TeleVideo®
TPC I System Reference Manual

TELEVIDEO®
TPC I SYSTEM REFERENCE MANUAL

TeleVideo Part Number 125039-00 Rev. A
15 November 1983

TeleVideo® is a registered trademark of TeleVideo Systems, Inc.
Z80A® is a registered trademark of ZILOG Corporation.
CP/M® is a registered trademark of Digital Research, Inc.
SuperMouse,T.M. TeleWrite,T.M. TeleCalc,T.M. and TeleChartT.M. are trademarks of
TeleVideo Systems.

TeleVideo Systems, Inc., 1170 Morse Avenue, Sunnyvale, California
96086   408/745-7760

STATEMENT OF LIMITED WARRANTY

TeleVideo Systems, Inc. ("TeleVideo") warrants to its distributors, systems houses, OEMs, and national accounts ("Buyer") that products, except software, manufactured by TeleVideo are free from defects in material and workmanship. TeleVideo's obligations under this warranty are limited to repairing or replacing, at TeleVideo's option, the part or parts of the products which prove defective in material or workmanship within 180 days after shipment by TeleVideo. That Buyer may pass along to its initial customer ("Customer") a maximum of 90 days coverage within this 180-day warranty period, provided that Buyer gives TeleVideo prompt notice of any defect and satisfactory proof thereof.

Products may be returned by Buyer only after a Return Material Authorization number ("RMA") has been obtained from TeleVideo by telephone or in writing. Buyer must prepay all freight charges to return any products to the repair facility designated by TeleVideo and include the RMA number on the shipping container. TeleVideo will deliver replacements for defective products or parts on an exchange basis to Buyer, freight prepaid to the Buyer or the Customer. Products returned to TeleVideo under this warranty become the property of TeleVideo. With respect to any product or part thereof not manufactured by TeleVideo, only the warranty, if any, given by the manufacturer thereof, applies.

EXCLUSIONS

This limited warranty does not cover losses or damage which occur in shipment to or from Buyer or Customer, or is due to, (1) improper installation or maintenance, misuse, neglect or any cause other than ordinary commercial or industrial application, or (2) adjustment, repair, or modifications by other than by TeleVideo-authorized personnel, or (3) improper environment, excessive or inadequate heating or air conditioning and electrical power failures, surges or other irregularities or (4) any statement made about TeleVideo's products by salesmen, dealers, distributors or agents, unless confirmed in writing by a TeleVideo officer.

THE FOREGOING TELEVIDEO LIMITED WARRANTY IS IN LIEU OF ALL OTHER WARRANTIES, WHETHER ORAL, WRITTEN, EXPRESS, IMPLIED OR STATUTORY. IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE DO NOT APPLY. TELEVIDEO'S WARRANTY OBLIGATIONS AND DISTRIBUTOR'S REMEDIES HEREUNDER ARE SOLELY AND EXCLUSIVELY AS STATED HEREIN. TELEVIDEO MAKES NO WARRANTY WHATSOEVER CONCERNING ANY SOFTWARE PRODUCTS, WHICH ARE SOLD "AS IS" AND "WITH ALL FAULTS". TELEVIDEO'S LIABILITY, WHETHER BASED ON CONTRACT, TORT, WARRANTY, STRICT LIABILITY OR ANY OTHER THEORY, SHALL NOT EXCEED THE PRICE OF THE INDIVIDUAL UNIT WHOSE DEFECT OR DAMAGE IS THE BASIS OF THE CLAIM. IN NO EVENT SHALL TELEVIDEO BE LIABLE FOR ANY LOSS OF PROFITS, LOSS OF USE OF FACILITIES OR EQUIPMENT, OR OTHER INDIRECT, INCIDENTAL OR CONSEQUENTIAL DAMAGES.

TeleVideo Systems, Inc.

Table of Contents

Appendices

1. INTRODUCTION

The TeleVideo Portable Computer (TPC I) is an 8-bit computer with a 4 MHz Z80A CPU, providing a 64 Kbyte dynamic RAM, a 32 Kbyte alpha and graphic display memory, and an 8 Kbyte EPROM. CP/M is the operating system with the GSX extension from Digital Research for graphics.

The computer's processor is designed to allow both alphanumeric computing as well as standard business graphics. The alphanumeric display is 24 lines by 80 characters. The graphics display features a resolution of 640 x 240 pixels.

Figure 1-1
TPC I

The slim-line, 5 1/4-inch floppy disk drives are mounted vertically and side by side next to the screen providing 363.6 Kbytes of formatted disk storage per drive. You can order the TPC I with either one or two drives.

I/O ports consist of one RS-232C port for a modem, one parallel port for a printer, and one port for the TeleVideo SuperMouse, a cursor manipulation and input device. Available also as an option is the TPC I-N, an RS-422 card, to provide a port for connecting the TPC I to a TeleVideo network system.

The TPC I package includes, besides the hardware and the operating system, three applications programs. They are:

* TeleWrite for executive word processing
* TeleCalc for spreadsheet analysis
* TeleChart for business graphics

By adding the TeleVideo SuperMouse, many graphics and menu-driven applications programs become even easier to use. Contact your computer store or dealer for further information about the SuperMouse.

SOFTWARE

CP/M, developed by Digital Research, is the operating system for the TPC I. This system allows you to use CP/M-compatible applications program software.

The diskette that arrived with your system is labeled **CP/M Version x.x.** This diskette is referred to as the **system diskette** throughout this manual.

The TPC I can use any high-level programming language designed to run under CP/M. Some of the more popular languages that are used on the TPC I are BASIC, FORTRAN, COBOL, PASCAL, APL, ALGOL, PL/1, FORTH and "C".

Besides TeleCalc, TeleChart, and TeleWrite, there are many other applications programs commercially available for a wide range of tasks from payroll to security. In addition, you may write your own applications programs in any CP/M programming language.

NOTATION CONVENTIONS

Throughout this Manual and the User's Manual are a few symbols that have specific meanings. These symbols allow us to more easily communicate particular actions to you.

<Return>   The **<Return>** symbol indicates that you are to press the key marked with the squared, left arrow key on the right of the typewriter keys. It is called the **Return** key and is like the carriage return key on a typewriter.

< >   When a key, other than an alphabet letter or number key, is written, it is enclosed in brackets. For example, if you are to press the key marked Ctrl, it would be written like this, **<Ctrl>.**

<BAR>   In a command that requires spaces (such as
                          DIR B:<Return>), for clarity the spaces are
                          indicated by the spacebar symbol, **<BAR>**.
                          Pressing the <BAR> enters a space character
                          at the cursor. Our example command is
                          written like this:

                          DIR<BAR>B:<Return>

                          This means to type **DIR**, press the space bar,
                          type **B:**, and press the <Return> key.

_   To indicate the cursor, which is a blinking
                          block on your screen, you will see the
                          cursor symbol (_).

/   For some commands, you must press two keys
                          simultaneously. We show this action with the
                          slash (/) symbol. For example, you may be
                          asked to press the <Ctrl> and the C keys at
                          the same time. The keys will be written with
                          a slash dividing them. Using our example,
                          when you are to press the <Ctrl> and C keys
                          at the same time, you will see it as
                          **<Ctrl>/C. Do not press the slash key when
                          you see it in a command.** Press the slash key
                          ONLY when it is enclosed in brackets like
                          this: </>.

Special Notes

Two types of notes call attention to information of special
importance.

NOTE!   This is a general note emphasizing important
               information.

STOP!   This indicates special information concerning possible
               loss of data or damage to the system. **When you see
               this, STOP and read the note before proceeding.**

Step-by-step procedures are used in this manual to show how to
perform various functions. These procedures show what you are to
do and how your computer responds to that action. The following
is an example of the procedure format:

       USER: 1.   **Your Action**

       SYSTEM: 2.   Screen Display or Action

Read the entire procedure before beginning the operation and
completely read the step you are working on before entering a
response. Many steps offer a choice. All of your actions are
shown in **bold print**.

2. SETUP

The step-by-step instructions for setting up the TPC I are in the User's Manual. This chapter gives information pertaining to the more technical aspects of the installation such as power sources, voltage regulators, switch settings, and peripherals and baud rates.

INSTALLING YOUR SYSTEM

For optimum performance, locate the system at least five feet from other computing equipment, electrical appliances, or equipment such as elevators, radio transmitters, and television sets, that generate magnetic fields.

Power Cord Connection

The power cord that came with your system plugs into the back of the main unit as shown in Figure 2-1. The source power supply and proper connections are described here.

Figure 2-1
Power Cord Connection
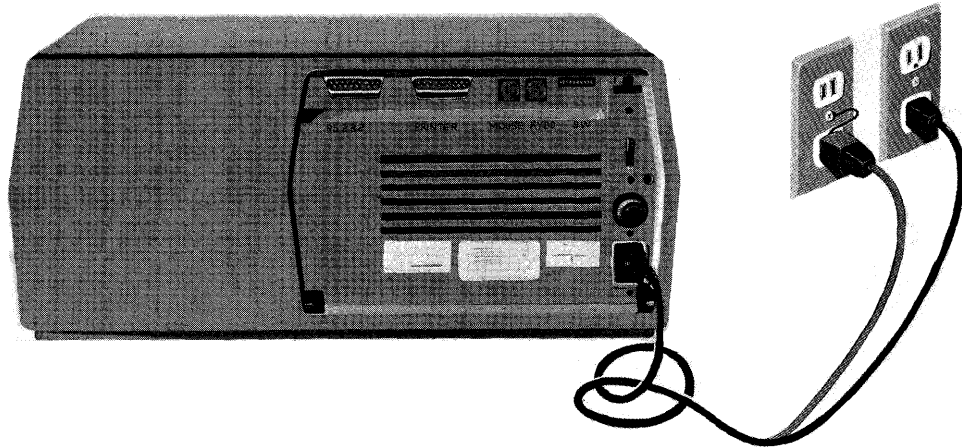
The TPC I requires a three-prong electrical outlet. The power cord has a three-prong plug. If you use it with a two-prong adapter, ground it with a **pigtail.** See Figure 2-1.

Internally, the power cord wires are color-coded as follows:

> Green   Earth ground
> Black   Primary power (hot)
> White   Primary power return (neutral)

The TPC I requires the following source power:

> USA   115 VAC at 1.5 amperes
> International   230 VAC at 0.75 amperes

NOTE!   Appendix M gives directions for changing the TPC I to
              correspond to 230 volts, the international electrical
              power standard.

NOTE!   Incorrect or fluctuating line voltages can cause disk
              errors or damage the system.

Line Voltage Regulator

We recommend using a line voltage regulator. By using a line
voltage regulator, you can safeguard your system from power
surges and voltage spikes that may interfere with the normal
operation of your system. Contact your computer store for
information on availability and installation.

Switch Settings

Switch settings control many system functions. Most switches can
be set according to your preference, but others must be set in
required positions. During installation, it is important that
you check the switch settings to match your system requirements.

The TPC I has one DIP switch, located on the rear panel labeled
SW. This switch contains eight sections (they look like small
levers). These sections control various system functions.
Figure 2-2 illustrates the positions of the sections as they are
set by TeleVideo (default settings).

