PC-DRIVER OPERATOR'S MANUAL FOR ES-SERIES EMULATORS MODEL PCES

Applied Microsystems
CORPORATION

5020 148th Avenue N.E.
P.O. Box C-1002
Redmond, WA 98073-1002
(206) 882-2000 1-800-426-3925
TRT TELEX 185196/APMCRO UT

920-11205-01
SEPTEMBER, 1985

Copyright Applied Microsystems Corporation, 1985. All rights reserved.

IBM PC is a trademark of International Business Machines
Data General/One is a trademark of Data General
PC-Driver is a trademark of Applied Microsystems Corporation

Applied Microsystems Corporation has made every effort to document this product accurately and completely. However, Applied Microsystems assumes no liability for errors or for any damages that result from use of this manual or the equipment it accompanies. Applied Microsystems reserves the right to make changes to this manual without notice at any time.

PREFACE

PC-Driver is a family of products designed to support the development of microprocessor software on *IBM-PCs and PCs with *IBM compatibility. PC-Driver is available exclusively through Applied Microsystems Corporation and/or its sales and distribution channels.

This manual is your operator's guide for the use of the PCES version for use with Applied Microsystems' ES Series or 16-bit microprocessor emulators. PCES is designed to provide a user-friendly interface between the PC and the emulator.

We have organized the manual into four sections and an appendix. The first section is an introductory treatise dealing with the concept and scope of PCES. Those who are already familiar with the ES-Series emulators could begin with Section 2.

Sections 2 and 3 provide a guide to the installation and operation of the product.

Section 4 describes some of the advanced features and techniques available in PCES, and is recommended reading for advanced users after they have become familiar with the basic operation of the product.

The Appendix contains information which has been included for technical reference.

*Reference IBM as trademark of International Business Machines Corporation.

PC-DRIVER/PCES OPERATOR'S MANUAL

TABLE OF CONTENTS

SECTION 1. INTRODUCTION

1.1 SYSTEM CONCEPT 1-2

1.2 EMULATION AS A TOOL 1-3

SECTION 2. INSTALLATION

2.1 EQUIPMENT REQUIREMENTS 2-2
- 2.1.1 The Host Computer 2-2
- 2.1.2 Microprocessor Emulator 2-2
- 2.1.3 The PCES Symbolic Debugging Package 2-2

2.2 INTERCONNECTING THE EQUIPMENT 2-3
- 2.2.1 Safety Precautions 2-3
- 2.2.2 Setting Up The Emulator 2-3
- 2.2.3 Connecting The Emulator To The Target System 2-4
- 2.2.4 Connecting The Emulator To The PC 2-4

2.3 INSTALLING THE PROGRAM 2-6
- 2.3.1 Creating A Working Diskette 2-7
- 2.3.2 Installing PCES On A Hard Disk 2-7
- 2.3.3 Setting Path 2-8

2.4 STARTING THE PROGRAM 2-9

2.5 CONFIGURING PCES 2-10

2.6 INITIALIZING THE ES EMULATOR 2-12

2.7 USING THE EMULATOR 2-14

SECTION 3. OPERATION

3.1	INTRODUCTION	3-2
3.2	THE MAIN MENU	3-3
3.3	GETTING ON-LINE HELP	3-5
3.4	EMULATION	3-6
	3.4.1 The ES Initialization Procedure	3-6
	3.4.2 Normal Emulation Mode	3-8
3.5	CONFIGURATION	3-10
	3.5.1 Configuration Mode Selection	3-10
	3.5.2 Setting the Parameters	3-11
	3.5.3 Setting the Serial Port	3-11
	3.5.4 PC Baud Rate	3-12
	3.5.5 Host Baud Rate/ Terminal Baud Rate	3-12
	3.5.6 ES User Manual	3-13
3.6	CHANGE DIRECTORY	3-14
	3.6.1 The Drive Specifier	3-14
	3.6.2 The Path String	3-15
	3.6.3 Optional Filespec	3-15
3.7	UPLOAD (F4)	3-17
3.8	DOWNLOAD (F5)	3-19
3.9	EXIT TO DOS	3-20

SECTION 4. ADVANCED FEATURES

4.1 THE MACRO FEATURE 4-2
4.1.1 Loading A Macro Buffer 4-2
4.1.2 Executing A Macro 4-2
4.1.3 Notes On Using Macros 4-3
4.1.4 Clearing Macros 4-4

4.2 USING PCES IN MULTIPLE DIRECTORIES 4-5

4.3 EXECUTING DOS FUNCTIONS WITHIN PC-DRIVER 4-6

APPENDIX A. THE PROGRAM FILES AND MACHINE COMPATIBILITY

A.1 THE PROGRAM FILE PCES.EXE A-2
A.2 THE HELP FILE PCES.HLP A-3
A.3 CONFIGURATION FILE PCES.DAT A-3

APPENDIX B. SERIAL INTERFACE

B.1 SERIAL INTERFACE FOR THE PC-XT B-2
B.2 SERIAL INTERFACE FOR THE PC-AT B-3

APPENDIX C. INITIALIZATION OF THE ES EMULATOR

C.1 INTRODUCTION C-2
C.2 CONFIGURATION MODE 1 - ES INITIALIZATION C-2
C.3 CONFIGURATION MODE 2 - ES INITIALIZATION C-2
C.4 CONFIGURATION MODE 3 - ES INITIALIZATION C-3

LIST OF FIGURES

2-1	RS-232 Cable Connections From PC To Emulator	2-5
3-1	Main Menu Screen	3-3
B-1	Serial Interface For The PC-XT	B-2
B-2	Serial Interface For The PC-AT	B-3

SECTION 1. INTRODUCTION

1.1 SYSTEM CONCEPT
1.2 EMULATION AS A TOOL

1.1 SYSTEM CONCEPT

The PC-Driver is a family of design and debugging tools for software and hardware design professionals who need to add microprocessor emulation capability to a PC-compatible MS-DOS computer. PCES is just one product in the family providing the interface between the PC and Applied Microsystems' family of ES-series (16-bit) emulators.

The program provides a convenient scrollable display of the emulation session, as well as providing configuration and directory control, and file access.

Since the ES-series emulators can accept binary files in a variety of formats, programs may be developed on the PC using many popular assemblers, compilers and linkers, then downloaded to the target system (or emulator RAM overlay) where they can be debugged and tested.

During emulation, all commands are passed directly to the ES, where they are processed by ESL (ES-Series command language). The full power of the ES is available for symbolic debugging and setting complex breakevents. Both PCES and ESL have macro capability, so commonly used commands may be shortened to as few as one or two keystrokes.

All of the PCES programs provide menu screens for the main functions. All PCES programs also have "on-line" help. For PCES there are over 30 pages of help contained in the help file. Access to help for a specific situation is available by pressing the '?' key.

The menus are selected by using the function keys on the PC. The functions provided by the keys are displayed at the bottom of the screen at all times.

