 13
FIELD NUMBER	CONTENTS OF FIELD (UP TO 40 CHARACTERS-USE 2 LINES)	DISPLAY	LINE	INDENT	LENGTH	TYPE (P/U)	NO. EDITS	DUPLICATE (YES/NO)	EDIT RULE (1-17)	ERROR OVERRIDE (YES/NO)	BEFORE	SAME	OPPOSITE	SET	LOW	HIGH	TABLE NAME
28		B	13	77	1	U	2		5	YES				1			
									5				1		H	H	
29	RESULTING	B	16	25	1Ø	P											
30	INDICATORS	B	17	25	1Ø	P											
31		B	18	29	2	U	3		1Ø								
									11	YES				1			INDCTR
									5				1		Ø1	99	
32		B	19	29	2	U	3		1Ø								
									11	YES				1			INDCTR
									5				1		Ø1	99	
33	COMMENT	B	19	42	15	P											
34		B	2Ø	29	2	U	3		1Ø								
									11	YES				1			INDCTR
									5				1		Ø1	99	
35		B	2Ø	42	15	U											

Figure C-17. Coding Sheet, RPGC.D901, Fields 28 through 35

CONTRL	INDCTR		OPERTN	
AN	L0	H7	ADD	TESTZ
OR	L1	H8	SUB	ENDSR
SR	L2	H9	DIV	EXSR
LR	L3	U1	MULT	EXIT
L0	L4	U2	READ	MVR
L1	L5	U3	MOVE	PARM
L2	L6	U4	MOVEA	RLABL
L3	L7	U6	MOVEL	BITOF
L4	L8	U7	COMP	BITON
L5	L9	U8	CHAIN	BEGSR
L6	LR	OA	DISPLY	ERPGC
L7	MR	OB	GOTO	EXTCV
L8	H0	OC	LOKUP	FORCE
L9	H1	OD	TAG	MHHZO
	H2	OE	DEBUG	MHLZO
	H3	OF	SQRT	RPGCV
	H4	OG	Z-ADD	SETOF
	H5	OV	Z-SUB	SETON
	H6	1P	TESTB	MLLZO
		U5	TESTN	MLHZO
				XFOOT

Figure C-18. Tables Used by RPGC

Figure C-19 shows a program to take the individual records from disc and put them into an 80-character string variable as a card image. This program is designed to work like MYPROG in Section V: SDE chains to it, it copies the temporary file into its own file, then it chains back to SDE.

```
1   COM A,A$[11],AØ$[6],A1$[6],B$[6],BØ$[40],B1$[40],C$[4],CØ$[254]
2   COM C1$[254],D$[254],DØ$[12],D1$[40],E$[5],EØ$[40],E1$[6],F$[255]
3   COM AØ,A1,A2,A3,A4,A5,A6,A7,A8,A9,B,BØ,B1,B2,B3,B4,B5,B6,B7,B8
4   COM B9,C,CØ,C1,C2,C3,C4,C5,C6,C7,C8,C9,D,DØ,D1,D2,D3,D4,D5,D6
5   COM A[15ØØ],B[9],C[25]
1Ø  REM
2Ø  REM TAKE FIELDS FROM TEMP FILE AND PUT INTO STRING VARIABLE K$
3Ø  REM   THEN WRITE CARD IMAGE TO FILE "RPGCRD"
4Ø  REM
5Ø  DIM H1$[4Ø],K$[8Ø]
6Ø  FILES *,RPGCRD
7Ø  REM
8Ø  REM   ASSIGN TEMP FILE TO #1
9Ø  ASSIGN A1$,1,J
1ØØ  REM
11Ø  REM   INITIALIZE FIRST 6 COLUMNS OF CARD IMAGE
12Ø  K$="     C"
13Ø  REM
14Ø  REM   COLLATE FIELDS FROM TEMP FILE INTO K$
15Ø  IF  END #1 THEN 21Ø
16Ø  READ #1;T1,H1$
17Ø  K$[LEN(K$)+1]=H1$
18Ø  GOTO 16Ø
19Ø  REM
2ØØ  REM   WRITE CARD IMAGE TO RPGCRD
21Ø  K$[75,8Ø]="      "
22Ø  ADVANCE #2;32767,J
23Ø  PRINT #2;K$
24Ø  REM
25Ø  REM   SET RETURN FLAG AND GO BACK TO SDE
26Ø  A=1
27Ø  CHAIN "*SDEXØ1"
28Ø  END
```

Figure C-19. Sample Program to Write Card Image

Figure C-20 shows the completed RPGC form displayed on the 2640A terminal.