If you want to change any settings, see Table 2-1 for
alternative switch settings.

STOP!   The sections of the DIP switch are small individual
              switches. Gently push the switch to the desired
              position with a ball-point pen, and always give the
              switch a second push to make certain that it is seated
              properly in the position you have chosen. **DO NOT USE A
              PENCIL!** Pencil lead is an electrical conductor, and
              any small grains of lead falling into the switch
              sections may cause a malfunction.

Figure 2-2
Rear Panel Switch and Default Switch Settings
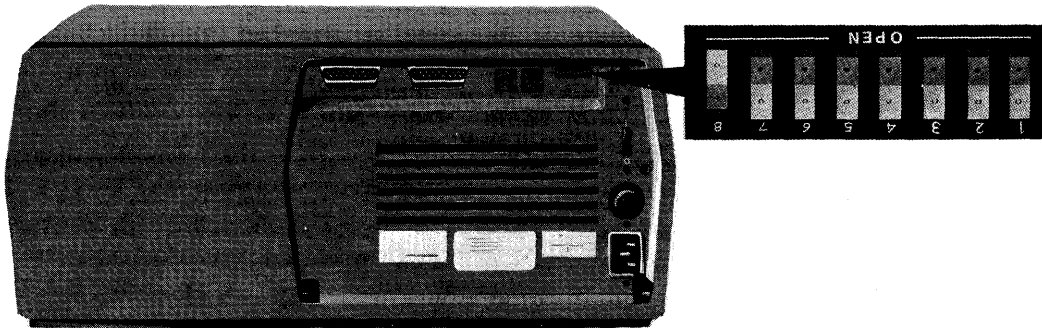
Table 2-1 lists the settings for the DIP switch. Read through
the table, and set the sections according to your requirements.
The starred (*) settings are the default settings.

Table 2-1
Switch Settings

Section	Setting (set to up or down)	Function
1	closed (down)	*Baud rate (see Table 2-2)
2	closed (down)	*Baud rate (see Table 2-2)
3	closed (down)	*Baud rate (see Table 2-2)
4	closed (down)	*Required
5	closed (down) open (up)	*Local Remote
6	closed (down)	*Not Used
7	closed (down) open (up)	*60 Hz (for 115V) 50 Hz (for 230V)
8	closed (down) open (up)	Black on yellow screen *Yellow on black screen

* Default settings

CONNECTING PERIPHERALS

Auxiliary devices such as a printer, a modem, a plotter, and the
SuperMouse can be connected to the TPC I. Instructions are given
here for connecting a printer and a modem. The parallel printer
interface allows the TPC I to be used with most printers that
support the Centronics parallel interface currently available on
the market. The TPC I is also equipped with an RS-232C-
compatible serial port for a modem that supports baud rates of
75, 150, 300, 600, 1200, 2400, 4800, and 9600.

The SETUPTPC Program allows you to quickly and easily change the
baud rate and the modem port data format to adapt to a variety of
peripherals. You can also indicate whether or not the SuperMouse
is attached. A complete description of the capabilities of the
SETUPTPC Program as well as step-by-step instructions for using
it, are in Chapter 5.

NOTE!   If you are adding a plotter or the SuperMouse, please
           follow the installation instructions that accompany
           those devices. Also see the SETUPTPC Program in
           Chapter 5.

Attaching Cables

Cables are needed to connect the TPC I to a printer or a modem.
The types of cables needed are determined by the requirements of
the device being attached to the TPC I. You can obtain the
appropriate cables for attaching peripheral devices at your
computer store.

Cable connectors commonly have D-shaped end connectors. (See
Figure 2-3.) These fit onto a D-shaped connector on the rear
panel of the system. To install a cable, turn the connector end
of the cable to fit the pin connector on the device, then gently
push the cable on to the connector. The screws can be finger
tightened to prevent accidentally pulling the cable off.

Figure 2-3
D-Shaped Cable Connector
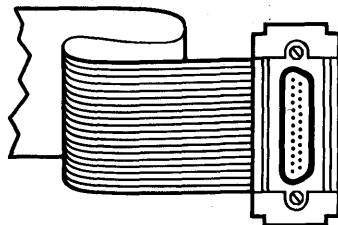
Leave some slack as you connect the cables. If you are using a round cable, coil it loosely and secure it with a rubberband. If you are using a flat, ribbon cable, fold it accordion-style as shown in Figure 2-4. Coiling ribbon cable can adversely affect system performance.

Figure 2-4
Correctly-Folded Excess Ribbon Cable
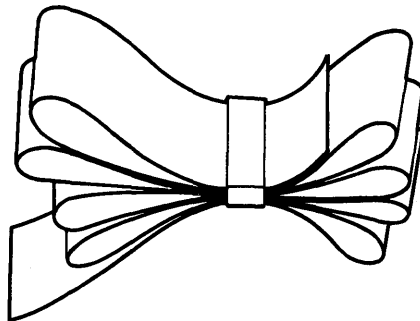
Connecting a Parallel Printer

The parallel port labeled **PRINTER** is ready to connect to a parallel printer when the unit is shipped from the factory. The TPC I is set to work with printers that support Centronics parallel interface printers. Be sure to read the instructions that came with your printer for correct interfacing information.

The TPC I and printer should be within 50 feet of each other.

Attach the end of the interface cable with the 25-pin D-shaped connector to the port labeled **PRINTER**. Attach the other end to the 36-pin connector on the printer. See Appendix E for a listing of the pin connector assignments.

Changing the Default Printer Device

To change the default printer device and to establish different power-up default values, refer to the instructions in Appendix G.

Connecting a Modem

You can connect a modem to the TPC I. The RS-232C port labeled **RS 232** is ready to connect to a modem.

The TPC I and modem should be within 50 feet of each other.

Attach one end of an RS-232C interface cable to the port labeled **RS 232.** Attach the other end to the modem's RS-232C connector.

The default baud rate for a modem is 9600. The baud rate setting can be changed by the SETUPTPC program (See Chapter 5), or by changing the switch settings. See Table 2-2.

Table 2-2
Modem Baud Rate Switch Settings

Dip Switch Section
     1 2 3   Baud Rate

     C C C   9,600 (default)
     O C C   4,800
     C O C   2,400
     O O C   1,200
     C C O   600
     O C O   300
     C O O   150
     O O O   75

Legend
C = Switch closed (down)
O = Switch open (up)

3. USING CP/M, THE OPERATING SYSTEM

CP/M stands for **Control Program/Monitor** or **Control Program for Microcomputers.** CP/M is a computer program, just like a word processor or accounts payable package and comes on a floppy diskette like other programs. CP/M is called an **operating system** because it controls all the operations and programs in your computer. The diskette that CP/M comes on is called the **system diskette** in this manual.

CP/M controls **all** of the operations in your computer. It controls the read and write operations to the disk drives. It loads the various applications programs that you want to run and supervises their execution. It organizes the data for communications to a printer or over a modem to a remote station. One way to think of CP/M is as the "personality" that resides in your computer: it is the overall set of commands that is used to run everything else on your system.

CP/M TERMS

The following table describes some of the basic technical terms used in computing. By becoming familiar with them, you will more easily be able to master the use of your computer.

Table 3-1
Technical Terms

Term	Definition
Cold Boot	On power up, or system reset, this is the process the computer goes through to check for hardware failures, clear memory, and load the operating system from the system diskette in drive A (in the TPC I). On the TPC I, a cold-boot is started by turning on the power or by pressing <Ctrl>/<Alt>/<Del> simultaneously.

Command Line   This is the text line that you type to the
                        right of the system prompt. The format of
                        the input line for each command is detailed
                        in the command description in the Transient
                        Commands in this chapter and the Utility
                        Programs chapter. Commands for editing the
                        command line, before you send it to the
                        computer with a <Return>, are described in
                        the CP/M Command Lines section of this
                        chapter.

Cursor   A display position indicator to emphasize a
                        character to be corrected or a position in
                        which data is to be entered or erased.

Logged Drive   The logged drive is the disk drive that CP/M
                        is currently working from. The letter in
                        the system prompt indicates the currently
                        logged drive. If you enter a transient
                        command, this is the drive that CP/M
                        searches to load the command.

Operating System   A collection of programs that gives
                        instructions to the computer for the
                        execution of applications programs on a
                        computer.

Resident Program   The kernel of CP/M is loaded once into a
                        special area in memory, and resides there
                        until memory is cleared with a cold boot or
                        reset. The resident commands are a part of
                        this kernel and the system can execute them
                        without referring to the disk for more
                        information.

System Diskette   The diskette that contains the CP/M operating
                        system that came with the TPC I. It
                        contains the resident commands on tracks 0
                        and 1, and the transient commands in .COM
                        files.

System Prompt   The system prompt looks like this:

                            A>

                        It is displayed at the left side of your
                        screen and indicates that CP/M is ready for
                        you to enter a command.

Transient Program   This is any applications program or CP/M
                        command that is stored on the diskette in a
                        .COM file. When you issue a command to use
                        the programs in these files, the programs are
                        loaded into the user memory area (transient
                        program area) and executed.

Warm Boot   By entering a <Ctrl>/C you cause CP/M to
                   restart (press the <CTRL> and C keys
                   simultaneously).

User Memory Area   The user memory area is the central region of
                   the Random Access Memory (RAM) in your
                   computer. The areas above (high memory) and
                   below (low memory) are reserved for CP/M.

                   The user memory area is where transient
                   programs are loaded and executed, while CP/M
                   surrounds and supervises the execution. CP/M
                   transient commands (such as STAT and PIP)
                   and applications programs are stored on the
                   diskette in .COM files until they are called
                   into memory by your command.

                   The user memory area is also called the
                   transient program area or TPA.

STARTING UP CP/M

There are several steps for starting your system and activating
the operating system. The following is a quick review of those
steps as they relate to CP/M.

System Diskette

In the User's Manual you learned how to set up your TPC I for
comfortable and safe operation. It is important that you make
a copy of the master system diskette and use the copy as your
everyday working copy. See the chapter, Utility Programs, for
instructions on how to make a working copy of the system diskette
using the COPYDISK utility program. Keep the original system
diskette as a backup in case your working copy is damaged or
wears out.

Start Up Review

Open the floppy drive by turning the latch to the vertical
position and turn the computer's power switch ON. This message,
followed by a blinking cursor, appears on the screen telling you
that the **cold boot** self-tests are running:

    System "boot" (x-Rev.x) from floppy disk in progress

If no hardware errors occur, insert the system diskette and close
the floppy drive door by turning the latch to the horizontal
position. CP/M displays this sign-on message:

        TeleVideo System TPC I Vx.x
        (c) 1983 TeleVideo Systems, Inc.
        xxK CP/M version y.y (double or single drive)

        A>

Reset Command

The following paragraphs discuss how to use the CP/M commands.
If you enter a command that doesn't seem to work, or the system
"freezes" after you enter a command, remember that you can always
reset (cold boot) CP/M by pressing the keys

        <Ctrl>/<Alt>/<Del>

Press the <Ctrl>, <Alt>, and <Del> keys simultaneously. This
command causes the computer to clear its memory and reload the
CP/M operating system. This is the same as turning the power off
and on. If this does not work, try turning the power off and on,
being sure to open the drive doors first. Close the drive doors
after you turn on the power. If you still have problems, run
through the installation procedure in Chapter 2 a few times and
check the Troubleshooting section in Chapter 8. If the problem
persists, contact your computer store.