1.2 EMULATION AS A TOOL

The tools and equipment for developing microprocessor-based products have been steadily evolving since Applied Microsystems introduced its first transparent real-time emulator in 1978. Traditional electronic equipment was not adequate for microprocessor system development. The time-dependent nature of computer architecture, coupled with the fact that many circuits and signals are integrated in silicon and inaccessible, created the need for a revolutionary approach.

Emulation offers many benefits to system development. Without it, new product development can be a long and time-consuming process, especially in the integration phase, where hardware and software are integrated into the final product.

Emulation, in the context of microprocessors, is the concept of replacing the microprocessor itself with a device that provides the pin-equivalent function, while at the same time it allows the user control and visibility over the functions.

Emulators do more than simply provide a microprocessor with a front panel. Applied Microsystems Corporation emulators have Random Access Memory (RAM) which can be used in place of allocated memory space in the target system during development of the software. This RAM can be configured to act like ROM so that ROM code can be checked and modified before programming the actual chips.

Other features include the ability of the emulator to trace the runtime events of the target program, displaying the executed code in a convenient mnemonic format.

Emulators are useful in production too. The microprocessor socket is an excellent interface to most systems since it usually has access to most of the hardware. Emulators can be used to checksum ROM, test RAM memory, exercise I/O ports, etc., all without the overhead of including the test program in the product.

Tests can be automated or run individually. They can be run from RAM overlay or from macros stored in PCES. When hardware problems are discovered, an emulator and an oscilloscope are quite often the best pair for troubleshooting them, since the emulator can stimulate the problem and the oscilloscope (or logic analyzer) can be used to analyze it.

SECTION 2. GETTING STARTED

2.1 EQUIPMENT REQUIREMENTS
- 2.1.1 The Host Computer
- 2.1.2 Microprocessor Emulator
- 2.1.3 The PCES Symbolic Debugging Package

2.2 INTERCONNECTING THE EQUIPMENT
- 2.2.1 Safety Precautions
- 2.2.2 Setting Up the Emulator
- 2.2.3 Connecting the Emulator to the Target System
- 2.2.4 Connecting the Emulator to the PC

2.3 INSTALLING THE PROGRAM
- 2.3.1 Creating a Working Diskette
- 2.3.2 Installing PC-Driver on a Hard Disk
- 2.3.3 Setting PATH

2.4 STARTING THE PROGRAM

2.5 CONFIGURING PCES

2.6 INITIALIZING THE ES EMULATOR

2.7 USING THE EMULATOR

2.1 EQUIPMENT REQUIREMENTS

This PCES symbolic debugging and emulation controller package requires the following system components to run:

2.1.1 The Host Computer

To run PCES requires an IBM-PC, PC-XT or PC-AT or compatible clone with the following:

- 128Kb RAM
- Serial port configured as COM1 or COM2
- 5 1/4" double-sided, double-density floppy disk drive
- MS-DOS or PC-DOS version 2.00, 2.10 or 3.00 (or later)

2.1.2 Microprocessor Emulator

This package is designed to work with the Applied Microsystems' ES-Series 16-bit microprocessor emulators.

2.1.3 The PCES Symbolic Debugging Package:

AMC Part Number	Description
• 345-10305	Floppy disk containing the control programs
• 920-11205	Installation and Operation Guide
• 600-10486	Interconnecting Cable for PC or PC-XT

2.2 INTERCONNECTING THE EQUIPMENT

To run PCES you must first connect your ES-series emulator to the PC via an RS-232 cable and connect your target system to the emulator via the pod cable. The following procedure is a step-by-step procedure describing the setup.

2.2.1 Safety Precautions

The emulator contains a 3-wire cord with a 3-terminal polarized plug for connection to the power source and protective ground. The grounding terminal is connected to the metal chassis parts of the instrument. The emulator provides electrical shock protection only if the plug is connected to the mating outlet with a protective ground contact that is properly grounded.

-NOTE-

> Grounding conflicts may arise when the emulator is connected between two pieces of equipment with differing ground potentials. Use caution when connecting between target equipment and the PC.

2.2.2 Setting Up The Emulator

PCES can be configured to run with any one of these ES-series emulators:

MODEL	PROCESSOR #
ES -	8086/8088
ES -	80186/80188
ES -	68000/10/08
ES -	Z8001/2/3

2.2.3 Connecting The Emulator To The Target System

1. Be sure that the emulator is compatible with the microprocessor being emulated. Review the manual that came with the emulator you are using if you are not sure.

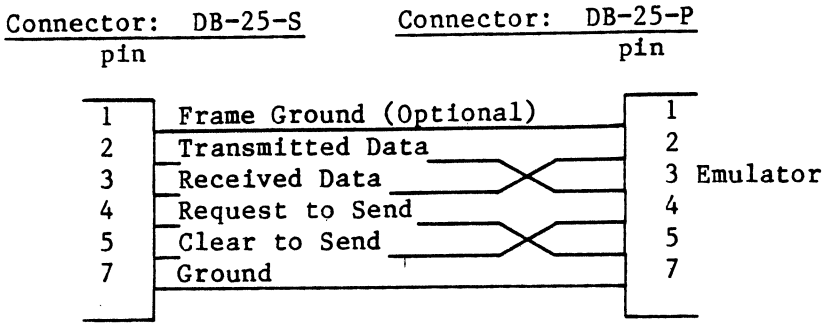
2. With the power off for both the emulator and the target system, remove the microprocessor from its socket noting the location of Pin 1. Replace it with the emulator's probe tip. Make sure the bevel or Pin 1 indication on the emulator probe tip is aligned with Pin 1 on the microprocessor socket.

CAUTION

The pins that make up the probe tip on the emulator are fragile. If they are banged against other objects, they may fracture and break. If they are broken, chances are the emulator will not function properly. The probe tip and/or pins must be replaced if pins are broken or damaged.

2.2.4 Connecting the Emulator To The PC

The emulator is connected to the PC via a RS-232 cable. The pin connections for the cable for most PC host machines are shown in Figure 1.1. For pin-out for other machines, such as the PC-AT, and additional information, refer to Appendix B.



Note that pins 2 and 3 are swapped, as are 4 and 5.

Figure 2.1 RS-232 Cable Connections From PC To Emulator

NOTE

If you are using a standard IBM-PC (not the XT or AT hard disk version) a serial port must be added in the chassis. This port will have a male connector. The printer port is a female.

Configure the PC Serial Port hardware for either COM1 or COM2. Port COM1 is assumed for the rest of this manual. If the PC has selection for RS-232 or current-loop, make sure that RS-232 voltage levels are selected. Usually, the PCs are shipped with COM1, RS-232 selected. Appendix B contains diagrams for configuring IBM serial ports.

2.3 INSTALLING THE PROGRAM

PCES is shipped on floppy disk formatted for 360Kb double-sided, double-density soft-sectored disk drives. There are several files on the disk:

• README	Revision information and notes on installation
• PCES.EXE	Main PC-Driver Program
• PCES.HLP	On-line help file
• PCES.DAT	Configuration file
• INSTALL.BAT	Hard-disk installation batch file

The files whose name begins PCES are the only ones necessary to run the program.

