

January 1980

FMS-11/RSX Release Notes

Order No. AA-H857A-TC

SUPERSESSON/UPDATE INFORMATION: This is a new manual.

OPERATING SYSTEM AND VERSION: RSX-11M V3.2
RSX-11M-PLUS V1

To order additional copies of this document, contact the Software Distribution Center, Digital Equipment Corporation, Maynard, Massachusetts 01754

digital equipment corporation • maynard, massachusetts

First Printing, January 1980

The information in this document is subject to change without notice and should not be construed as a commitment by Digital Equipment Corporation. Digital Equipment Corporation assumes no responsibility for any errors that may appear in this document.

The software described in this document is furnished under a license and may only be used or copied in accordance with the terms of such license.

No responsibility is assumed for the use or reliability of software on equipment that is not supplied by DIGITAL or its affiliated companies.

Copyright © 1980 by Digital Equipment Corporation

The postage-prepaid READER'S COMMENTS form on the last page of this document requests the user's critical evaluation to assist us in preparing future documentation.

The following are trademarks of Digital Equipment Corporation:

DIGITAL	DECsystem-10	MASSBUS
DEC	DECTape	OMNIBUS
PDP	DIBOL	OS/8
DECUS	EDUSYSTEM	PHA
UNIBUS	FLIP CHIP	RSTS
COMPUTER LABS	FOCAL	RSX
COMTEX	INDAC	TYPESET-8
DDT	LAB-8	TYPESET-11
DECCOMM	DECSYSTEM-20	TMS-11
ASSIST-11	RTS-8	ITPS-10
VAX	VMS	SBI
DECnet	IAS	PDT
DATATRIEVE	TRAX	

CONTENTS

	Page
1.0 INTRODUCTION	1
1.1 Using FMS-11/RSX	1
1.2 The FMS-11/RSX Documentation Set	1
2.0 INSTALLING FMS-11/RSX	1
2.1 Installation Procedure	2
2.2 Verifying Installation Procedures	3
3.0 USER ENVIRONMENT TEST PACKAGE (UETP)	4
3.1 Running the UETP	4
3.2 UETP Step-by-Step Procedure	5
4.0 VT52 SUPPORT	10
5.0 FORM DRIVER RESIDENT LIBRARY	11

1.0 INTRODUCTION

This document describes the installation procedures for FMS-11/RSX software, the FMS-11/RSX User Environment Test Package (UETP), information about using FMS-11 on a VT52 terminal, and information about the Form Driver Resident Library.

1.1 Using FMS-11/RSX

The following steps summarize how to use your FMS-11/RSX software:

1. Follow the installation procedures in Section 2 of this document.
2. Verify that your hardware and software are working properly by running the demonstration programs provided and by completing the UETP procedure described in Section 3 of this document.
3. Read the FMS-11/RSX Software Reference Manual to learn to use the FMS-11/RSX software components.
4. Use the extended examples that are referenced in the FMS-11/RSX Software Reference Manual to become acquainted with the software.
5. Write your own small FMS-11 application.

1.2 The FMS-11/RSX Documentation Set

The three documents for the FMS-11/RSX software and their order numbers are:

1. FMS-11/RSX Software Reference Manual AA-H855A-TC
2. FMS-11/RSX Mini-Reference AV-H856A-TC
3. FMS-11/RSX Release Notes AA-H857A-TC

2.0 INSTALLING FMS-11/RSX

The procedure for installing FMS-11 on an RSX-11M or an RSX-11M-PLUS system consists of copying the files from the distribution media to the device where they will be used and building the FMS-11 components. The entire process is controlled by an indirect command file.

When the command files finish executing, the FMS utilities can be copied to the system account and installed with the MCR INSTALL command.

You must be logged in under a privileged account in order for the installation to successfully complete. If you are installing FMS-11 on an RSX-11M-PLUS system, your terminal must be set for MCR mode commands. It is assumed that FMS-11 is being installed on the system device. All FMS-11 files are moved into account [30,10]. A VT100 is required to run the Form Editor and the demonstration programs and the UETP built as part of the installation procedure.