SYSTEM PROMPT AND THE LOGGED DRIVE

When CP/M is successfully loaded into memory, the system prompt
is displayed at the left margin of your screen:

        A>

The system prompt tells you that CP/M is waiting for your next
command, which will be displayed just to the right of the
greater-than (>)character as you type it in. The **A** in the prompt
tells you that the **logged drive** is logical drive A.

The **logged** drive is the disk drive that CP/M is currently **working
from.** For example, if you want to look up the status of a file
on disk using the **STAT** command and you don't specify which disk
drive the file is on, CP/M assumes that you mean the logged
drive. You can change the logged drive with this command:

        A>b:<Return>
        B>

The second line is CP/M's response, showing you that the logged
drive is now B and that it is ready for another command.

CP/M COMMAND LINES

CP/M takes your commands in the form of a **command line.** A command line is a string of characters that names the command and supplies any data necessary (such as a filename). The command line is terminated and sent to the computer for processing when you press the <Return> key.

CP/M also accepts several control characters as commands such as ^C. These are directly interpreted without entering <Return>, and are useful for editing a command line or entering an override command.

Upper- and Lower-Case

You may have noticed that CP/M accepted a lower-case b in the example above, but displayed an upper-case B in its response. CP/M does not distinguish between cases on characters that you type. You may find it convenient to enter commands in lower-case so that you can distinguish your entries from CP/M responses on the screen. When you name a file with lower-case letters, the name is displayed in upper-case letters in the diskette directory.

NOTE!   Please note that this is only a characteristic of CP/M; other programs, such as BASIC, do not behave in the same way.

Control Characters

The key marked **Ctrl** is the CONTROL key. To use the <Ctrl> key, press the <Ctrl> and a character key simultaneously. For example, the command to restart CP/M is:

     <Ctrl>/C

Press down the CTRL and C keys simultaneously.

You use this command often to abandon a program and to restart CP/M. This form of restart is called a **warm boot** because CP/M is already loaded in memory and is not reloaded from the system diskette.

Line Editing

CP/M has several control characters and single-key commands that allow you to edit a complex command line before you send it to the computer with a <Return>. Table 3-2 summarizes the available functions.

Table 3-2
CP/M Control Functions

Key	Function
^C	System warm boot; doesn't clear user memory.
^E	Physical end of line; cursor wraps to beginning of next line, but <Return> not sent.
^H	Backspace; moves cursor back one space and deletes character that was there.
BACK SPACE	Same as ^H.
DEL	Delete character to the left of cursor when used in the sequence DEL<Return>. The character(s) that are to be deleted are displayed (echoed) on the screen.
^I	Moves the cursor to the next tab (8, 16, 24...).
^P	Activates the printer; deactivates the printer with a second ^P (toggle).
^R	Re-display command line; useful after DEL echoes deleted characters on command line.
^S	After a file or a list starts to display on the screen, ^S starts or stops (toggles) the scrolling of the display.
^U	Cancel current command line; positions cursor under command line for easy re-typing.
^X	Backspace to prompt (erases current line).
Return	Send command line to computer; <Return>.
<Ctrl>/<Alt>/<Del>	System cold boot; clears memory and reloads CP/M from drive A.

DISK FILES

The **disk file** is the basic unit of organization in CP/M. All
resident commands refer to file operations. Each diskette
directory can hold up to 64 file entries, but large files may
prevent you from using all 64. A file can hold any kind of
information: text, machine language program code, or graphics
commands. Every file is labeled with a **unique filename** so that
the computer can always find the exact file specified in a given
command.

Filenames

A filename is composed of three parts:

     drive prefix:filename.extension

A full unique filename might be:

     A:ACCOUNTS.BAS

The **drive prefix** is a single letter that corresponds to a specific drive. By leaving the drive prefix and colon off the unique filename, CP/M assumes the currently logged drive.

The **filename** is a unique combination of letters and numbers that identifies the file in the disk directory. You can use one to eight characters in the filename, mixing numbers and letters any way you want to arrive at a file-naming scheme that is useful to you.

The **extension** is designated by a period at the end of the filename, followed by three characters. It is an extension of the filename and is useful for identifying different types of files in the disk directory. Table 3-3 lists some common filetype extensions.

Table 3-3
Common Filetype Extensions

Extension	Definition
.ASM	Assembly program source file
.BAK	Back-up file
.BAS	BASIC program source file
.COB	COBOL program source file
.COM	CP/M executable file (transient command)
.DOC	Document file
.HEX	Intel hexadecimal code file
.REL	Relocatable machine code file
.SUB	SUBMIT command file
.TXT	Text file
.$$$	System temporary file

Wildcard File References

In the example above, you saw that you could leave off the drive prefix so that CP/M would supply the currently logged drive. This is handy for speeding up your file manipulation work. You can also insert special **wildcard** characters into a file specification so that CP/M can find multiple files based on a controlling prefix or extension.

* (asterisk)	In the place of a filename or file extension means "substitute all directory entries here."
? (question mark)	In the place of one or several character positions in the file specification means "substitute all possible characters here."

When CP/M finds a wildcard in a file specification it tries to do a file-match. It performs the desired command on all files that match the specification. For example:

    A>STAT<BAR>*.BAS<Return>

returns the status for all .BAS files on the logged drive (no drive prefix was given).

    A>stat<BAR>b:sam.*<Return>

returns the status for all files named SAM on drive B, regardless of their filetype extension. (Remember, CP/M converts lower-case to upper-case when you enter a command.)

Use the "?" wildcard for single- and multiple-character substitutions:

    A>era<BAR>d???.*<Return>

erases all files on the logged drive that begin with "d" followed by up to three characters. (It also erases files that begin with "d" and have less than four characters.) This is known as a file-match "mask." The following command erases all files beginning with the letter d:

    A>era<BAR>d*.*<Return>

RESIDENT COMMANDS

When CP/M is loaded into memory it includes a **kernel** (core group) of built-in or **resident** commands. These are executable directly from their "home" in memory.

CP/M also has a set of **transient commands** which stay on the system disk until they are needed. See the section, Transient Commands, in this chapter. This section describes the CP/M command line format for each of the **resident commands.** The resident commands discussed in this chapter are:

DIR	List a diskette directory
ERA	Erase a file
REN	Rename a file
TYPE	Display or list a file on the screen

Unrecognized Commands

If you type in a command that is not recognized by CP/M, usually by making a typing error, CP/M repeats the command as you entered it on the next line followed by a question mark:

    A>statt<Return>
    STATT?

    A>

DIR - List a Directory

The DIR command displays a directory of the files on a diskette.

    A>dir<BAR>x:filename.ext<Return>

By typing **dir<Return>** you get a full directory listing for the currently logged disk. You can narrow the listing down, or list the directory for a different drive by supplying a file specification with or without "wildcard" characters.

ERA - Erase a File

The ERA command erases the file or files specified.

    A>era<BAR>x:filename.ext<Return>

In the special case of "erase all files" on a diskette:

    A>era<BAR>b:*.*<Return>

ERA responds with a safety verification:

    ALL (Y/N)?

Type **Y** to erase all the files, **N** to cancel the command, followed by **<Return>.**

REN - Rename a File

The REN command is used to rename a file in the diskette directory.

    A>ren<BAR>newname.ext=oldname.ext<Return>

Notice that the replacement goes right-to-left, that is, the existing file comes second in the command line and will end up with the name at the beginning. This "parameter order" is important! See the transient command PIP also.

If you enter as a new name the name of a file that already exists, you get the message:

     FILE EXISTS

If you enter an incorrect file to be renamed, the computer won't be able to find it, and displays the message:

     NO FILE

TYPE – List a File on the Screen

This command displays the contents of a file on the screen:

     A>TYPE<BAR>x:filename.ext<Return>

The computer sends information to the screen as fast as it can scroll, so a long file seems to fly by. Use the ^S command to stop and start the scrolling action.

TRANSIENT COMMANDS

CP/M comes with a set of very useful utility programs called **transient commands.** Some of them are powerful program development aids such as the text editor ED, a dynamic debugging monitor DDT, and a file printing routine called DUMP. You can see a list of the transient commands that came on your CP/M system diskette by typing:

     A>DIR<BAR>*.COM<Return>

The filetype extension **.COM** is used to denote a file that contains an executable CP/M program or "COMmand." Buying programs is a matter of adding to the number of .COM files available to your system. For example, the word processing program TeleWrite is usually stored in a file called "TW.COM." So the CP/M transient command to run TeleWrite is:

     A>TW<Return>

The program is loaded from disk into the user memory area, or **Transient Program Area** (TPA), and begins execution.

Two of the transient command programs supplied with CP/M are very useful to the beginning computer operator, and they are described below. For a description of the more advanced program development tools, refer to one of the books recommended in Appendix B.

The two transient commands discussed below are:

STAT   List the STATus of a file. Use this to get an
             alphabetical column display of the files on your
             disk.

PIP   The Peripheral Interchange Program. Use this to
             move and to copy files.

STAT – List the Status of a File

The STAT command is used for a variety of tasks. The primary
function is to list the status of a file, or files, or of an entire
diskette. This status list is displayed on the screen. STAT can
also be used to change the protection status on a file or group
of files. Finally, STAT can be used to change the status of the
logical device assignments on your system.

To list the status of the files on a diskette enter:

A>stat<BAR>*.*<Return>

This causes a column listing of the diskette files in
alphanumeric order. This is often more useful than the DIR all-
on-a-line listing. In addition to the sorted listing, STAT
displays other useful information about each file. The STAT
listing looks like this:

```
Recs  Bytes  Ext Acc

   4    12k    1  R/W  A:ASM.COM
  13    62k    5  R/W  A:DUMP.COM
  11    43k    3  R/O  A:PIP.COM
   3     9k    1  R/W  A:STAT.COM
```

where:

Recs   is the number of 128-byte records in the file.
Bytes is the length of the file in bytes.
Ex   is the number of logical extents (16k-byte blocks).
Acc   is the file access status.

The file access status can be used to protect valuable files from
unwanted overwriting or accidental erasure. R/W means that a
file can be read from and written to. R/O means that a file can
be read only. To change the access status of a file use the
command line:

A>stat<BAR>x:filename.ext<BAR>$R/O<Return>

By adding the $R/O to the STAT you are adding a parameter to the
command line. STAT relabels the file as Read Only. You can use
the STAT command again to change the file back to Read/Write.

PIP - Peripheral Interchange Program

PIP is used to copy files from one disk drive to the other. The
basic format for a PIP command is:

> **PIP<BAR>y:file2.ext=x:file1.ext[zzz]<Return>**

where:

file2.ext	is the destination file, where the copy will end up
file1.ext	is the source file, where the copy comes from
zzz	are parameters that allow special handling
x and y	are the disk drives

Notice that, like REN, the action proceeds from right-to-left on
the command line.