If you are using a PC that does not have a hard disk, you will want to create a working diskette containing PCES. The original diskette should be stored away in a safe place as a backup. Follow the steps in Section 2.3.1 only.

If you are using a PC that has a hard disk, we recommend that PCES be permanently installed on it. The file "INSTALL.BAT" has been provided to simplify the installation. Follow the steps in Section 2.3.2 and 2.3.3 to install PCES.

2.3.1 Creating A Working Diskette

If your PC has only one floppy disk drive, use the /s option with the DISKCOPY utility to copy the PCES disk. This will load the DOS files onto the copy diskette along with the PCES files. You may then use this single disk to boot the system and run PCES.

If your PC has dual floppy disk drives then you may use either the COPY command or DISKCOPY utility. See your DOS reference manual for instructions on running COPY or DISKCOPY.

Proceed with Section 2.3.4.

2.3.2 Installing PC-Driver On A Hard Disk

While PCES may be installed in any directory on your hard disk, we recommend that you reserve a directory by itself named "AMCTOOLS" for PCES and any other utility you may obtain from Applied Microsystems. The batch file INSTALL.BAT will create the directory /amctools in the root directory of disk specified.

To install PCES, insert the PC-Driver diskette into drive A. Type "INSTALL n:" where 'n' is the drive specifier for the hard disk (usually c). PCES will be installed on the hard disk in directory /AMCTOOLS.

2.3.3 Setting PATH

You will most likely want to execute PCES from any directory. In this way you can have a "working" directory where you maintain copies of your work, and several other directories, each containing programs that you will use to do the work. For example, you may be working on a program MIFILE.ASM in directory /WORK, have an assembler stored in directory /ASMB, and editor in directory /UTIL, and PCES stored in directory /AMCTOOLS.

In order for DOS to find these programs, the designers of MS-DOS have included a concept called PATH. PATH is used in DOS versions 2.00 and later.

To read the current PATH, type "PATH" at the DOS prompt. DOS will return with a string specifying the current path. If no path has been set, DOS will display "No Path".

To set a new PATH for the example above, type "path=\asmb;\util;\amctools" at the DOS prompt. Once set, this path string becomes part of the DOS environment and will instruct DOS to first look in the current directory, then in directory\ASMB, the \UTIL and finally \AMCTOOLS for a program when the program name is typed at the prompt.

See DOS reference for more information about PATH.

2.4 STARTING THE PROGRAM

You are now ready to run PCES with the power on and the DOS prompt on the PC, type PCES. PCES will take a few seconds to load, then the header page should appear on the screen.

The following things should be noted. The program name and version number should appear on the upper left-hand corner of the screen. The Serial Number should be displayed in the upper right-hand corner of the screen. This is the only time that the Version and Serial Number are displayed. There should be a prompt near the top of the screen stating:

Change Config? (Y/N)

If this is the first time you have run PCES, it may need to be configured to your application, and you should answer 'Y' to the prompt by typing 'Y'. This will bring up the configuration menu.

2.5 CONFIGURING PCES

This section is intended to guide first-time users who are anxious to begin using PCES. For a more complete description of the configuration menu, see the Operations section in this manual.

If you have just completed the above startup procedure, the configuration menu should be on the screen. If it is not, press the 'F10' key followed by the 'F2' function key.

The configuration menu should now be displayed. There are 3 modes available for PCES. Each of these modes has a slightly different configuration menu. The modes are selected by pressing the numeric keys 1 through 3.

For each mode selection, PCES should display a block diagram depicting the configuration of the development hardware for different applications. In the block diagram the PC is shown on the left. The "host" is defined as the device providing the files containing the target program, which could be uploaded or downloaded to the emulator. The "terminal" is defined as an ASCII terminal which could optionally be used in the set up.

For first time users, we recommend you configure PCES for Mode 3 by pressing '3' then <Enter>.

You now need to set the Communications Port, the Baud Rate and the User Number for this mode. Most of the selection can be done with the space bar and enter keys. The cursor keys with the up/down, left/right cursor keys. If they do not seem to work, the problem is probably the Num-Lock key. On some keyboards this key toggles the cursor keypad to a numeric keypad and back again, and it may be in the numeric mode. If you get stuck, help is available for each prompt by typing '?', or look in the Operations section of this manual for more detail information.

Set the Communications Port to COM1: (or COM2:), the Baud rate to 9600, and the ES Users number to 0. Press <Enter> after each selection is made to go to the next line. After all the parameters have been set, PCES will prompt for "SAVE TO DISK AS DEFAULT SETUP". Answer 'Y' to the "Save to Disk" prompt, and the program should return to the Main Menu.

You are now ready to begin an emulation session.

2.6 INITIALIZING THE ES EMULATOR

At this point in the procedure, the PC should be running PCES and the target system and the Emulator should be off. The emulator should be connected to the target system.

Press F1 to display the Emulation menu. PCES should display the message:

"DOES THE ES NEED INITIALIZATION (Y/N)?"

Answering 'Y' to this prompt will command PC-Driver to go through a setup and initialization procedure. This procedure will send commands to the ES to set up its baud rates and screen parameters, and should be done whenever the PCES configuration is changed.

The intialization procedure will instruct you when to turn power on to the emulator. If using the emulator with a target system, you would usually turn the target system power on prior to running the procedure; however, it is not absolutely mandatory and could be turned on after the emulator is initialized.

The initialization procedure may ask you to change the setting of the parameter switch on the ES. This switch is a thumbwheel switch, and is mounted behind the front panel on the ES emulator. Access to this switch is by loosening the thumb screws on either side of the front panel, and tipping it forward slightly. The position of the thumbwheel switch is read by ESL when the emulator is powered up. Therefore, for the procedure to work correctly, the thumbwheel must be set prior to power up of the emulator.

The procedure may also ask you to interchange the serial cables on the back of the ES emulator. This is necessary for correct initialization of baud rates, etc.

Answer 'Y' to the prompt and follow the procedure.

If the intialization procedure encounters an error condition, it will display an error message and go back to the prompt again. If this happens, correct the error and re-run the procedure. Once complete, the program will no longer prompt for initialization until the program is restarted or the configuration changed.

NOTE:

> If for some reason you feel that the procedure was not carried out correctly and you wish to repeat it, you can "fool" the software into prompting for initialization again by selecting the configuration menu, and entering the mode number again. (Example, press 'F2', '3', 'F1' if you are using mode 3).

At the end of the setup procedure you may be asked whether you wish to save the parameters in EEPROM. If you answer 'Y' to the prompt, you will not have to repeat the setup procedure next time the emulator is powered up. The emulator will configure itself to the configuration specified for the current position of the parameter switch. Consult the ES manual for more information on the Parameter Switch and the LD and SAV commands.

A complete description of the ES initialization procedure is given in Appendix C of this manual.

2.7 USING THE EMULATOR

You should now have an ES prompt on the screen '>'. Commands typed on the keyboard are sent to ESL where they are processed. Responses from the emulator are displayed on the screen and also in the screen buffer. The cursor keys and the PgUp/PgDn keys can be used to review the buffer. The 'Home' key will display the oldest page from the buffer, the 'End' key will display the latest page.