Approximately 3600 disk blocks are required to install FMS-11/RSX if all the demonstration programs are built. The demonstration program in a particular language is built only if the corresponding language processor is installed in the system.

In the installation procedure detailed below, the parameter 'ddn' is the name and unit number for the device FMS-11 is being installed on. The parameter 'dev' is the name and unit number of the device on which the distribution media is mounted.

2.1 Installation Procedure

```
MCR>ASN ddn:=SY:           ! Make the installation device
                           ! the default system device
MCR>UFD ddn:[30,10]       ! Create the UFD for FMS-11
MCR>SET /UIC=[30,10]     ! Set the default UIC
```

For magnetic tapes on RSX-11M:

```
MCR>ALL dev:              ! Allocate the drive
MCR>FLX SY:=dev:FMSINS.CMD/DO ! Copy installation command file
```

For magnetic tapes on RSX-11M-PLUS:

```
MCR>MOU dev:/FOR         ! Mount tape as foreign device
MCR>FLX SY:=dev:FMSINS.CMD/DO ! Copy installation command file
```

For disks on both systems:

```
MCR>MOU dev:FMSRSX      ! Mount the disk
MCR>PIP SY:/NV=dev:FMSINS.CMD ! Copy installation command file
```

For all media:

```
MCR>@FMSINS             ! Copy files from distribution
                           ! media and build utilities
                           ! and demo programs
```

The installation command file prompts for the distribution device. The FMS-11 files are copied to account [30,10] on the system device. The Form Editor (FED) and the Form Utility (FUT) are task built. The MACRO versions of the demonstration program and the UETP are built automatically. The COBOL, BASIC-PLUS-2, FORTRAN IV, and FORTRAN IV-PLUS versions of the demonstration program are built only if the corresponding compilers are installed in the system. The same is true for the FORTRAN IV and FORTRAN IV-PLUS versions of the UETP. If the FORTRAN IV-PLUS compiler is installed in the system, it is assumed that the file LB:[1,1]F4POTS.OLB exists as the FORTRAN IV-PLUS OTS. The installation command file asks the user if this is the case. If not, the FORTRAN IV-PLUS versions of the programs cannot be built. Otherwise, the command file queries the user as to whether FCS or RMS support is provided. The FORTRAN IV-PLUS demonstration program is then task built accordingly. The UETP program is only built if FCS support is present.

Two prebuilt Form Driver libraries are distributed with FMS-11: an object library using FCS (FDVLIB.OLB) and an object library using RMS (FDVLRM.OLB). The distributed versions of the Form Driver provide support for the VT100 only and require the full duplex terminal driver. Support for all features other than debug mode is included. If you wish to reconfigure the Form Driver, tailoring it to specific application requirements, execute the command file FDVBLD.CMD.

The system on which FED and FUT are run must include checkpointing to a system checkpoint file and the full duplex terminal driver.

2.2 Verifying Installation Procedures

The installation procedure is almost self-verifying. As part of the installation, demonstration programs are compiled and task built if the appropriate language processors are installed in the system. The programs MACDEM, FORDEM, F4PDEM, BASDEM and CBLDEM all implement the same application in MACRO, FORTRAN IV, FORTRAN IV-PLUS, BASIC-PLUS-2, and COBOL. Listings of these programs and a detailed description of the application are included in Appendix B of the FMS-11/RSX Software Reference Manual. To verify that these programs operate correctly, you should run them after the installation is complete. For example, to run the MACRO version of the demonstration program type the following:

```
MCR>SET /UIC=[30,10]
MCR>RUN MACDEM
```

Note that any time one of the demonstration programs distributed with FMS-11 is run (including the UETP), it is necessary that the default UIC be set to the account containing the form library and data files required by the program.