To copy a file from drive A to drive B, use the command line:

> **A>pip<BAR>b:=a:myfile.bas<Return>**

There are now two copies of the same data, one in the file
"MYFILE.BAS" on drive A, and one in the file "MYFILE.BAS" on
drive B. You may, or may not want to ERAse the old copy (disk
space is a premium, so developing good housekeeping practices can
help you get the most out of your system. Don't wait for a disk-
full error message!)

By omitting any drive specifications and using a different
filename, you can make copies of a file on the same disk.

> **A>pip<BAR>yourfile.bas=myfile.bas<Return>**

One special parameter is very useful, and easy to append to the
PIP command line:

v	causes PIP to verify that the copy is an exact duplicate of the original file by doing a byte-by-byte comparison. This may cause delays on long (200K plus) files.

To use this parameter in the file transfer from the above
example, use the command line:

> **A>pip<BAR>b:=a:myfile.bas[v]<Return>**

This command assures that the file is copied with no errors. The
parameters are enclosed in square brackets, and added directly
after the file extension.

For more details about the PIP command, refer to one of the
excellent CP/M books listed in Appendix B.

APPLICATIONS SOFTWARE

CP/M is the most popular operating system for microcomputers.
This means that the widest possible library of applications
software programs is available for your TPC I. You may want to
purchase additional software for some of the following
applications:

    *   word processing
    *   accounting
    *   financial analysis
    *   business graphics
    *   program development
    *   entertainment

Included with the TPC I are three applications programs:

    *   TeleWrite for executive word processing
    *   TeleCalc for spreadsheet analysis
    *   TeleChart for graphics presentation of spreadsheet data

A minimum of nine CP/M-compatible programming languages including
ALGOL, APL, BASIC, C, COBOL, FORTH, FORTRAN, PASCAL, and PL/1 can
be used on TeleVideo systems for your programming applications.

Ask the computer software specialists at your computer store for
name of software programs and languages run well on your TPC I.

4.   KEYBOARD

The TPC I has a detachable, low-profile keyboard that includes a numeric keypad and programmable function keys. This chapter discusses the types of keys on the keyboard and lists each key and its function.

KEY DESCRIPTIONS

The keyboard has a separate numeric keypad for accounting applications and ten programmable function keys in addition to the alphanumeric and character keys of the main keypad. All of the keys on the numeric keypad perform one function when the <Num Lock> key is pressed, and another function when the <Num Lock> key is pressed again. The <Num Lock> key acts as a toggle switch changing the keys on the numeric keypad from the numeric mode to the cursor movement and editing mode.

Figure 4-1
Character Keys
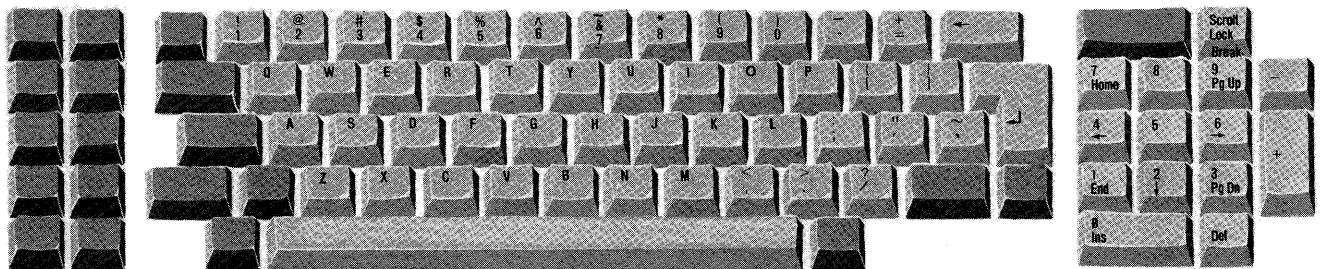
Character Keys

The character keys highlighted in Figure 4-1 include all alphabet characters (A through Z), numerals (0 through 9), punctuation marks, and mathematical symbols. All character keys repeat when pressed for more than one-half second.

There are several special keys that make inputting information easier. This chapter includes a description of the special keys

on the main keypad and on the numeric keypad, and of the function keys for a programmer.

Many of the special keys are not programmed to be used as labeled when you are using CP/M, the operating system. They are available for customization by a programmer to a specific applications program. Therefore, if you use them, you may see some symbols and characters on the screen that do not seem to make any sense.
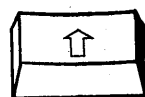
Figure 4-2
Function Keys and Special Keys
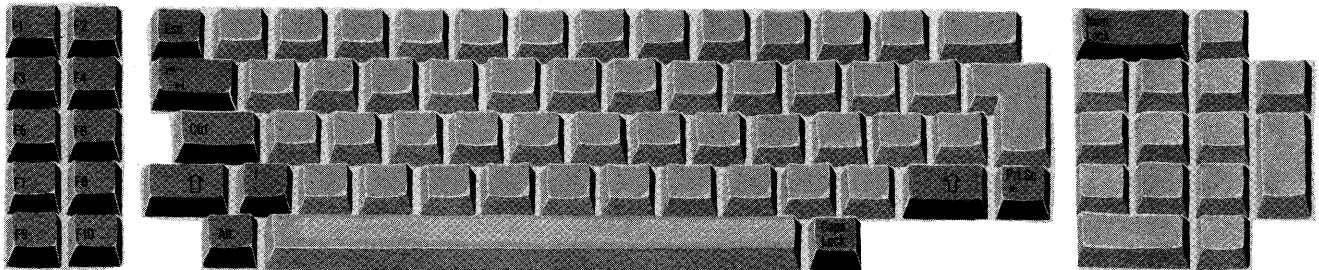
Function Keys and Special Keys

Table 4-1 summarizes the function of the special keys which are highlighted in Figure 4-2.

Table 4-1
Key Functions

Keycap Label	Description
Space Bar	Causes a blank space to appear on the display and transmits an ASCII space code.
	The <Shift> key (wide, up arrow), located on each side of the main keypad, selects the upper character inscribed on a key, changes operation of most special keys, and capitalizes alpha characters.

Toggles the alpha keys so that they transmit codes for upper-case characters. Press again to toggle back to lower-case characters.
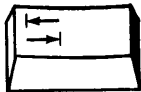
The <Tab> key (forward and backward arrow to line) moves the cursor forward to typewriter tabs. The cursor moves to the next tab setting in that line. When it reaches the last tab setting in that line, it stops. If the cursor is beyond the last tab set on a line, pressing <Tab> has no effect.
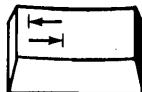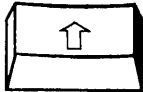
Moves the cursor backward to typewriter tabs. If no tab has been set between the first column position and the cursor position, the cursor moves to the first column position.
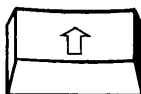
These keys, when pressed simultaneously, enable the user to toggle the display from graphics to alphanumerics and vice versa without affecting the running program.

Generates 32 normally-nondisplayed ASCII control codes when used with another key.
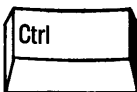
The control key combinations are used for special action by the terminal and/or the applications program in the computer.

When the <Ctrl> key is used with an alpha or symbol key, the output data of the character which is typed becomes 00 through 3F Hex, thus changing the code transmitted by that character. For example, if **M** alone is pressed, the code for **M** is sent. If you press ^M, the code for <Return> is sent.

The <Esc> key sends an ASCII code for escape to the display processor. The key is generally used to momentarily leave (escape) an applications program in order to use a special feature or function.

Another function of the <Esc> key is to display on the screen the next control character entered. This avoids having to use the system mode monitor feature.

When the display is in the alphanumeric mode, the <Shift>/<Esc> key sequence allows you to change or access the display attributes without transmitting them to the computer. After the <Shift>/<Esc> key sequence is pressed, the next character entered is interpreted as a command by the terminal.

NOTE!   The <Esc> key is used with another character(s) in the command sequence; i.e., **the <Esc> key is pressed and released before the second key is pressed.**
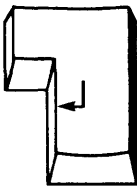
The <Return> key (squared, left arrow located on the right of the main keypad) is used like the carriage return on a typewriter. It sends the ASCII code for a carriage return <Return> to the display processor within the TPC I. Depending on the communication mode used, the code causes the display processor to transmit a <Return> to the computer.

NOTE!   The computer features an auto wraparound function that eliminates the need to manually enter an <Return> and a linefeed (LF) at the end of each 80-character line.
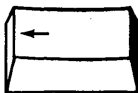
The left arrow key on the top right of the main keypad acts like a back space key on a typewriter. It moves the cursor one character to the left. If the cursor is at home, it has no effect.

F1 through F10 (Function Keys)   Each function key sends a three-code sequence to the computer. This sequence may initiate a special form or subroutine in the program that causes the video display to display or perform a particular function. Six more function keys may be obtained if needed by pressing <Ctrl>/F1 - F6.
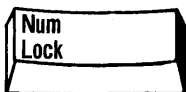
NOTE!   The following keys function as described when the <Num Lock> key is on the numeric keypad is pressed. You can toggle back and forth between the numeric mode (where the keys operate as numerals) and special key mode (where the keys operate as described here) by pressing the <Num Lock> key.

The <Home> key moves the cursor to column one of row (line) one. This position is referred to as the **home** position.

The left arrow key numeric keypad moves the cursor one character to the left. If the cursor is at home, it has no effect.

The up arrow key on the numeric keypad moves the cursor key is pressed) up one line. If the cursor is on the top line, it has no effect.

The down arrow key moves the cursor down one line. If the cursor is on the bottom line of the screen, it has no effect.

The right arrow key moves the cursor one character to the right. If the cursor is at the end of the line, it goes to the first column position of the next line. If the cursor is at the end of the screen, it scrolls up the screen.
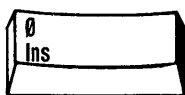
Pressing <Ins> alone or <Shift>/<Ins> or <Caps Lock>/<Insert> inserts a line consisting of spaces at the cursor position. This causes the cursor to move to the start of the new line and all following lines to move down one line, resulting in the loss of the last line of the page. If half intensity is on, half-intensity spaces replace the erased characters.
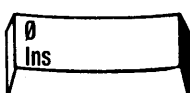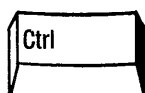
Pressing <Ctrl>/<Ins> or <Ctrl>/<Shift>/<Ins> causes the character at the cursor to move right one column position and enters a space at the cursor position. As characters are inserted, characters reaching column 80 are lost. If half intensity is on, half-intensity spaces replace the erased characters.

Pressing the 5 key on the numeric keypad deletes all characters from the cursor to the end of the line.

The <Del> key sends an ASCII DEL character to the computer. The cursor moves to the right displaying the character(s) deleted.

Pressing <Shift>/<Del> or <Shift>/<Ctrl>/<Del> deletes the character at the cursor position and moves all following characters left one position. At the end of the delete function, a space is written into the last position on the line. If half intensity is on, half-intensity spaces replace the erased characters.