IMPORTANT

> If the system cursor is not displayed on the screen, then the buffer is not positioned on the last page. The cursor must be visible for commands to be sent to the emulator. Press 'End' to restore screen to the last page of the buffer.

PCES is not a replacement for ESL (ES emulator command language). You will need to be familiar with ESL in order to efficiently operate the emulator. While PCES provides some features that enhance the usefulness and flexibility of ESL, you will need to master the command language in order to get the most satisfaction with the product. See the ES manual that came with your emulator for operational information.

SECTION 3. OPERATION

3.1 INTRODUCTION

3.2 THE MAIN MENU

3.3 GETTING ON-LINE HELP

3.4 EMULATION
- 3.4.1 Normal Emulation Mode

3.5 CONFIGURATION
- 3.5.1 Configuration Mode Selection
- 3.5.2 Setting the Parameters
- 3.5.3 Setting the Serial Port
- 3.5.4 PC Baud Rate
- 3.5.5 Host Baud Rate/Terminal Baud Rate
- 3.5.6 ES User Manual

3.6 CHANGE DIRECTORY
- 3.6.1 The Drive Specifier
- 3.6.2 The Path String
- 3.6.3 Optional Filespec

3.7 UPLOAD (F4)

3.8 DOWNLOAD (F5)

3.9 EXIT TO DOS

3.1 INTRODUCTION

This section details the basic features available in PCES. PCES has five "screens" or menus, plus a main menu. These five "sub-menus" are often referred to as operation modes. (Not to be confused with configuration modes). The names of the five modes are displayed as selections in the main menu plus an Exit-To-DOS.

Each mode has its own display "screen" which is organized to allow sufficient interface for that particular function. The names of the modes and their corresponding functions are listed below.

1. **Emulation.** Provides access to the ES emulator. It contains a large screen buffer which maintains a transcript of the emulation session. The buffer may be viewed by scrolling.

2. **Configuration.** Contains parameters for setting up PCES and the ES emulator for a specific application.

3. **Change Directory.** Allows the current directory path or drive specifier to be changed. Displays the current directory.

4. **Upload.** Displays the current directory. Allows opening of a file for writing.

5. **Download.** Displays the current directory. Allows a filename to be entered, and then opened for reading.

6. **Exit to DOS.** Displays the Main Menu. Prompts for Exit to DOS, which, if confirmed, terminates PCES and returns to DOS.

The following sections describe the keystrokes and their actions for each of the operation modes as well as for the Main Menu.

3.2 THE MAIN MENU

The following is a replica of the Main Menu display. There are several ways to select an operation mode:

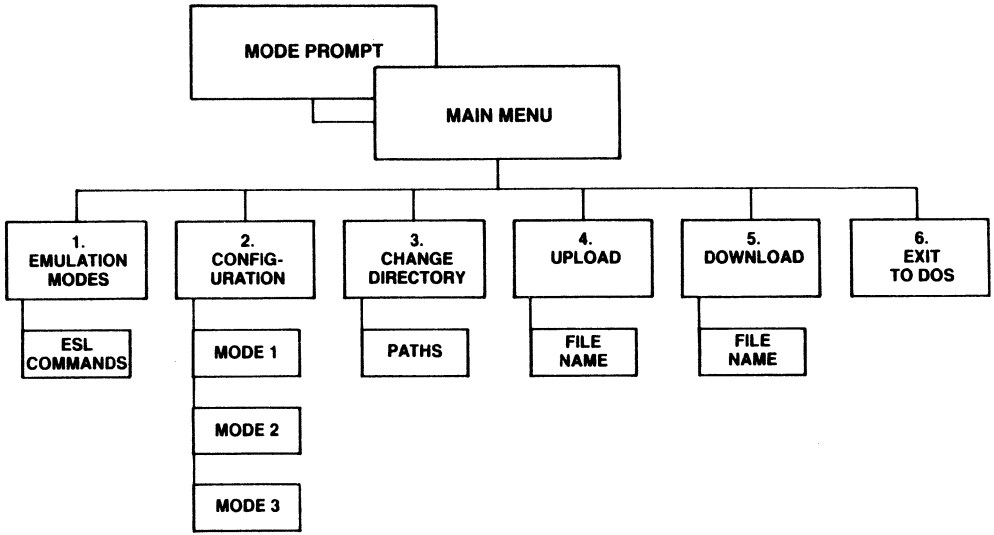
```
-------------------------------------------------
     Applied Microsystems Corp.  Type ? for Help

     1.  Emulation            4.  Upload
     2.  Configuration        5.  Download
     3.  Change Directory     6.  Exit to DOS

Enter Selection __

-------------------------------------------------
```

Figure 3.1. Main Menu Screen

When the Main Menu is displayed, the software highlights the name of the currently selected mode. The cursor keys may be used to move the selection to a different one. The numeric keys 1 through 6 can also be used to make a selection. With any of the above methods of selection and display the sub-menu.

To make selections faster, the function keys F1 through F6 are programmed to select and display the new mode screen immediately, simplifying keystrokes.

The main menu need not be accessed to select a new menu. The Function keys F1 through F6 will always select and display the new menu, regardless of the menu currently being displayed.


MODE PROMPT
MAIN MENU
1. EMULATION MODES
2. CONFIG-URATION
3. CHANGE DIRECTORY
4. UPLOAD
5. DOWNLOAD
6. EXIT TO DOS
ESL COMMANDS
MODE 1
MODE 2
MODE 3
PATHS
FILE NAME
FILE NAME

3.3 GETTING ON-LINE HELP

The PCES program includes an on-line help feature. Help is available when you need assistance with an operation mode or function and can be accessed almost anywhere in the program by pressing the '?' key.

When help is requested, a window is opened on the screen and a help message is displayed. All of the help messages are displayed from the file PCES.HLP which is included in the package. Sometimes there are several pages of help available. Each subsequent page may be viewed by pressing the space-bar. Any other key is taken as input and processed by PCES. When the help feature exits, the screen is restored.

Help is also available for error messages. There are two sources of errors. PCES may generate error messages and place them in the upper right corner of the screen. A single '?' will cause an explanation of the error to be displayed.

ESL may generate its own error messages in response to a command or target system conditions. Entering two '?'s will cause PCES to ask ESL for a detailed error message. (The first '?' displays a normal PCES help page - ignore this.)

PCES looks for the help file in the same directory that PCES.EXE was found in. Care should be taken to insure that PCES.EXE, and PCES.HLP are all kept in the same directory. If PCES cannot find the help file, an error message is displayed.

3.4 EMULATION (F1)

This operation mode will be used most of the time when using PCES. It is within this mode that most of the commands to the emulator will be given.

The screen is implemented as a full-duplex ASCII terminal. Commands typed by the user are sent to the emulator, and data received from the emulator is displayed.

The screen is actually displayed from a buffer. The buffer can contain several pages of data, which is saved as characters are received from the emulator and placed to the screen on a character by character basis. By maintaining a screen buffer, it is possible to scroll back through the emulation session to review past events.