The demonstration programs implement a menu-driven application that allows you to enter customer information, parts descriptions, and employee information. The initial menu form allows the user to select one of the three functions or to exit. When the detail form(s) pertaining to a function have been completed, a second menu form is displayed. At this point the options are to repeat the same function, to return to the initial menu form, or to exit from the program. Each time a function is initiated, a file is created and the data collected is written sequentially into the file. The names of the files correspond to the functions: NEWCUS.DAT for customer information, PARTS.DAT for parts descriptions, and EMPLOY.DAT for employee information. You may inspect these files with an editor to verify the data.

To verify that the Form Utility (FUT) was built successfully, use it to get a directory of one of the form libraries distributed with FMS-11 and a listing of a form description. The following sample session with FUT gives a directory of the form library DEMLIB.FLB and a listing of the form PARTS contained in that library.

```
MCR>SET /UIC=[30,10]
MCR>RUN FUT
FUT>DEMLIB.FLB/LI
```

```
FUT V01.00
4-DEC-79
```

```
Library DL0:[30,10]DEMLIB.FLB;1 created: 4-DEC-79
Directory is 1 block long.
```

<u>Form</u>	<u>Date</u>	<u>Impure area (bytes)</u>
FIRST	4-DEC-79	369
CUSTPR	4-DEC-79	326
LAST	4-DEC-79	275
EMPLOY	4-DEC-79	812
PARTS	4-DEC-79	794
CUSTO	4-DEC-79	612

FUT>PARTS=DEMLIB.FLB/FD

DL0:[30,10]DEMLIB.FLB Form name? PARTS

DL0:[30,10]DEMLIB.FLB Form name? <CR>

FUT>

The file [30,10]PARTS.FMD contains the form description for the form PARTS. The file may be spooled to a line printer to obtain a hard copy listing.

To verify that the Form Editor (FED) was built correctly, use the example provided in Section 2.7 of the FMS-11/RSX Software Reference Manual to create a new form and modify an existing form.

3.0 USER ENVIRONMENT TEST PACKAGE (UETP)

The User Environment Test Package (UETP) for FMS-11/RSX is designed as an integrity test of the Form Management System. The intent is to verify that a typical FMS-11 application can be built and executed with the installed software. The UETP application is a Simple Inventory System, referred to throughout the remainder of this document as SIS.

The UETP is built as part of the installation procedure. The MACRO version (SIS.TSK) is built automatically. The FORTRAN IV (SISF.TSK) and FORTRAN IV-PLUS (SISF4P.TSK) versions are built only if the corresponding compilers are installed in the system. The UETP requires FORTRAN IV-PLUS with FCS support for file I/O. Therefore, a FORTRAN IV-PLUS version of the UETP is only built if the FORTRAN IV-PLUS OTS on the system (LB:[1,1]F4POTS.OLB) provides support for FCS rather than RMS.

The UETP is built to include a script processor for automatic execution. The script provided is designed to exercise a majority of the capabilities of the Form Driver component of FMS-11.

3.1 Running the UETP

To run the UETP, set the default UIC to [30,10]. Initialize the necessary application files by running the initialization program as follows:

MCR>RUN INITF

You can then run either the MACRO or one of the FORTRAN versions of the UETP. To run the FORTRAN IV version, for example, type

MCR>RUN SISF

Since the program executes under script control, character input is taken from the script file rather than the terminal keyboard. Script characters are supplied every half second to simulate human typing. Some delays have been included in the script to facilitate reading. The progress of the script can be controlled from the keyboard. Type 'S' to temporarily halt the script; type 'G' to resume the script. After halting the script by typing 'S', you can step through it one character at a time by typing <SPACE>. Since escape sequences are several characters long, it is necessary in some cases to step through more than one character before any action is taken by the Form Driver. The script can be aborted at any time by typing 'Z' (not control Z). After aborting the script, all subsequent input is taken from the keyboard rather than the file. Should you abort in the middle of an escape sequence, the Form Driver may signal an error (by sounding the bell) in response to the first few characters entered manually, but it should not persist.