Pressing <End> deletes the line at the cursor position and all following lines move up one line. The cursor moves to column one of the line and spaces are loaded into the last line of the page. If half intensity is on, half-intensity spaces replace the erased characters.

These keys are not supported on the TPC I:

    PrtSc
    PgUp
    PgDn
    Scroll Lock/Break

DISABLING/ENABLING THE KEYBOARD

You can disable (lock) all keys and the function keys by using a command sent from the computer.

Once the keyboard is disabled, it can only be enabled by another command from the computer or by resetting the system.

STOP!   If your applications program echoes all codes, the keyboard may be accidentally disabled.

To disable the keyboard, send:

    ESC ‡

The keyboard remains disabled until one of the following occurs:

    The terminal emulator receives an ESC " sequence.

    You reset the system by pressing <Ctrl>/<Alt>/<Del> simultaneously. This might result in data loss.

KEYCLICK AND BELL

You can control the keyclick (the noise made as each key is activated) and the bell with the code sequences in Table 4-2.

Table 4-2
Keyboard Audio Commands

Function	Command
Keyclick on	ESC>
Keyclick off	ESC<
Ring bell	^G

FUNCTION KEYS

The function keys (F1 through F16) send a user-defined or default code. For example, the user-defined code may be a frequently-used escape or control code sequence in a text editing application.

There are ten function keys, but you can obtain six more function keys by pressing <Ctrl>/F1 through F6 simultaneously which become F11 through F16. Using them in combination with <Shift> allows up to 32 sets of codes to be transmitted.

Programming the Function Keys

When the terminal is first turned on, the function keys are already programmed with default messages. If you do not program the function keys, the default values shown in Table 4-3 are transmitted by each function key.

Table 4-3
Default Function Key Values

Key	Unshifted Code	Shifted Code	Ctrl/ Unshifted	Ctrl/ Shifted
F1	^ A @ CR	^ A ` CR	^ A J CR	^ A j CR
F2	^ A A CR	^ A a CR	^ A K CR	^ A k CR
F3	^ A B CR	^ A b CR	^ A L CR	^ A l CR
F4	^ A C CR	^ A c CR	^ A M CR	^ A m CR
F5	^ A D CR	^ A d CR	^ A N CR	^ A n CR
F6	^ A E CR	^ A e CR	^ A O CR	^ A o CR
F7	^ A F CR	^ A f CR		
F8	^ A G CR	^ A g CR		
F9	^ A H CR	^ A h CR		
F10	^ A I CR	^ A i CR		

You may program the function keys by following the procedure described here.

Enter the following code in the exact sequence shown:

    <Shift>/<Esc> | p1 p2 message ^Y

where

      p1   is the number of the function key. The values of p1 are in Table 4-4.

Table 4-4
Function Key Values

Key	Unshifted	Shifted	Ctrl/ Unshifted	Ctrl/ Shifted
F1	1	<	;	F
F2	2	=	G	L
F3	3	>	H	M
F4	4	?	I	N
F5	5	@	J	O
F6	6	A	K	P
F7	7	B		
F8	8	C		
F9	9	D		
F10	:	E		

      p2   is the following value:

          1 = Send to the computer

message   can contain up to 63 bytes per key

    ^Y   is the termination character

Because the control, escape, cursor position, and similar function keys are not normally stored, a ^P embedded in the text of the function key message may be used to store the next character entered.

For example, assume that the message desired for key Fl is:

    TURN ON THE PRINTER

Precede this message with the following:

    <Shift>/<Esc>|
    The key number (1 for key Fl)
    The value 1 (for send to computer)

The entry is:

    <Shift>/<Esc> | 1 1 TURN ON THE PRINTER <Return> ˆY

NOTE!   1.   Entering the message followed by <Return> enters a
                     carriage return at the end of the function.

               2.   Entering the message without <Return> then
                     requires the user to manually press <Return> to
                     execute the command.

5.   UTILITY PROGRAMS

TeleVideo has supplied a set of **utility programs** to facilitate the use of the TPC I. These programs perform several housekeeping functions such as formatting blank diskettes, making backup copies of the operating system, and copying diskettes. The SETUPTPC utility program allows you to customize the TPC for specific peripherals.

The four utilities included on the system diskette are:

SETUPTPC   SETUPTPC is a menu-driven utility software program for customizing the CP/M operating system running under TPC hardware.

SYSGEN   Generates a resident CP/M system on tracks 0 and 1 of a new diskette or a master applications program diskette. Use this to copy the portion of CP/M that has the resident commands to another diskette.

FORMAT   Formats a blank diskette for use on the TPC I. The directions are divided into two parts - a section for the single-drive TPC and a section for a dual-drive TPC.

COPYDISK   Copies data from one diskette to another diskette. The directions are divided into two parts - a section for the single-drive TPC and a section for a dual-drive TPC.

The utility programs differ from the standard transient commands in that they require more operator interaction than the single command line format. Each program is illustrated below in a SYSTEM/USER dialogue, or script. Just type in your part!

NOTE!   If you have one drive, the screen messages tell you when to remove the diskette in the drive and replace it with another.

If you have two drives, always put the system diskette in drive A (the left drive). The screen message tells you when to insert a diskette in drive B.

SETUPTPC PROGRAM

SETUPTPC is a menu-driven utility software program for customizing the CP/M operating system running under TPC I hardware. SETUPTPC provides the capability to change systems attributes, such as baud rate and printer protocol. For example, a user need only specify the new baud rate to change the baud rate of the RS-232 modem port. SETUPTPC makes the change to the system. This change is effective after the next reset and is permanent.

The capability of the SETUPTPC program is as follows:

1.   Changing the default I/O BYTE

      After power-on or reset, a default physical device is
      selected for each logical device based on the value of the
      default I/O BYTE. The default I/O BYTE may be changed to
      select any one of the four physical devices associated with
      logical devices.

2.   Configuring the modem port

      The modem port can be configured to be used as a second
      printer port for different printer protocols.

3.   Configuring the port attribute

      The port attribute configuration allows for baud rate
      specification for an external device such as a printer or
      modem, and for transmit/receive data format.

4.   Turning on and off the SuperMouse hardware flag for
      graphics

5.   Configuring the autoload feature

      Any command file may be automatically executed during the cold
      boot. Cold boot occurs after power-on or a system reset has
      been performed.

The following example illustrates how to change the baud rate for the RS-232 port.

Be sure that the file, SETUPTPC.COM exists in the currently-logged drive.

USER: 1.   Enter

                  SETUPTPC<Return>

SYSTEM: 2.   Displays

                 SYSTEM   SETUP   Version x.x   TeleVideo Systems, Inc.

                 Please select the following system type:

                      1)   Stand alone TPC I
                      2)   Multi user TPC I
                      3)   Abort SETUP and return to CP/M

                 - <--Enter appropriate number followed by RETURN
                      or CTRL-C to return to CP/M

USER: 3.   Enter

                 1<Return>

SYSTEM: 4.   Displays

                 SYSTEM   SETUP   Version x.x   TeleVideo Systems, Inc.

                 Current system: TPC I
                 Operating system: Stand alone CP/M
                 Current Mouse status: OFF

                 Select modification category:
                      1)   To attach mouse
                      2)   To detach mouse
                      3)   To change CP/M default I/O byte
                      4)   To change Modem port BAUD rate
                      5)   To change Modem port data format
                      6)   To change Autoload file name
                      7)   To save current configuration permanently
                           and return to CP/M
                      8)   To save current configuration temporarily
                           and return to CP/M
                      9)   To abandon this setup

                 -- <--Enter appropriate number followed by RETURN
                       or CTRL-C to return to CP/M

USER: 5.   Enter

                 4<Return>

SYSTEM: 6.   Displays

           SYSTEM SETUP   Version x.x   TeleVideo Systems, Inc.

           Current system: TPC I
           Operating system: Stand alone CP/M
           Current Modem port BAUD rate: 9,600 DIPSW

           Please select the following baud rate:

              1)   9,600 BAUD
              2)   4,800 BAUD
              3)   2,400 BAUD
              4)   1,200 BAUD
              5)   600 BAUD
              6)   300 BAUD
              7)   150 BAUD
              8)   75 BAUD
              9)   Set BAUD rate from switch
            10)   Use current BAUD rate and exit

         -- <--Enter appropriate number followed by
               RETURN or CTRL-C to return to CP/M

USER: 7.   Enter

           4<Return>

           (if you wish to change baud rate to 1200)

SYSTEM: 8.   Displays

           SYSTEM SETUP   Version x.x   TeleVideo Systems, Inc.

           Current system: TPC I
           Operating system: Stand alone CP/M
           Current Modem port BAUD rate: 1,200

           Please select the following baud rate:

              1)   9,600 BAUD
              2)   4,800 BAUD
              3)   2,400 BAUD
              4)   1,200 BAUD
              5)   600 BAUD
              6)   300 BAUD
              7)   150 BAUD
              8)   75 BAUD
              9)   Set BAUD rate from switch
            10)   Use current BAUD rate and exit

         -- <--Enter appropriate number followed by
               RETURN or CTRL-C to return to CP/M

USER: 9.   Enter

                 10<Return>

                 (if you wish to use the current BAUD rate and
                 exit)

SYSTEM: 10.   Displays

                 SYSTEM SETUP   Version x.x TeleVideo Systems, Inc.

                 Current system: TPC I
                 Operating system: Stand alone CP/M
                 Current Mouse status: OFF

                 Select modification category:
                     1)   To attach mouse
                     2)   To detach mouse
                     3)   To change CP/M default I/O byte
                     4)   To change Modem port BAUD rate
                     5)   To change Modem port data format
                     6)   To change Autoload file name
                     7)   To save current configuration permanently
                          and return to CP/M
                     8)   To save current configuration
                          temporarily and return to CP/M
                     9)   To abandon this setup

                 -- <--Enter appropriate number followed by RETURN
                        or CTRL-C to return to CP/M

USER: 11.   Type

                 7<Return>

                 (if you wish to save current configuration
                 permanently)

SYSTEM: 12.   Displays

                 End of execution

                 A>

                 (if the system you are currently logged onto is
                 drive A.)

USER: 13.   Reset the system by pressing simultaneously

                 <Ctrl>/<Alt>/<Del>

The baud rate for the RS-232 port should now be permanently
changed to 1200 baud.

SYSGEN

SYSGEN copies the system tracks (0 and 1) of the system diskette
to a new diskette. To use your applications program diskettes in
drive A, you must run SYSGEN. This allows you to boot from your
applications program diskette and use drive B for your data
files.

USER: 1.   Insert the system diskette into the drive (drive A
                  if you have two drives).

             2.   If the system prompt is not displayed, reset the
                  TPC I by pressing <Ctrl>/<Alt>/<Del>
                  simultaneously. If the system prompt is
                  displayed, go to step **4.**

SYSTEM: 3.   Displays

                  TeleVideo System TPC I Vx.x
                  (c) 1983 TeleVideo Systems, Inc.
                  xxK CP/M version y.y (double drive)

                  A>

USER: 4.   Type

                  SYSGEN<Return>

SYSTEM: 5.   Displays

                  SYSGEN Vx.x
                  (c) 1982 TeleVideo Systems, Inc.