When entering this mode, the software checks to see if the configuration has been changed since last time. If so, a special prompt is displayed asking the operator if the ES needs initialization. If the answer is yes, an initialization procedure is executed.

3.4.1 The ES Initialization Procedure.

PCES has a special routine whose purpose is to initialize some of the ES parameters to those selected in the configuration menu. Included in these are the baud rate and serial interface parameters for the terminal and computer ports of the ES emulator.

The whole purpose of the procedure is to make initialization simple and easy for the user. The user need only to follow the instructions given by the procedure as it executes. During the procedure, the program makes reference to the Parameter Switch. The Parameter Switch is a thumbwheel switch located behind the front panel on the ES series emulator, and accessed by loosening the two thumb screws on either side of the front panel, then tipping it forward slightly.

A description of what is actually accomplished by the procedure is included in Appendix C in this manual for reference.

After all of the parameters are set, the program issues a reset to the emulator (Ctrl-Z or 1Ah). This should cause the emulator to perform a hardware reset.

NOTE

> If the ES emulator has been used in a previous application where the default reset character was changed for some reason, it will have to be set by hand to a $1A. The reset character value may be read by the command "SET". For most ES emulators, the command to retore the reset character to a Ctrl-Z would be "SET 2, $1A".

PCES uses Ctrl-S and Ctrl-Q for the X-on/X-off protocol. These parameters are set in the ES at the factory. If they have been changed, they will also have to be set by hand. On most ES emulators, the command to restore the parameters is "SET 3, $11, $13".

3.4.2 Normal Emulation Mode

After the initialization procedure is done (or if you have elected not to run the initialization procedure), PCES will display the emulator transcription buffer.

When entering this mode, the last page of the buffer is displayed on the screen. This will contain the last data received from the emulator. If the initialization procedure was just executed, the commands executed by the procedure will probably be in the buffer. The cursor is also placed on the screen at the position where the next character will appear.

You are now communicating directly with the ES emulator. ASCII keys typed on the keyboard are sent to the emulator, and characters transmitted by the emulator are displayed on the screen.

Requesting help by typing '?' key in this mode will solicit help from the help file included with PCES. If you wish to receive help from the ES emulator, you will have to press the '?' key a second time.

The up and down cursor control keys will cause the display to scroll up and down one line with relation to the buffer in memory. The PgUp and PgDn keys will scroll the screen 16 lines at a time.

The 'Home' key will cause the first page (oldest saved data) to be displayed. Likewise, the 'End' key will display the last (or current) page to be displayed.

IMPORTANT

> The cursor is displayed on the screen only when the last page is being displayed. If the cursor is not visible, characters typed are sent to the emulator, but echoed characters are not displayed before continuing with the session, and will prevent temporary lockup of the emulator.

It is a good idea to press the "End" key after you are done reviewing previous emulation history. This will insure that the current page is being displayed before continuing with the session, and will prevent temporary lockup of the emulator.

NOTE

> On some MS-DOS systems the cursor may remain on the screen even if the last page is not displayed. Some interrupt driven "background" tasks may have been installed that affect the cursor attributes set by PCES.

3.5 CONFIGURATION (F2)

The configuration menu is provided so that PCES may be conveniently tailored to a specific application. The configuration menu is accessed by pressing Function key F2.

There are three configuration "modes" supported by PCES. Examples of applications using the different modes is included in the Advanced Features section of this manual. Mode 3 is the most common configuration for this program. The following paragraphs describe how to set the modes and parameters in this menu.

3.5.1 Configuration Mode Selection

When the configuration menu is first selected, the user is prompted for a "mode" selection. One of three modes are supported. Pressing the numeric keys 1 through 3 will select the corresponding mode. The software is also programmed to respond to the left or right cursor keys to make the selection.

For each selection, the software will display in block diagram form the hardware interconnection diagram characteristic of that mode. Select the configuration that most resembles your setup:

- Mode 1 assumes that the PC is going to be used to communicate to the emulator, and that an external "host" machine (which could be another PC) is going to be supplying the files containing formatted images of the target program.

- Mode 2 assumes that a standard ASCII terminal is going to be used to control the emulator, and that the PC is intended as the "host" machine, supplying the files containing the formatted images of the target program.

- Mode 3 assumes that the PC is going to be used for both the "host" machine and the "terminal" in the application.

After the correct mode has been selected, press <Enter> to access the individual parameters for the mode.

3.5.2 Setting the Parameters

The parameters vary depending on which mode was selected above. However, regardless of the mode selection, the parameters are all selected in pretty much the same way. The following paragraphs describe the settings for all of the parameters. Program the settings for the parameters that apply for the mode you have selected.

3.5.3 Selecting the Serial Port

The Serial Port refers to the serial port on the PC that is used to communicate to the ES emulator. PCES supports either COM1 or COM2. If only one port is available in the PC, the selection must match the port physically installed. Appendix B in this manual contains information on how to configure the IBM serial interface physical hardware for either COM1 or COM2.

The software selection is made by using the left or right cursor keys until the desired selection is showing on the screen. As an alternative, the space-bar may also be used to make the selection. When the correct setting is displayed, press <Enter> to go on to the next parameter.

When communication is first attempted to the ES emulator, the chosen serial port will be initialized, and an interrupt-driven handler will be assigned to it. If the port does not exist, an error will be displayed.

3.5.4 PC Baud Rate

The Baud Rate selection specifies the speed of asynchronous serial communications channel in bits per second, and refers to both the send and receive data.

Use the space-bar or the cursor control keys to select a baud rate. We recommend using the highest baud rate possible for the best performance, the slower baud rates are provided for special cases where modems must be used. Press <Enter> when the selection is made.

3.5.5 Host Baud Rate / Terminal Baud Rate

Use the space-bar or the cursor control keys to select the baud rate that matches the host or terminal that you are planning to use. Press <Enter> when the selection is made.

3.5.6 ES User Number

This selection defines the User Number for which the setup is to be saved in the ES. The User Number is selected using the space-bar, just as was done for previous parameters.

The ES emulator will save a default setup for 2 users. After the initialization program is run, the setup may be saved in EEProm within the ES. If this is done, then the next time the ES is used, you can skip the initialization routine and instead simply power up the ES with the Interface Parameter Switch set to the appropriate position.

PCES tells you which position to use for subsequent default power up. After selecting a user number, the Parameter Switch setting for default power up for this number and mode is shown in the block diagram near the bottom of the screen. Make a mental note of the setting if you plan to use this feature.

The ES reports the setting of the parameter switch when it performs its self test. This way you can check to see if the parameter switch is still set for your setup next time you use it without having to peek behind the front panel.

3.6 CHANGE DIRECTORY (F3)

PCES supports full directory path syntax. Often it is desirable to change to a different directory within the program. Pressing function key F3 will display the current directory and prompt for a new directory name or path. The command syntax is:

[d:][path][\filespec.tag]

(the brackets indicate optional parameters)

The directory specifier may contain a drive specifier, a path string, sub-directory name, and optionally a filespec.