3.2 UETP Step-by-Step Procedure

The following description of the execution of the script is divided so that each step corresponds to a single form displayed by the application. Within each step, the input from the script and the resulting actions are described.

STEP 1 - SIS INTRODUCTION

1. The script types the <HELP> key (the "PF2" key on the keypad) and a one line explanation of how to respond to the prompt NEXT? is displayed on the last line of the screen.
2. The script types <HELP> again, and FDV displays the HELP form that is associated with the introduction form.

STEP 2 - INTRODUCTION HELP FORM

The script types the <RETURN> key, and FDV displays the introduction form again. The cursor is at the NEXT? prompt.

STEP 3 - SIS INTRODUCTION

The script accepts the default response (4) to request a correction form. SIS processes the response, and the correction form is displayed.

STEP 4 - CORRECTION REQUEST

The script overrides the default response (N) and types D to request the form for changing inventory descriptions. SIS processes the response and displays the requested form.

STEP 5 - CHANGE DESCRIPTION

1. The script types a stock number, advances to the next input field ("Description") by typing the <TAB> key, and types the name of the item.
2. The script types in several new descriptions, advancing from field to field by using the <TAB> key, and types <RETURN> when the form is complete.
3. SIS processes the entries and displays the next form, a menu of SIS processes.

Since the program executes under script control, character input is taken from the script file rather than the terminal keyboard. Script characters are supplied every half second to simulate human typing. Some delays have been included in the script to facilitate reading. The progress of the script can be controlled from the keyboard. Type 'S' to temporarily halt the script; type 'G' to resume the script. After halting the script by typing 'S', you can step through it one character at a time by typing <SPACE>. Since escape sequences are several characters long, it is necessary in some cases to step through more than one character before any action is taken by the Form Driver. The script can be aborted at any time by typing 'Z' (not control Z). After aborting the script, all subsequent input is taken from the keyboard rather than the file. Should you abort in the middle of an escape sequence, the Form Driver may signal an error (by sounding the bell) in response to the first few characters entered manually, but it should not persist.

3.2 UETP Step-by-Step Procedure

The following description of the execution of the script is divided so that each step corresponds to a single form displayed by the application. Within each step, the input from the script and the resulting actions are described.

STEP 1 - SIS INTRODUCTION

1. The script types the <HELP> key (the "PF2" key on the keypad) and a one line explanation of how to respond to the prompt NEXT? is displayed on the last line of the screen.
2. The script types <HELP> again, and FDV displays the HELP form that is associated with the introduction form.

STEP 2 - INTRODUCTION HELP FORM

The script types the <RETURN> key, and FDV displays the introduction form again. The cursor is at the NEXT? prompt.

STEP 3 - SIS INTRODUCTION

The script accepts the default response (4) to request a correction form. SIS processes the response, and the correction form is displayed.

STEP 4 - CORRECTION REQUEST

The script overrides the default response (N) and types D to request the form for changing inventory descriptions. SIS processes the response and displays the requested form.

STEP 5 - CHANGE DESCRIPTION

1. The script types a stock number, advances to the next input field ("Description") by typing the <TAB> key, and types the name of the item.
2. The script types in several new descriptions, advancing from field to field by using the <TAB> key, and types <RETURN> when the form is complete.
3. SIS processes the entries and displays the next form, a menu of SIS processes.

Since the program executes under script control, character input is taken from the script file rather than the terminal keyboard. Script characters are supplied every half second to simulate human typing. Some delays have been included in the script to facilitate reading. The progress of the script can be controlled from the keyboard. Type 'S' to temporarily halt the script; type 'G' to resume the script. After halting the script by typing 'S', you can step through it one character at a time by typing <SPACE>. Since escape sequences are several characters long, it is necessary in some cases to step through more than one character before any action is taken by the Form Driver. The script can be aborted at any time by typing 'Z' (not control Z). After aborting the script, all subsequent input is taken from the keyboard rather than the file. Should you abort in the middle of an escape sequence, the Form Driver may signal an error (by sounding the bell) in response to the first few characters entered manually, but it should not persist.