                  SOURCE DRIVE NAME (OR RETURN TO SKIP)

USER: 6.   Enter

                  A

SYSTEM: 7.   Displays

                  SOURCE ON A:, THEN TYPE RETURN

USER: 8.   Press

                  <Return>

SYSTEM: 9.   Displays

                  FUNCTION COMPLETE
                  DESTINATION DRIVE NAME (OR RETURN TO REBOOT)

USER: 10.   a.   If you have one drive, remove the system
                     diskette from the floppy drive and insert a
                     formatted diskette or an applications program
                     master diskette in the floppy drive and enter

                     A

                b.   If you have two drives, insert a formatted
                     diskette or an applications program master
                     diskette in drive B and enter

                     B

SYSTEM: 11.   a.   Displays (if you have one drive)

                     DESTINATION ON A:, THEN TYPE RETURN

                b.   Displays (if you have two drives)

                     DESTINATION ON B:, THEN TYPE RETURN

USER: 12.   Press

                <Return>

SYSTEM: 13.   Displays

                FUNCTION COMPLETE

                (The destination diskette now contains the CP/M
                resident program on tracks 0 and 1.)

                DESTINATION DRIVE NAME (OR RETURN TO REBOOT)

USER: 14.   a.   To copy the CP/M tracks 0 and 1 on another
                     diskette, remove the destination diskette,
                     insert another diskette, and go to Step 10.

                b.   To end the program, press

                     <Return>

SYSTEM: 15.   Displays

                A>

FORMAT (SINGLE DRIVE)

FORMAT formats floppy diskettes. All new diskettes must be
formatted before they can be used. Used diskettes containing
data can also be formatted. If a used diskette is formatted, all
data on the diskette is erased; once formatted, the used diskette
is ready for new data.

THESE DIRECTIONS ARE FOR THE SINGLE-DRIVE TPC ONLY!

STOP!   **FORMAT OVERWRITES ALL DATA PRESENTLY ON THE DISKETTE;
          ANY FILES AND PROGRAMS ON THE DISKETTE ARE ERASED.**

USER: 1.   Put the system diskette in the drive.

          2.   If the system prompt is not displayed, reset the
                TPC by pressing <Ctrl>/<Alt>/<Del> simultaneously
                or by turning the power off and on. If the system
                prompt is displayed, go to Step 4.

SYSTEM: 3.   Displays

                TeleVideo System TPC I Vx.x
                (c) 1983 TeleVideo Systems, Inc.
                xxK CP/M version y.y (single drive)

                A>

USER: 4.   Enter

                FORMAT<Return>

SYSTEM: 5.   Displays

                FLOPPY DISK FORMAT Vx.x
                (c) 1983 TeleVideo Systems, Inc.

                1.   TS 801/802 AS STAND ALONE
                2.   TS 801/802 AS SATELLITE USERSTATION
                3.   TS 802H AS STAND ALONE/SATELLITE USERSTATION
                4.   TS 803/803H/TPC I AS STAND ALONE
                5.   TS 803/803H/TPC I AS SATELLITE USERSTATION
                6.   TS 806
                7.   EXIT

                Select system configuration option by number

USER: 6.   Type

                4

SYSTEM: 7.   Displays

                 TS 803/803H/TPC I AS STAND ALONE is selected.
                 Press RETURN if correct. (SPACE BAR to retry, ESC to abort.)

USER: 8.   Press

                 <Return>

SYSTEM: 9.   Displays

                 Insert a diskette to be formatted into floppy disk drive.
                 Press RETURN when ready. (ESC to abort.)

USER: 10.   Remove the system diskette and put the diskette
                 that you want to format into the floppy drive.

           11.   Press

                 <Return>

SYSTEM: 12.   Displays

                 formatting...

USER: 13.   Listen for a clicking sound as the program formats
                 the diskette.

SYSTEM: 14.   Displays

                 Formatting Completed.

                 Format another?
                 Insert a diskette to be formatted into floppy disk drive.
                 Press RETURN when ready. (ESC to abort.)

USER: 15.   a.   To format another diskette, insert it in the
                       floppy drive and go to Step 11.

                 b.   To end the program, press

                       <Esc>

SYSTEM: 16.   Displays

                 Insert SYSTEM diskette back into the drive.
                 Press ANY KEY when ready.

USER: 17.   Press any key.

SYSTEM: 18.   Displays

                 End of Execution
                 A>

FORMAT (DUAL DRIVE)

FORMAT formats floppy diskettes. All new diskettes must be
formatted before they can be used. Used diskettes containing
data can also be formatted. If a used diskette is formatted, all
data on the diskette is erased; once formatted, the used diskette
is ready for new data.

THESE DIRECTIONS ARE FOR THE DUAL-DRIVE TPC I ONLY!

STOP!   **FORMAT OVERWRITES ALL DATA PRESENTLY ON THE DISKETTE;
              ANY FILES AND PROGRAMS ON THE DISKETTE ARE ERASED.**

USER: 1.   Put the system diskette in drive A.

           2.   Insert the diskette to be formatted into drive B.

           3.   If the system prompt is not displayed, reset the
                TPC by pressing <Ctrl>/<Alt>/<Del> simultaneously.
                If the system prompt is displayed, go to Step 5.

SYSTEM: 4.   Displays

                TeleVideo System TPC I Vx.x
                (c) 1983 TeleVideo Systems, Inc.
                xxK CP/M version y.y (double drive)

                A>

USER: 5.   Enter

                FORMAT<Return>

SYSTEM: 6.   Displays

                FLOPPY DISK FORMAT Vx.x
                (c) 1983 TeleVideo Systems, Inc.

                1.   TS 801/802 AS STAND ALONE
                2.   TS 801/802 AS SATELLITE USERSTATION
                3.   TS 802H AS STAND ALONE/SATELLITE USERSTATION
                4.   TS 803/803H/TPC I AS STAND ALONE
                5.   TS 803/803H/TPC I AS SATELLITE USERSTATION
                6.   TS 806
                7.   EXIT

                Select system configuration option by number

USER: 7.   Enter

                4

SYSTEM: 8.   Displays

                 TS 803/803H/TPC I AS STAND ALONE is selected.
                 Press RETURN if correct. (SPACE BAR to retry, ESC to abort.)

USER: 9.   Press

                 <Return>

SYSTEM: 10.   Displays

                 Drive?(A or B. ESC to abort.)

USER: 11.   Enter

                 B

                 (the drive indicating where the diskette to be
                 formatted is)

SYSTEM: 12.   Displays

                 Insert a diskette into floppy disk drive B
                 Press RETURN when ready.(ESC to abort.)

USER: 13.   Press

                 <Return>

SYSTEM: 14.   Displays

                 formatting.......................................

USER: 15.   Listen for a clicking sound as the program formats
                 the diskette.

SYSTEM: 16.   Displays

                 Formatting Completed.

                 Format another?
                 Drive?(A or B. ESC to abort.)

USER: 17.   a.   To format another diskette, insert it in
                       drive B and go to Step 11.

                 b.   To end the program, press

                       <Esc>

SYSTEM: 18.   Displays

                 End of Execution
                 A>

COPYDISK (SINGLE DRIVE)

COPYDISK copies the contents of one floppy diskette onto another floppy diskette. During the copying, the program checks for data transfer errors.

Use COPYDISK to copy the TPC I system diskette as well as applications program diskettes. Store the original program diskettes in a safe place as a backup to the working copy.

THESE DIRECTIONS ARE FOR THE SINGLE-DRIVE TPC ONLY!

NOTE!   Before you run COPYDISK, remember to format all new
             floppy diskettes before using them. If you do not know
             how to format a new diskette, refer to the utility
             program FORMAT.

USER: 1.   Insert the system diskette into the drive.

          2.   If the system prompt is not displayed, reset the
                TPC by pressing <Ctrl>/<Alt>/<Del> simultaneously.
                If the system prompt is displayed, go to Step **4**.

SYSTEM: 3.   Displays

                TeleVideo System TPC I Vx.x
                (c) 1983 TeleVideo Systems, Inc.
                xxK CP/M version y.y (single drive)

                A>

USER: 4.   Enter

                COPYDISK<Return>

SYSTEM: 5.   Displays

                COPYDISK (TS 803/TS 803H/TPC I) Vx.x
                (c) 1983 TeleVideo Systems, Inc.

                Warning:COPYDISK will destroy all data
                on destination disk with no chance of recovery.
                If a destination disk is inserted as a master disk by
                mistake, all data on master disk will be destroyed.
                Press RETURN when ready. (ESC to abort.)

USER: 6.   Press

                <Return>

SYSTEM: 7.   Displays

           Copying (Track) -->
           Insert " MASTER " diskette in the drive. Press RETURN when ready.

USER: 8.   Insert the source diskette to be copied into the
                 floppy drive.

           9.   Press

                 <Return>

SYSTEM: 10.   Displays

      Insert "DESTINATION" diskette in the drive. Press RETURN when ready.

USER: 11.   If you have one drive, remove the diskette
                 presently in the drive (the source diskette),
                 insert another diskette (the destination
                 diskette), and press

                 <Return>

          12.   Listen for a clicking sound as the program copies
                 the diskette.

SYSTEM: 13.   Displays alternately several times while copying

      Insert " MASTER " diskette in the drive. Press RETURN when ready.

      Insert "DESTINATION" diskette in the drive. Press RETURN when ready.

          14.   Displays when complete

                 Disk copy completed.

                 Copy another?
                 Warning:COPYDISK will destroy all data
                 on destination disk with no chance of recovery.
                 If a destination disk is inserted as a master disk by
                 mistake, all data on master disk will be destroyed.
                 Press RETURN when ready.(ESC to abort.)

USER: 15.   a.   To copy more diskettes, go back to Step 6.

                 b.   To end the program, insert the system
                       diskette in drive A and press

                       <Esc>

SYSTEM: 16.   Displays

          Insert SYSTEM diskette into the drive. Press ANY KEY when ready.

USER: 17.   Press any key.

SYSTEM: 18.   Displays

          A>

COPYDISK Error Messages

If this error message appears on the screen

    Data Transfer Error(s) on "Drive Name"
    Verification Error(s).

try one of the suggestions listed below.

1.   Run COPYDISK again, reversing the source and destination
     drive designations.

2.   Make sure that the diskette used as the destination is not
     defective or damaged; if necessary, use a new formatted
     diskette.

3.   Try copying another diskette.

4.   Run the program using another system diskette.

COPYDISK (DUAL DRIVE)

COPYDISK copies the contents of one floppy diskette onto another
floppy diskette. During the copying, the program checks for data
transfer errors.

Use COPYDISK to copy the TPC system diskette as well as
applications program diskettes. Store the original program
diskettes in a safe place as a back-up to the working copy.

THESE DIRECTIONS ARE FOR THE DUAL-DRIVE TPC ONLY!

NOTE!   Before you run COPYDISK, remember to format all new
          floppy diskettes before using them. If you do not know
          how to format a new diskette, refer to the utility
          program FORMAT.

USER: 1.   Make sure the system diskette is in drive A.