3.6.1 The Drive Specifier

If included, the drive specifier must be the letter designation of a valid drive and must be followed by a colon ':'.

WARNING

The change drive is accomplished by using DOS system calls. If the drive specified is not ready, it will cause a critical error, which if aborted, could cause the program to be terminated by DOS. If the drive is a floppy, make sure there is a disk in the drive and that the door is closed.

3.6.2 The Path String

If specified, the path string is an MS-DOS directory path. If the first character in the path is a backslash, then the path is assumed to start from the root directory. If the string does not start with a backslash, then the path is assumed to start from the current directory. Special cases include the '..' directory, which is defined to be the parent of the current directory. All directory names must be separated by a backslash (\).

3.6.3 Optional Filespec

If a filespec is included in the string, then only files which match the filespec are displayed from the directory. This is useful when working with a large directory, and only files with a certain name or tag are of interest. A filespec must include a "dot" to be identified as a filespec. Wild cards are valid. The '*' wildcard character specifies that any character or string of characters are valid. The '?' character specifies that any character may occupy that position only in the filespec.

This convention essentially matches that defined for DOS.

NOTE

> Since the '?' character is used to get help in PCES, it must be pressed twice to enter it in the filespec. To avoid this inconvenience, the '+' character may be used instead. In this special case, the '+' character will show as a '?' character in the filespec.

EXAMPLES:

a:\ <Enter>
Changes the current directory to the root directory on drive 'a' and displays the directory.

*.exe <Enter>
Displays only filenames that have a tag (extension) of "EXE" from the current directory.

When PCES is terminated, the original directory and drive are restored.

3.7 UPLOAD (F4)

Pressing function key F4 is used to open a file for writing. The menu for this function asks you to enter a filename. After the file is open, the program will return to the Emulate mode. From this point on, characters received from the emulator are saved in the file. To close the file, press any function key.

The filename specified must conform to DOS filename syntax. The filenames are 1 to 8 characters in length and can be followed by a 1 to 3 character filename extension.

The following characters can be used for filenames:

A-Z 0-9 $ & # @ ! ' ` ( ) - { } _ ^ ~

Any other characters are invalid in the filename.

The filename may include a drive specifier and/or directory path, if desired. The same syntax that is used for Change-Directory is valid, except that wild card characters cannot be used in the filename. The entire string must be 12 characters or less, however.

If a file already exists with the same filename that you specify, you will be asked to select Append, Overwrite or Rename. Enter A, O or R (the key-strokes could be in lower case) to continue.

The Upload feature can be used in several ways. The most common use is to upload a formatted image of a target program from the emulator. The following procedure will produce a "clean" copy of the up-loaded program:

- In the Emulate mode, type the upload command to the emulator. Usually it will be of the form: "UPL 0 to 100" - Do not type the <Enter> key yet...

- Press 'F4' for Upload and type the filename where the data is to be stored.

- Press <Enter> to open the file. You will now be back in the Emulate mode, and there should be a line near the bottom of the screen stating "Writing to file xxx".

- Press <Enter> again. This will terminate the "UPL 0 to 100" command to the ES, and the emulator will begin uploading the data.

- As soon as the data transfer is complete, press 'F1'. This will close the file and you will still be in the Emulate mode.

The Upload feature can be used to save a transcript of the emulation session. If the file is opened early in the session, it will save a copy of everything transmitted from the emulator. This is useful for creating hard copies of trace memory, memory maps or breakevent setups.

3.8 DOWNLOAD (F5)

This function is similar to Upload, except that it opens a file for reading. It is used mostly for downloading a formatted binary target program from a disk file to the Emulator and/or Target system memory.

A line near the bottom of the screen will indicate that the file is being read. The download will be aborted when any function key is pressed.

The filename specified must conform to DOS filename syntax. The filenames are 1 to 8 characters in length and can be followed by a filename extension.

The following characters can be used for filenames:

A-Z 0-9 $ & # @ ! ´ ` ( ) { } _ ^ ~

Any other characters are invalid in the filename.

The filename may include a drive specifier and/or directory path, if desired. The same syntax that is used for Change-Directory is valid, except that wild card characters cannot be used in the filename. The entire string must be 12 characters or less, however.

To use this feature for downloading a target program, use the following steps:

- While in the Emulate mode, command the ES emulator to receive a file for downloading. Usually the command will be simply **"DNL <Enter>."**

- The ES is now waiting for the formatted data. Press ´F5´ to enter the Download mode on the PC and type the filename of the file to be sent.

- After the filename is entered, press **<Enter>."** This will open the file and begin sending it to the emulator.

- When the file has been sent, it will automatically be closed, and PCES will once again be in the Emulate mode.

During the time that the file is being transferred, PCES displays a count of the lines as they are sent. If an error is encountered, the download process is aborted.

The ES is compatible with several download formats. The format must be selected to be compatible with the file that is being sent. See the ES manual for information on setting upload/download formats.

The Download feature can also be used to send commands to the emulator. A commands file can be prepared using an editor and downloaded (at the ES prompt) to accomplish complex setups of breakevents and/or memory maps.

3.9 EXIT TO DOS

When you are done using PCES, and wish to return to DOS, press F10. This will bring you back to the Main Menu. Pressing F6 or F10 will then select the "Exit to DOS" function. Confirm that this is what you want to do by pressing **<Enter>.**

When the program terminates, the original current drive and directory is restored. If you have set up any macros, they are saved in file PCES.DAT. This way they will be reinstated the next time the program is run.

SECTION 4. ADVANCED FEATURES

4.1 THE MACRO FEATURE

4.1.1 Loading a Macro Buffer

4.1.2 Executing a Macro

4.1.3 Notes on Using Macros

4.1.4 Clearing Macros

4.2 USING PCES IN MULTIPLE DIRECTORIES

4.3 EXECUTING DOS FUNCTION WITHIN PCES

4.1 THE MACRO FEATURE

There are 10 macro buffers available in PCES. This feature allows the user to create up to ten macro functions if desired. All of these macro buffers are saved in file PCES.DAT when PCES terminates. This way they are still intact next time the program is loaded.

4.1.1 Loading a Macro Buffer

The macro strings are set up by holding down the Ctrl key and simultaneously pressing one of the Function keys (F1 through F10).

To load a macro for 'F1', press:

Ctrl-F1 (simultaneously)

From this point on, all keystrokes will be saved in a macro defined for the function key 'F1'. A maximum of 255 keystrokes may be saved in the macro.

To end the loading of a macro, the Ctrl-Function key must be pushed again. In the above example, pressing **Ctrl-F1** again should terminate the loading of the macro buffer. Macros may not contain calls to other macros. Only one macro may be loaded at a time.

4.1.2 Executing a Macro

Any time within PCES, a macro buffer can be opened, and the key sequence stored there used as input in place of the keyboard. To execute a macro sequence, press and hold the **Shift key**, then press the **Function key** corresponding to the macro.