3.2 UETP Step-by-Step Procedure

The following description of the execution of the script is divided so that each step corresponds to a single form displayed by the application. Within each step, the input from the script and the resulting actions are described.

STEP 1 - SIS INTRODUCTION

1. The script types the <HELP> key (the "PF2" key on the keypad) and a one line explanation of how to respond to the prompt NEXT? is displayed on the last line of the screen.
2. The script types <HELP> again, and FDV displays the HELP form that is associated with the introduction form.

STEP 2 - INTRODUCTION HELP FORM

The script types the <RETURN> key, and FDV displays the introduction form again. The cursor is at the NEXT? prompt.

STEP 3 - SIS INTRODUCTION

The script accepts the default response (4) to request a correction form. SIS processes the response, and the correction form is displayed.

STEP 4 - CORRECTION REQUEST

The script overrides the default response (N) and types D to request the form for changing inventory descriptions. SIS processes the response and displays the requested form.

STEP 5 - CHANGE DESCRIPTION

1. The script types a stock number, advances to the next input field ("Description") by typing the <TAB> key, and types the name of the item.
2. The script types in several new descriptions, advancing from field to field by using the <TAB> key, and types <RETURN> when the form is complete.
3. SIS processes the entries and displays the next form, a menu of SIS processes.

Since the program executes under script control, character input is taken from the script file rather than the terminal keyboard. Script characters are supplied every half second to simulate human typing. Some delays have been included in the script to facilitate reading. The progress of the script can be controlled from the keyboard. Type 'S' to temporarily halt the script; type 'G' to resume the script. After halting the script by typing 'S', you can step through it one character at a time by typing <SPACE>. Since escape sequences are several characters long, it is necessary in some cases to step through more than one character before any action is taken by the Form Driver. The script can be aborted at any time by typing 'Z' (not control Z). After aborting the script, all subsequent input is taken from the keyboard rather than the file. Should you abort in the middle of an escape sequence, the Form Driver may signal an error (by sounding the bell) in response to the first few characters entered manually, but it should not persist.

3.2 UETP Step-by-Step Procedure

The following description of the execution of the script is divided so that each step corresponds to a single form displayed by the application. Within each step, the input from the script and the resulting actions are described.

STEP 1 - SIS INTRODUCTION

1. The script types the <HELP> key (the "PF2" key on the keypad) and a one line explanation of how to respond to the prompt NEXT? is displayed on the last line of the screen.
2. The script types <HELP> again, and FDV displays the HELP form that is associated with the introduction form.

STEP 2 - INTRODUCTION HELP FORM

The script types the <RETURN> key, and FDV displays the introduction form again. The cursor is at the NEXT? prompt.

STEP 3 - SIS INTRODUCTION

The script accepts the default response (4) to request a correction form. SIS processes the response, and the correction form is displayed.

STEP 4 - CORRECTION REQUEST

The script overrides the default response (N) and types D to request the form for changing inventory descriptions. SIS processes the response and displays the requested form.

STEP 5 - CHANGE DESCRIPTION

1. The script types a stock number, advances to the next input field ("Description") by typing the <TAB> key, and types the name of the item.
2. The script types in several new descriptions, advancing from field to field by using the <TAB> key, and types <RETURN> when the form is complete.
3. SIS processes the entries and displays the next form, a menu of SIS processes.

Since the program executes under script control, character input is taken from the script file rather than the terminal keyboard. Script characters are supplied every half second to simulate human typing. Some delays have been included in the script to facilitate reading. The progress of the script can be controlled from the keyboard. Type 'S' to temporarily halt the script; type 'G' to resume the script. After halting the script by typing 'S', you can step through it one character at a time by typing <SPACE>. Since escape sequences are several characters long, it is necessary in some cases to step through more than one character before any action is taken by the Form Driver. The script can be aborted at any time by typing 'Z' (not control Z). After aborting the script, all subsequent input is taken from the keyboard rather than the file. Should you abort in the middle of an escape sequence, the Form Driver may signal an error (by sounding the bell) in response to the first few characters entered manually, but it should not persist.