            2.   If the system prompt is not displayed, reset the
                  TPC by pressing the <Ctrl>/<Alt>/<Del> keys
                  simultaneously. If the system prompt is
                  displayed, go to Step 4.

SYSTEM: 3.   Displays

                  TeleVideo System TPC I Vx.x
                  (c) 1983 TeleVideo Systems, Inc.
                  xxK CP/M version y.y (double drive)

                  A>

USER: 4.   Enter

                  COPYDISK<Return>

SYSTEM: 5.   Displays

                  COPYDISK (TS 803/TS 803H/TPC I) Vx.x
                  (c) 1983 TeleVideo Systems, Inc.

                  Source Drive:(ESC to abort.)

USER: 6.   Enter

                  A

                  (Either drive can be the source from which the
                  copy is made.)

SYSTEM: 7.   Displays

                  Destination Drive:(ESC to abort.)

USER: 8.   Enter

                  B

                  (The diskette in this drive will contain the new
                  copy.)

SYSTEM: 9.   Displays

                  Confirmation:COPYDISK from "A" to "B".

                  Warning:COPYDISK will destroy all data
                  on destination disk with no chance of recovery.
                  Press RETURN when ready.(ESC to abort, BAR to re-start.)

USER: 10.   Remove the system diskette from drive A (if
                 you want to copy a different diskette than the
                 system diskette).

          11.   Insert the diskette to be copied into drive A
                 (source drive).

          12.   Insert the diskette you want to copy onto in drive
                 B (destination drive).

          13.   Check the labels of the inserted diskettes to make
                 sure they are in the correct drives.

                 NOTE!   If you entered incorrect drive names for
                                the source drive and destination drive,
                                press the space bar to re-enter the
                                correct drive.

          14.   Press

                 <Return>

SYSTEM: 15.   Displays

                 Copying(Track) --> 0123456789012345678901234567890123456789
                 Disk Copy Completed.

USER: 16.   Listen for a clicking sound as the program copies
                 the diskette.

SYSTEM: 17.   Displays

                 Source Drive:(ESC to abort.)

USER: 18.   a.   To copy more diskettes, go back to Step 6.

                 b.   To end the program, insert the system
                       diskette in drive A and press

                       <Esc>

SYSTEM: 19.   Displays

                 A>

COPYDISK Error Messages

If this message appears on the screen

    Data Transfer Error(s) on "Drive name"
    Verification error(s).

try one of the suggestions listed below.

1.   Run COPYDISK again, reversing the source and destination drive designations.

2.   Make sure that the diskette used as the destination is not defective or damaged; if necessary, use a new formatted diskette.

3.   Try copying another diskette.

4.   Run the program using another system diskette.

6.   PROGRAMMING THE TERMINAL EMULATOR

This chapter explains the procedures that a programmer must
follow to take advantage of the video attributes of the TPC I.
Those users with little technical background need not read this
information. They should carefully read the earlier chapters of
this manual, as well as any documentation that accompanies the
applications programs they want to run on the TPC I.

SUBSYSTEMS OVERVIEW

The TPC I video display can be described as comprising several
basic subsystems:

        CPU
        Terminal Emulator/Graphics Driver (EPROM)
        System RAM
        Graphics RAM

These subsystems operate together to control the video display
for either screen I/O operations or display graphics. The
sections of this chapter describe the various functions that are
available to the programmer to control the video display. Figure
6-1 illustrates the relationships of these subsystems to one
another.

Figure 6-1
Subsystems Communications
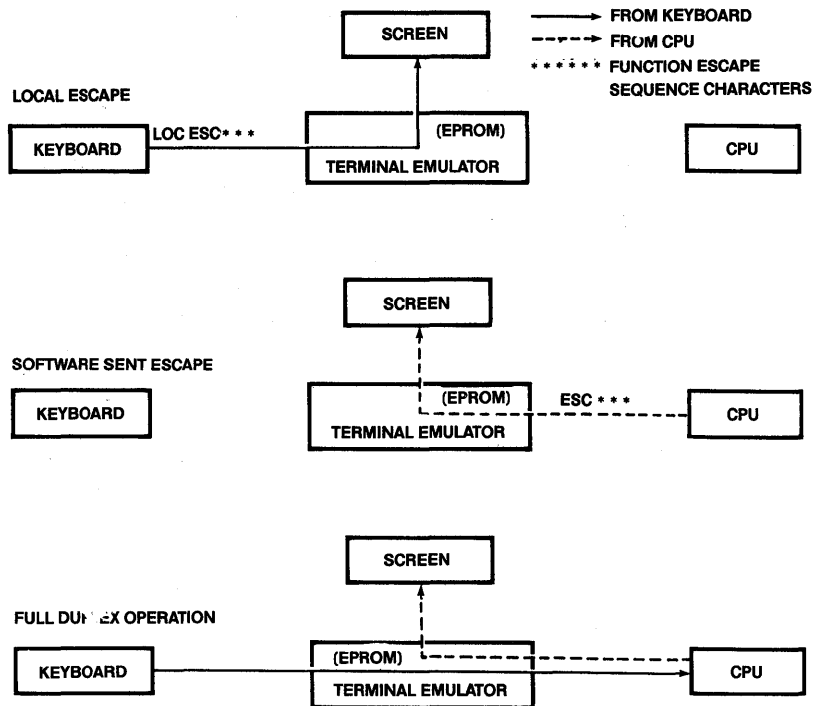
The CPU is a ZILOG Z80A operating at 4 MHz. The system has three
different kinds of memory: an 8K EPROM containing the system
boot, diagnostics, and display control; a 32K block of video
display memory which is divided into a 12K alphanumerics section
and a 20 Kbyte graphic memory section as a 640 x 240 pixel bit-
map or 80 x 24 line display; and a 64K block of system RAM.
System logic and BIOS enhancements control the memory bank
switching which is illustrated in Figure 6-2.

Figure 6-2
Memory Architecture

```
FFFF HEX ┌──────────────────────────┐        FFFF  ┌──────────────────────────┐
         │           BIOS           │              │           BIOS           │
CBASE + 1600 ─────────────────────────        CBASE + 1600 ─────────────────────────
         │           BDOS           │              │           BDOS           │
CBASE + 0800 ─────────────────────────        CBASE + 0800 ─────────────────────────
         │           CCP            │              │           CCP            │
CBASE ────────────────────────────────        CBASE ────────────────────────────────
         │      PORTION OF TPA      │              │                          │
C000 HEX ─────────────────────────────              │                          │
         │     ALPHANUMERIC RAM     │              │           TPA            │
B800 HEX ─────────────────────────────              │                          │
         │       GRAPHICS RAM       │              │                          │
4000 HEX ─────────────────────────────              │                          │
         │         NOT USED         │        0100  ─────────────────────────────
1FFF HEX ─────────────────────────────              │    SYSTEMS PARAMETERS    │
         │     GRAPHIC DRIVER       │        0000  └──────────────────────────┘
         │      ALPHA DRIVER        │
         │   DIAGNOSTICS EPROM      │
0000 HEX └──────────────────────────┘
                  PAGE 0                                     PAGE 1
```
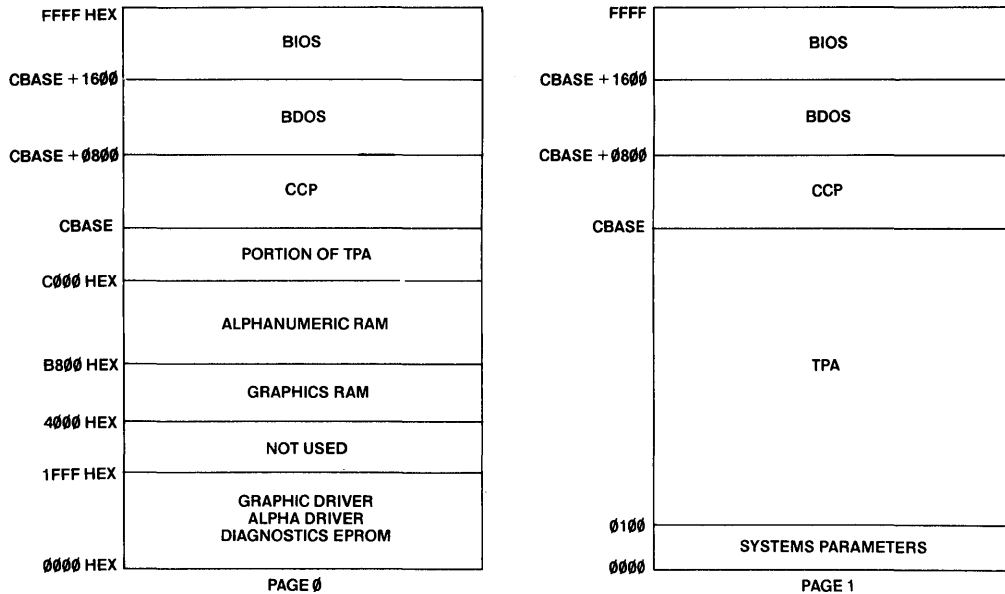
After the power is first turned on, or after a system reset, the
hardware automatically selects the memory configuration shown on
the left of Figure 6-2. This includes the system EPROM at
addresses 0000 - 1FFF, the graphics RAM block at 4000 - BFFF, and
the upper 16K of system RAM at C000 - FFFF.

The EPROM contains all necessary initialization logic for loading
the CP/M program. CP/M is loaded into the high RAM addresses
above 48K. When loading is complete, the control logic switches
to the full 64K RAM memory configuration and the software
initializes the lower RAM system scratch space.

The EPROM then serves as a video display graphics driver. When
special graphics functions, BIOS calls are made from applications
software, the graphics memory configuration (PAGE 0) is swapped
in and the CPU fetches instructions from the EPROM firmware.
The CPU then updates the graphics/alphanumerics RAM which it
shares with the video controller on alternate cycles.

TERMINAL ATTRIBUTES EMULATOR

The video display on your TPC I is controlled by firmware stored
in the 8K on-board EPROM. The firmware has two sections: the
Terminal Attributes Emulator, and a Graphics Primitives Driver.
Both are easily accessed by the programmer. The Terminal
Emulator controls all standard video terminal display functions,
such as alphanumerics, as well as special video attributes and
cursor addressing. For a description of function calls to the
graphics driver, see Chapter 7, Video Graphics.

Default Video Attributes

The video display of the TPC I can be set for black characters on a yellow background, by closing the eighth section of the DIP switch. Yellow letters on a black background can be set by opening the eighth section of the DIP switch (this is the default setting). Chapter 2 shows the location of the DIP switch and illustrates the procedure for changing a switch setting. See Table 2-2 and Appendix K for the switch settings.

The display intensity (brightness) can be adjusted with the contrast knob on the lower-left side of the screen on the front panel of the main unit.

Video Attributes

The video attributes of each character as well as the spaces (blanks) on the screen can be controlled to define the appearance of the screen. The following attributes are available.