4.1.3 Notes on Using Macros

There are some inherent difficulties with implementing a Macro processor of this type. First of all, the macro may be invoked anywhere within PCES. Suppose you defined macro 'F1' while you were in the download menu. You might have started by typing the filename, then downloading the file and closing the macro load.

If the same macro was executed when in the Emulate mode, it would have an entirely different and unwanted effect. For this reason, it is good practice to begin defining a macro with the first character being a function key. This will guarantee that the macro begins execution from the same place every time and all of the subsequent keycodes should match the original prompts.

Another area where macros do not work well is when used in time-dependent functions, such as with the emulator. Setting up a macro to cause run to Breakevent on the emulator, then within the same macro continue with other keystrokes, will usually cause unwanted results. When the macro was entered, the operator presumably waited until the emulator reached the breakpoint before continuing to enter keys. When later executed, the macro processor does not wait for the emulator operation to terminate, hence sends the next keytrokes immediately. In this example, emulation will probably be stopped prematurely.

Macros are quite useful for other things, however. For instance, several breakpoint setups can be saved and later loaded by using the macro feature. You simply open a macro for loading at the time you enter each breakpoint setup. Later, these may be recalled from the macro buffer. "Canned" target system tests can also be saved as a macro and executed when needed.

4.1.4 Clearing Macros

Clearing a macro can be accomplished by opening and closing the macro without placing any commands in the macro to execute. For example, to clear macro F1, you would open macro F1 by pressing **CTRL** and **F1** simultaneously. By opening the macro you have erased any existing information. Instead of supplying any commands, simply close the macro by pressing **CTRL** and **F1** together again. The macro is now cleared. These same steps can be repeated for any of the ten macros, F1 through F10.

For more information on macros see Section 4.1.

4.2 USING PCES IN MULTIPLE DIRECTORIES

PCES is designed so that it can be used for several projects on one machine. As explained before, PCES should be installed in its own directory (or \AMCTOOLS) and defined in the DOS environment (PATH).

If you create a directory for each project, it is possible to configure PCES individually for each project. The technique is to copy the file "PCES.DAT" into the directory where the work is to be done.

PCES.DAT is a small configuration file containing the configuration menu setup, as well as the macro buffers. When PCES is started, it will look first in the current directory for this file. If it is found, that copy is used for the session.

The PCES.DAT file used by PCES does not change even if you change the current directory during the session. If PCES.DAT cannot be found when PCES starts up, (either in the current directory or with PCES.EXE), it is created in the current directory with factory default settings.

NOTE

> If PCES is invoked using a command which includes a path under DOS 3.00, PCES may not know how to find its companion files PCES.HLP, PCES.DAT. Including a PATH to PCES within the DOS Environment is recommended.

4.3 EXECUTING DOS FUNCTIONS WITHIN PCES

PCES includes a DOS "Shell Escape" feature. This is an advanced feature and should be used after you have familiarized yourself with the first three sections of this manual.

Often it is useful to be able to run a DOS function or other utility without having to terminate PC-Driver. PCES provides a system call to COMMAND.COM. To invoke COMMAND.COM from within PCES, press and hold the **Alt** key and press **1**. The screen should clear and DOS should begin running.

If this does not occur, it may be for one of the following reasons:

- There is no path specified to COMMAND.COM in PATH
- COMMAND.COM path does not include a drive specifier, and is on another disk.
- There is not enough memory left to load COMMAND.COM.
- The version of DOS was sensed as 2.00 or earlier. This feature only works on DOS version 2.1 or later.

Usually, however, COMMAND.COM will begin executing, and you will get the DOS prompt. What has actually happened is that a new copy of DOS has been loaded into RAM in the PC. This copy inherits its own environment from the parent copy. PCES is still resident in memory, but is inactive. PCES has saved its screen and its status in memory.

Provided there is still unused memory in the PC, the user can run utilities, create files, look at the time-of-day, or load another copy of PCES.

CAUTION

> **Loading programs that remain resident (such as device drivers) could cause memory allocation errors due to the fact that they will probably be loaded above PCES. These errors may not surface until PCES is terminated and another process is attempted.**

To return to the inactive PCES, the user should type **EXIT.** This will terminate COMMAND.COM and processing will be returned to PCES. PCES will restore its screen and continue where it left off.

NOTE

> Failing to exit the new COMMAND.COM will tie up memory in the PC.

You should be aware of the side effects from using this feature. As an example, if the directory is changed or files added, they may not appear in the directory display when PCES is restarted. This is because PCES restores the screen from memory, and may not know that the directory has changed.

APPENDIX A. THE PROGRAM FILES AND MACHINE COMPATIBILITY

A.1 THE PROGRAM FILE PCEM.EXE

A.2 THE HELP FILE PCES.HLP

A.3 CONFIGURATION FILE PCES.DAT

A.1 THE PROGRAM FILE PCEM.EXE

PCES.EXE is a compiled relocatable load module which is loaded by DOS. Part of the initialization done by PCES.EXE is to load a configuration file PCES.DAT.

PCES.EXE needs to access the PC screen, keyboard, disks, and serial ports, in order to be usable. For the most part, DOS function calls and standard BIOS interrupts are used to preserve portability between most MS-DOS machines. However, in some cases this is not desirable for performance reasons. PCES has been on most of the PC clones with no problems. The following technical information is provided for those users who may experience problems using non-IBM versions of the PC.

The directory, and trace displays access the Screen memory directly and do not call BIOS. Certain save and restore utilities also access the Screen directly. On most MS-DOS machines the monochrome monitor uses segment address B800h. PCES senses which monitor is currently active by accessing the BIOS equipment flag directly, then uses the appropriate screen buffer.

PCES.EXE also uses an interrupt-driven serial interface. Since this is not supported by the system BIOS, it is contained in PCES.EXE. The IBM Asynchronous Communication Adapter or compatible device is assumed to be located at port address 3F8h through 3FEh for COM1:, and 2F8h thru 2FEh for COM2:. On the PC-AT, PCES, EXE assumes that the (standard) AT-Serial/Parallel adapter is installed. PCES uses IRQ4 for COM1: and IRQ3 for COM2:. Compatible hardware should also work with PCES, provided it has the same software interface and pinout.

The Data General-One™ battery-powered computer employs a serial port that is not compatible with PCES. Consult the factory if planning to run PCES on the Data General-One™.

A.2 THE HELP FILE PCES.HLP

When the operator requests help during the course of running PCES, this file is opened, and the help page(s) displayed for the particular situation.

PCES.HLP should also accompany PCES.EXE in the same directory.

A.3 CONFIGURATION FILE PCES.DAT

This file is loaded each time PCES.EXE is executed. It is updated when the user saves a configuration and also when PCES.EXE terminates normally. PCES.DAT is a binary file that contains the macros that have been defined within PCES. Also, it could contain breakpoint setup and configuration setup if saved by the operator.

If PCES.DAT is missing, or cannot be found by PCES, it is created in the current directory with factory default settings.