3.2 UETP Step-by-Step Procedure

The following description of the execution of the script is divided so that each step corresponds to a single form displayed by the application. Within each step, the input from the script and the resulting actions are described.

STEP 1 - SIS INTRODUCTION

1. The script types the <HELP> key (the "PF2" key on the keypad) and a one line explanation of how to respond to the prompt NEXT? is displayed on the last line of the screen.
2. The script types <HELP> again, and FDV displays the HELP form that is associated with the introduction form.

STEP 2 - INTRODUCTION HELP FORM

The script types the <RETURN> key, and FDV displays the introduction form again. The cursor is at the NEXT? prompt.

STEP 3 - SIS INTRODUCTION

The script accepts the default response (4) to request a correction form. SIS processes the response, and the correction form is displayed.

STEP 4 - CORRECTION REQUEST

The script overrides the default response (N) and types D to request the form for changing inventory descriptions. SIS processes the response and displays the requested form.

STEP 5 - CHANGE DESCRIPTION

1. The script types a stock number, advances to the next input field ("Description") by typing the <TAB> key, and types the name of the item.
2. The script types in several new descriptions, advancing from field to field by using the <TAB> key, and types <RETURN> when the form is complete.
3. SIS processes the entries and displays the next form, a menu of SIS processes.

Since the program executes under script control, character input is taken from the script file rather than the terminal keyboard. Script characters are supplied every half second to simulate human typing. Some delays have been included in the script to facilitate reading. The progress of the script can be controlled from the keyboard. Type 'S' to temporarily halt the script; type 'G' to resume the script. After halting the script by typing 'S', you can step through it one character at a time by typing <SPACE>. Since escape sequences are several characters long, it is necessary in some cases to step through more than one character before any action is taken by the Form Driver. The script can be aborted at any time by typing 'Z' (not control Z). After aborting the script, all subsequent input is taken from the keyboard rather than the file. Should you abort in the middle of an escape sequence, the Form Driver may signal an error (by sounding the bell) in response to the first few characters entered manually, but it should not persist.

3.2 UETP Step-by-Step Procedure

The following description of the execution of the script is divided so that each step corresponds to a single form displayed by the application. Within each step, the input from the script and the resulting actions are described.

STEP 1 - SIS INTRODUCTION

1. The script types the <HELP> key (the "PF2" key on the keypad) and a one line explanation of how to respond to the prompt NEXT? is displayed on the last line of the screen.
2. The script types <HELP> again, and FDV displays the HELP form that is associated with the introduction form.

STEP 2 - INTRODUCTION HELP FORM

The script types the <RETURN> key, and FDV displays the introduction form again. The cursor is at the NEXT? prompt.

STEP 3 - SIS INTRODUCTION

The script accepts the default response (4) to request a correction form. SIS processes the response, and the correction form is displayed.

STEP 4 - CORRECTION REQUEST

The script overrides the default response (N) and types D to request the form for changing inventory descriptions. SIS processes the response and displays the requested form.

STEP 5 - CHANGE DESCRIPTION

1. The script types a stock number, advances to the next input field ("Description") by typing the <TAB> key, and types the name of the item.
2. The script types in several new descriptions, advancing from field to field by using the <TAB> key, and types <RETURN> when the form is complete.
3. SIS processes the entries and displays the next form, a menu of SIS processes.