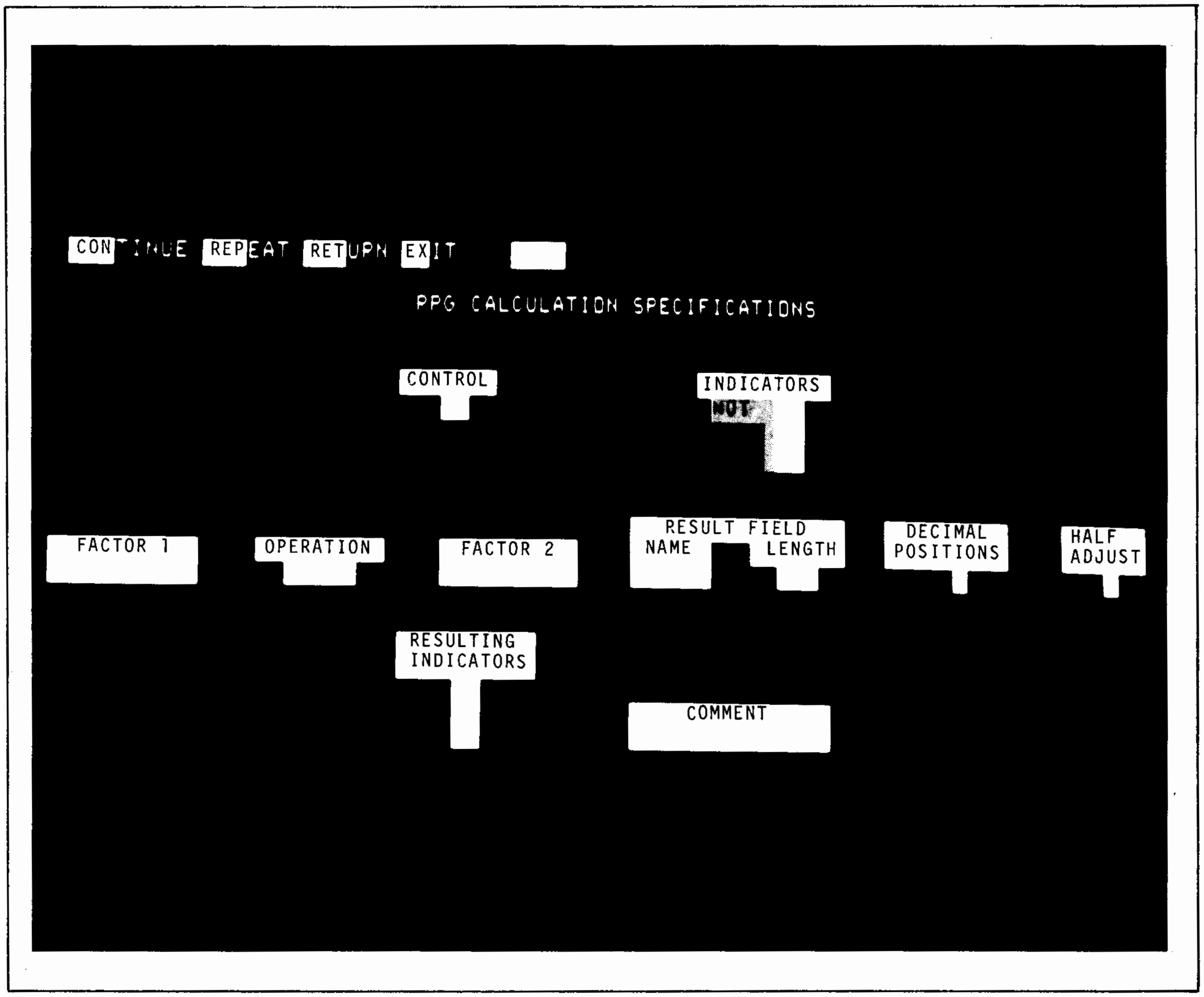


Figure C-20. RPGC

APPENDIX D

LAYOUT AND CODING SHEETS

The following Forms Layout sheets and Coding sheets are master copies which you can duplicate.

HEWLETT PACKARD

SDE/2000 FORMS LAYOUT SHEET

PAGE 1 (LINES 1 TO 22)

FILE NAME □□□□□□□□□□□

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
1																																																																																
2																																																																																
3																																																																																
4																																																																																
5																																																																																
6																																																																																
7																																																																																
8																																																																																
9																																																																																
10																																																																																
11																																																																																
12																																																																																
13																																																																																
14																																																																																
15																																																																																
16																																																																																
17																																																																																
18																																																																																
19																																																																																
20																																																																																
21																																																																																
22																																																																																

HEWLETT hp PACKARD

SDE/2000 CODING SHEET

PAGE ___ OF ___

FILE NAME □□□□□□□□□□□ NUMBER OF FIELDS □□□□□ NEXT FILE □□□□□□□□□□□

FIELD NUMBER	CONTENTS OF FIELD (UP TO 40 CHARACTERS-USE 2 LINES)	DISPLAY	LINE	INDENT	LENGTH	TYPE (P/U)	UNPROTECTED FIELDS		EDIT RULE (1-17)	ERROR OVERRIDE (YES/NO)	LOGICAL TESTS				RANGE TEST (EDIT RULE 5)		RULES 11, 12, 13
	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20						NO. EDITS	DUPLICATE (YES/NO)			BEFORE	SAME	OPPOSITE	SET	LOW	HIGH	TABLE NAME

HEWLETT PACKARD

SDE/2000 FORMS LAYOUT SHEET

PAGE 2 (LINES 25 TO 46)

FILE NAME □□□□□□□□□□□

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
25																																																																															
26																																																																															
27																																																																															
28																																																																															
29																																																																															
30																																																																															
31																																																																															
32																																																																															
33																																																																															
34																																																																															
35																																																																															
36																																																																															
37																																																																															
38																																																																															
39																																																																															
40																																																																															
41																																																																															
42																																																																															
43																																																																															
44																																																																															
45																																																																															
46																																																																															

HEWLETT PACKARD

SDE/2000 FORMS LAYOUT SHEET

PAGE 3 (LINES 49 TO 70)

FILE NAME □□□□□□□□□□□

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
49																																																																															
50																																																																															
51																																																																															
52																																																																															
53																																																																															
54																																																																															
55																																																																															
56																																																																															
57																																																																															
58																																																																															
59																																																																															
60																																																																															
61																																																																															
62																																																																															
63																																																																															
64																																																																															
65																																																																															
66																																																																															
67																																																																															
68																																																																															
69																																																																															
70																																																																															

INDEX

Part No. 20243-90001
Product No. 20243A
Printed in U.S.A. 2/76

HEWLETT hp PACKARD

Sales and service from 172 offices in 65 countries.
5303 Stevens Creek Blvd., Santa Clara, California 95050