APPENDIX B. SERIAL INTERFACE

B.1 SERIAL INTERFACE FOR THE PC-XT

B.2 SERIAL INTERFACE FOR THE PC-AT

B.1 SERIAL INTERFACE FOR THE PC-XT

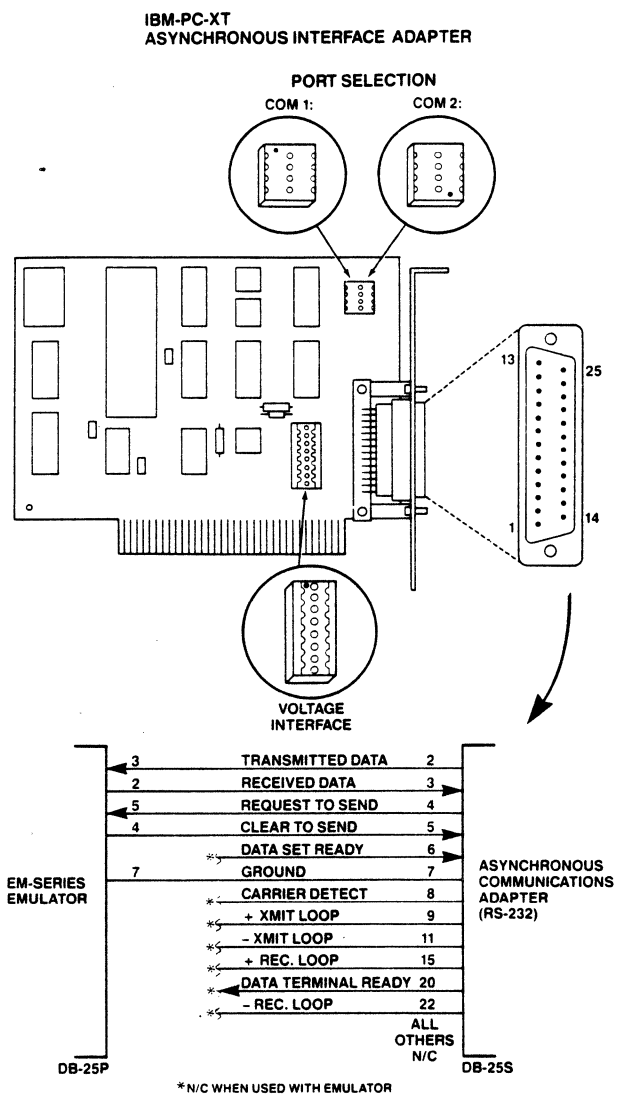


Figure B-1. Serial Interface For The PC-XT

B.2 SERIAL INTERFACE FOR THE PC-AT

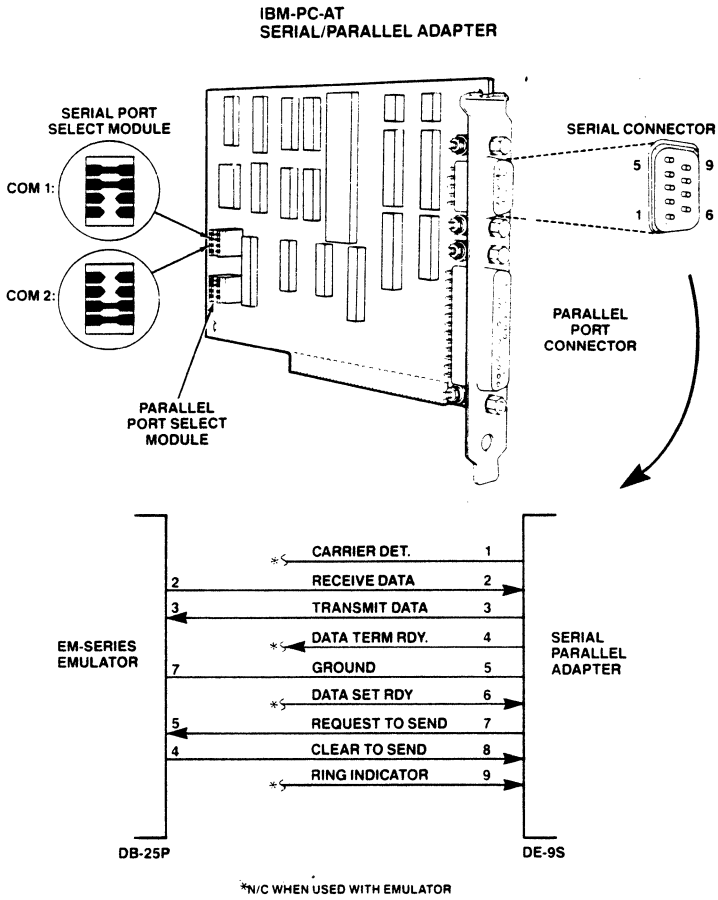


Figure B-2. Serial Interface For The PC-AT

APPENDIX C. INITIALIZATION OF THE ES EMULATOR

C.1 INTRODUCTION

C.2 CONFIGURATION MODE 1 - ES INITIALIZATION

C.3 CONFIGURATION MODE 2 - ES INITIALIZATION

C.4 CONFIGURATION MODE 4 - ES INITIALIZATION

C.1 INTRODUCTION

PCES sends a series of commands to the ES when the initialization program is executed. Executing this procedure is optional when first entering the Emulate mode.

The Commands sent vary depending on the configuration mode selected. The following is a list of the setup commands issued to the ES for each mode.

C.2 CONFIGURATION MODE 1 - ES INITIALIZATION

```
SET #10,#b      Set terminal port baud rate 'b'.
SET #11,1       Set terminal port to One Stop Bit.
SET #12,0       Set terminal port for No Parity.
SET #13,#23     Set for 23 lines per page.
SET #20,#b      Set computer port to baud rate 'b'.
SET #21,1       Set computer port to One Stop Bit.
SET #22,0       Set computer port for No Parity.
<Ctrl-Z>        Issue hardware rest command.
```

C.3 CONFIGURATION MODE 2 - ES INITIALIZATION

```
SET 1,#U        Set User Number (1 or 0).
SET #10,#b      Set terminal port baud rate 'b'.
SET #11,1       Set terminal port to One Stop Bit.
SET #12,0       Set terminal port for No Parity.
SET #13,#24     Set for 24 lines per page.
SET #20,#b      Set computer port to baud rate 'b'.
SET #21,1       Set computer port to One Stop Bit.
SET #22,0       Set computer port for No Parity.
SAV 0           Optional SAVE to EEPROM.
...Wait...      Wait for <Enter> from operator.
<Ctrl-Z>        Issue hardware reset command.
```

C.4 CONFIGURATION MODE 3 - ES INITIALIZATION

```
SET 1,#u        Set User Number (1 or 0).
SET #13,#23     Set for 23 lines per page.
SET #20,#b      Set computer port to baud rate 'b'.
SET #21,1       Set computer port to One Stop Bit.
SET #22,0       Set computer port for No Parity.
SAV 0           Optional SAVE to EEPROM
CCT             Transfer to Computer Port Control.
...Wait...      Wait for <Enter> from operator.
<Ctrl-Z>        Issue hardware reset command.
```