Since the program executes under script control, character input is taken from the script file rather than the terminal keyboard. Script characters are supplied every half second to simulate human typing. Some delays have been included in the script to facilitate reading. The progress of the script can be controlled from the keyboard. Type 'S' to temporarily halt the script; type 'G' to resume the script. After halting the script by typing 'S', you can step through it one character at a time by typing <SPACE>. Since escape sequences are several characters long, it is necessary in some cases to step through more than one character before any action is taken by the Form Driver. The script can be aborted at any time by typing 'Z' (not control Z). After aborting the script, all subsequent input is taken from the keyboard rather than the file. Should you abort in the middle of an escape sequence, the Form Driver may signal an error (by sounding the bell) in response to the first few characters entered manually, but it should not persist.

3.2 UETP Step-by-Step Procedure

The following description of the execution of the script is divided so that each step corresponds to a single form displayed by the application. Within each step, the input from the script and the resulting actions are described.

STEP 1 - SIS INTRODUCTION

1. The script types the <HELP> key (the "PF2" key on the keypad) and a one line explanation of how to respond to the prompt NEXT? is displayed on the last line of the screen.
2. The script types <HELP> again, and FDV displays the HELP form that is associated with the introduction form.

STEP 2 - INTRODUCTION HELP FORM

The script types the <RETURN> key, and FDV displays the introduction form again. The cursor is at the NEXT? prompt.

STEP 3 - SIS INTRODUCTION

The script accepts the default response (4) to request a correction form. SIS processes the response, and the correction form is displayed.

STEP 4 - CORRECTION REQUEST

The script overrides the default response (N) and types D to request the form for changing inventory descriptions. SIS processes the response and displays the requested form.

STEP 5 - CHANGE DESCRIPTION

1. The script types a stock number, advances to the next input field ("Description") by typing the <TAB> key, and types the name of the item.
2. The script types in several new descriptions, advancing from field to field by using the <TAB> key, and types <RETURN> when the form is complete.
3. SIS processes the entries and displays the next form, a menu of SIS processes.

Since the program executes under script control, character input is taken from the script file rather than the terminal keyboard. Script characters are supplied every half second to simulate human typing. Some delays have been included in the script to facilitate reading. The progress of the script can be controlled from the keyboard. Type 'S' to temporarily halt the script; type 'G' to resume the script. After halting the script by typing 'S', you can step through it one character at a time by typing <SPACE>. Since escape sequences are several characters long, it is necessary in some cases to step through more than one character before any action is taken by the Form Driver. The script can be aborted at any time by typing 'Z' (not control Z). After aborting the script, all subsequent input is taken from the keyboard rather than the file. Should you abort in the middle of an escape sequence, the Form Driver may signal an error (by sounding the bell) in response to the first few characters entered manually, but it should not persist.

3.2 UETP Step-by-Step Procedure

The following description of the execution of the script is divided so that each step corresponds to a single form displayed by the application. Within each step, the input from the script and the resulting actions are described.

STEP 1 - SIS INTRODUCTION

1. The script types the <HELP> key (the "PF2" key on the keypad) and a one line explanation of how to respond to the prompt NEXT? is displayed on the last line of the screen.
2. The script types <HELP> again, and FDV displays the HELP form that is associated with the introduction form.

STEP 2 - INTRODUCTION HELP FORM

The script types the <RETURN> key, and FDV displays the introduction form again. The cursor is at the NEXT? prompt.

STEP 3 - SIS INTRODUCTION

The script accepts the default response (4) to request a correction form. SIS processes the response, and the correction form is displayed.

STEP 4 - CORRECTION REQUEST

The script overrides the default response (N) and types D to request the form for changing inventory descriptions. SIS processes the response and displays the requested form.

STEP 5 - CHANGE DESCRIPTION

1. The script types a stock number, advances to the next input field ("Description") by typing the <TAB> key, and types the name of the item.
2. The script types in several new descriptions, advancing from field to field by using the <TAB> key, and types <RETURN> when the form is complete.
3. SIS processes the entries and displays the next form, a menu of SIS processes.

Since the program executes under script control, character input is taken from the script file rather than the terminal keyboard. Script characters are supplied every half second to simulate human typing. Some delays have been included in the script to facilitate reading. The progress of the script can be controlled from the keyboard. Type 'S' to temporarily halt the script; type 'G' to resume the script. After halting the script by typing 'S', you can step through it one character at a time by typing <SPACE>. Since escape sequences are several characters long, it is necessary in some cases to step through more than one character before any action is taken by the Form Driver. The script can be aborted at any time by typing 'Z' (not control Z). After aborting the script, all subsequent input is taken from the keyboard rather than the file. Should you abort in the middle of an escape sequence, the Form Driver may signal an error (by sounding the bell) in response to the first few characters entered manually, but it should not persist.

3.2 UETP Step-by-Step Procedure

The following description of the execution of the script is divided so that each step corresponds to a single form displayed by the application. Within each step, the input from the script and the resulting actions are described.

STEP 1 - SIS INTRODUCTION

1. The script types the <HELP> key (the "PF2" key on the keypad) and a one line explanation of how to respond to the prompt NEXT? is displayed on the last line of the screen.
2. The script types <HELP> again, and FDV displays the HELP form that is associated with the introduction form.

STEP 2 - INTRODUCTION HELP FORM

The script types the <RETURN> key, and FDV displays the introduction form again. The cursor is at the NEXT? prompt.

STEP 3 - SIS INTRODUCTION

The script accepts the default response (4) to request a correction form. SIS processes the response, and the correction form is displayed.

STEP 4 - CORRECTION REQUEST

The script overrides the default response (N) and types D to request the form for changing inventory descriptions. SIS processes the response and displays the requested form.

STEP 5 - CHANGE DESCRIPTION

1. The script types a stock number, advances to the next input field ("Description") by typing the <TAB> key, and types the name of the item.
2. The script types in several new descriptions, advancing from field to field by using the <TAB> key, and types <RETURN> when the form is complete.
3. SIS processes the entries and displays the next form, a menu of SIS processes.

Since the program executes under script control, character input is taken from the script file rather than the terminal keyboard. Script characters are supplied every half second to simulate human typing. Some delays have been included in the script to facilitate reading. The progress of the script can be controlled from the keyboard. Type 'S' to temporarily halt the script; type 'G' to resume the script. After halting the script by typing 'S', you can step through it one character at a time by typing <SPACE>. Since escape sequences are several characters long, it is necessary in some cases to step through more than one character before any action is taken by the Form Driver. The script can be aborted at any time by typing 'Z' (not control Z). After aborting the script, all subsequent input is taken from the keyboard rather than the file. Should you abort in the middle of an escape sequence, the Form Driver may signal an error (by sounding the bell) in response to the first few characters entered manually, but it should not persist.

3.2 UETP Step-by-Step Procedure

The following description of the execution of the script is divided so that each step corresponds to a single form displayed by the application. Within each step, the input from the script and the resulting actions are described.

STEP 1 - SIS INTRODUCTION

1. The script types the <HELP> key (the "PF2" key on the keypad) and a one line explanation of how to respond to the prompt NEXT? is displayed on the last line of the screen.
2. The script types <HELP> again, and FDV displays the HELP form that is associated with the introduction form.

STEP 2 - INTRODUCTION HELP FORM

The script types the <RETURN> key, and FDV displays the introduction form again. The cursor is at the NEXT? prompt.

STEP 3 - SIS INTRODUCTION

The script accepts the default response (4) to request a correction form. SIS processes the response, and the correction form is displayed.

STEP 4 - CORRECTION REQUEST

The script overrides the default response (N) and types D to request the form for changing inventory descriptions. SIS processes the response and displays the requested form.

STEP 5 - CHANGE DESCRIPTION

1. The script types a stock number, advances to the next input field ("Description") by typing the <TAB> key, and types the name of the item.
2. The script types in several new descriptions, advancing from field to field by using the <TAB> key, and types <RETURN> when the form is complete.
3. SIS processes the entries and displays the next form, a menu of SIS processes.