Table 6-1
Video Attributes

Attribute	Description
Normal Video	Restores the background of the screen to that selected by rear switch settings.
Reverse Video	Changes the background of the screen to the reverse of that which appears at power on (default). If the screen is normally black with yellow characters, it will now be yellow with black characters.
Underline	Creates a solid line below all characters on the line (including the line created by the underscore key). This attribute starts with the cursor position and continues until another attribute is encountered.
Blink	Causes all characters to blink. This attribute starts with the cursor position and continues until another attribute is encountered.
Invisible	Causes all data entered on the line to be invisible to you although the cursor moves and is transmitted to the computer. (A typical application might be for entering passwords to access a program.) This attribute starts with the cursor position and continues until another attribute is encountered.

Half Intensity   Changes the intensity to half of normal on a
                        character-by-character basis.

                   NOTE!   Half intensity differs from other
                                  visual attributes in two ways:

                            1.   Once it is set, it affects all
                                  characters entered, regardless
                                  of cursor position, until it
                                  is turned off.

                            2.   This attribute character never
                                  occupies a character space.

Setting Video Attributes

1. Your program should place the cursor one position **before** you
    want the attribute to start.

NOTE!   Remember that each attribute occupies a character
                position. If your program types over the attribute, it
                is lost.

2. Your program should send the appropriate escape sequence as
    listed in Table 6-2.

NOTE!   Escape sequences are shown with spaces. These spaces
                are included only for the sake of clarity and are not
                to be included in the sequence.

Escape Sequence Calls to the Terminal Emulator

The Terminal Emulator is a gate array processor that intercepts
escape sequences when they are sent from the CPU to the console
device and sets the appropriate terminal attribute or function.
These escape sequences are ASCII characters sent in a sequential
array just as in normal terminal communications.

The typical escape sequence is indicated by ASCII character 27
(decimal value) ESC. Some of the cursor control functions are
represented by ASCII control characters. For example, the **cursor
up** function is ^K, ASCII character 11 (decimal).

You can send a **local escape** to the Terminal Emulator directly
from the keyboard (by-passing the CPU) by pressing the <Shift> and
<Esc>,(<Shift>/<Esc>), keys simultaneously. This allows you to
experiment with the terminal attributes without a code-compile-
run overhead problem.

Table 6-2
Escape Sequences for Video Attributes

Attribute	Escape Sequence
Normal (default) video	ESC G 0
Invisible normal video	ESC G 1
Blink	ESC G 2
Invisible blink	ESC G 3
Reverse video (reverse of default)	ESC G 4
Invisible reverse	ESC G 5
Reverse and blink	ESC G 6
Invisible reverse and blink	ESC G 7
Underline	ESC G 8
Invisible underline	ESC G 9
Underline and blink	ESC G :
Invisible underline and blink	ESC G ;
Reverse and underline	ESC G <
Invisible reverse and underline	ESC G =
Reverse and underline and blink	ESC G >
Invisible reverse and underline and blink	ESC G ?
Half intensity on	ESC)
Half intensity off	ESC (

Escape Sequences for Clearing the Screen

The clear function is used to clear data from screen memory.
Clear commands are summarized in Table 6-3.

Table 6-3
Clear Commands Escape Sequences

Clear Command	Escape Sequence	Effect
Clear All to Nulls	ESC :	Clears all data on the page to nulls or to half-intensity nulls if half intensity is on.
Clear All to Half-Intensity Spaces	ESC ,	Clears all data on the page to half-intensity spaces.
Clear All to Spaces	ESC + or ^Z	Clears all data on the page to spaces or to half-intensity spaces if half intensity is on.
Clear All to Nulls and Reset Half Intensity	ESC *	Clears all data on the page to nulls. Resets half intensity.

Escape Sequences for Cursor Display

The cursor display may appear any one of five ways. To change the
appearance of the cursor, enter one of the following escape
sequences in Table 6-4.

Table 6-4
Cursor Display Escape Sequences

Appearance	Command
Cursor not displayed	ESC . 0
Blinking block cursor	ESC . 1
Steady block cursor	ESC . 2
Blinking underline cursor	ESC . 3
Steady underline cursor	ESC . 4

Cursor Control

The cursor can be directed to a location on the screen by using
the cursor control commands in Table 6-5, or by sending the cursor
to a screen address as described in the section on Cursor
Addressing in this chapter.

Table 6-5
Cursor Control Commands

Cursor Control	Control Code	Effect
Up	^K	Moves the cursor up one line until it encounters the top of the screen. Once it reaches the top of the screen, receipt of further codes has no effect.
Down	^V	Moves the cursor down one line. If the cursor is on the bottom line of the screen, the code has no effect.
Left	^H	Functions the same as BACKSPACE. Moves the cursor to the left. If the cursor is currently in the first column of the line, it moves to the last column of the preceding line. If the cursor is on home, it has no effect.

Right   ^L   Moves the cursor right one column. If
                                           the cursor is at column 80, the cursor
                                           moves to the first column of the next
                                           line. If the cursor is at the last
                                           column of the last line, it causes the
                                           screen to scroll up and moves the cursor
                                           to the first column of the new line.

Home   ^^   Moves the cursor to column one of the
                    (<Ctrl>/^)   first line. If the cursor is already at
                                           the home position, the code has no
                                           effect.

Carriage   ^M   Moves the cursor left to column one of
Return   the current line.

To move the cursor up one line, send

        ESC j

A reverse linefeed moves the cursor up one line for each reverse
linefeed code received. The screen scrolls down one line when
the cursor reaches the top line of the screen. As the display
scrolls down one line, a new line of data appears at line one of
the screen and the last line of the display is deleted. The data
consists of spaces.

Cursor Addressing

The computer can also position the cursor to a specific location
without repetitive cursor movement commands. This is called
loading or addressing the cursor.

To address the cursor, enter

        ESC = r c

where

        r is the desired row (line). Refer to Table 6-6 to find the
        ASCII code representing the desired row.

        c is the desired column. Refer to Table 6-6 to find the
        ASCII code representing the desired column.

NOTE!   If your applications program inserts nulls between
             characters, loading the cursor does not function as
             described. Instead, the cursor goes to an
             unpredictable position.

For example, if you want the cursor to go to Row 9 of Column 50, send

        ESC = (Q

Table 6-6
Cursor Coordinates

Row*/ Column	ASCII Code Transmitted	Column	ASCII Code Transmitted	Column	ASCII Code Transmitted
1	Space	33	@	65	`
2	!	34	A	66	a
3	"	35	B	67	b
4	#	36	C	68	c
5	$	37	D	69	d
6	%	38	E	70	e
7	&	39	F	71	f
8	'	40	G	72	g
9	(	41	H	73	h
10	)	42	I	74	i
11	*	43	J	75	j
12	+	44	K	76	k
13	,	45	L	77	l
14	-	46	M	78	m
15	.	47	N	79	n
16	/	48	O	80	o
17	0	49	P		
18	1	50	Q		
19	2	51	R		
20	3	52	S		
21	4	53	T		
22	5	54	U		
23	6	55	V		
24	7	56	W		
25	8	57	X		
26	9	58	Y		
27	:	59	Z		
28	;	60	[		
29	<	61	\		
30	=	62	]		
31	>	63	^		
32	?	64	-		

*Row (line) value may not exceed 24.

Tabs

The cursor may be moved on the screen to preset typewriter-style
tabs. Tab controls are summarized in Table 6-7.

Table 6-7
Tab Controls

Action	Control Code	Effect
Set tab*	ESC 1	Sets a typewriter-style column tab.
Tab	^I	Causes the cursor to advance to the next tab set. If no tabs are set, the code has no effect and the cursor does not move.
Back Tab	ESC I	Causes the cursor to go back to the previous tab position set. If no tabs are set or if the cursor is on the first tab position on the line, this code moves the cursor to the first column on the line.
Clear Tab**	ESC 2	Clears the tab where the cursor is located when this code is entered.
Clear All Tabs***	ESC 3	Clears all tabs regardless of the position of the cursor when the code is entered.

---

   *To set a tab, move the cursor to the column position where
    you want a tab. Be sure you enter a **numeral** one, not a
    lower-case **L.**
  **Position the cursor at the tab to be cleared before
    entering the sequence.
 ***The position of the cursor when this code is entered is
    not important.

Text Editing Functions

Changing text can involve the following three actions:

    Replacing (typing over) existing text; referred to here as
    editing

    Inserting new text which pushes existing text to the right
    from the cursor position

    Deleting existing text (by character or line) by moving text
    backward toward the cursor

Editing, inserting, and deleting can occur within the line on

which the cursor is positioned. Text which reaches the beginning
or the end of a line by these actions are, if moved **further,**
lost (**fall off** that line).

Table 6-8 summarizes the effects of the available editing
commands.

Table 6-8
Editing Commands

Edit Command	Escape Sequence	Effect
Character Insert	ESC Q	Causes character at the cursor to move right one column position and enters a space at the cursor position. As characters are inserted, characters reaching column 80 are lost. If half intensity is on, half-intensity spaces replace the erased characters.
Character Delete	ESC W	Deletes the character at the cursor position and moves all following characters left one position. At the end of the delete function, a space is written into the last position on the line. If half intensity is on, half-intensity spaces replace the erased characters.
Line Insert	ESC E	Inserts a line consisting of spaces at the cursor position. This causes the cursor to move to the start of the new line and all following lines to move down one line, resulting in the loss of the last line on the page. If half intensity is on, half-intensity spaces replace the erased characters.
Line Delete	ESC R	Deletes the line at the cursor position and all following lines move up one line. The cursor moves to column one of the line and spaces are loaded into the last line of the page. If half intensity is on, half-intensity spaces replace the erased characters.
Erase to End of Line	ESC T	Erases all characters from the cursor to the end of the line and replaces them with spaces. If half intensity is on, half-intensity spaces replace the erased characters.

Erase to End of Line with Nulls	ESC t	Erases all characters from the cursor position to the end of the line and replaces them with null characters. If half intensity is on, half-intensity nulls replace the erased characters.
Erase to End of Page	ESC Y	Replaces characters from the cursor position to the end of the screen with spaces. If half intensity is on, erased characters are replaced with half-intensity spaces.
Erase to End of Page with Nulls	ESC y	Erases all characters from the cursor position to the end of the page and replaces them with null characters. If half intensity is on, half-intensity nulls replace the erased characters.

Monitor Mode

You can display all characters (including all escape and control sequences) sent from the keyboard or generated by a program. This feature is called **monitor mode.** Having the escape and control commands on the screen provides visual clues to make program debugging easier.

To enable monitor mode, send:

ESC U

To terminate the display of the control commands, send either

ESC u or ESC X

Table 6-9 lists the monitor mode control characters and shows the monitor mode control characters as they appear.

Table 6-9
Monitor Mode Control Characters

Code	ASCII	Hex	Character Displayed
^@	NULL	00	none
^A	SOH	01	S H
^B	STX	02	S X
^C	ETX	03	E X

^D	EOT	04	E T
^E	ENQ	05	E Q
^F	ACK	06	A K
^G	BEL	07	B L
^H	BS	08	B S
^I	HT	09	H T
^J	LF	0A	L F
^K	VT	0B	V T
^L	FF	0C	F F
^M	CR	0D	C R
^N	SO	0E	S O
^O	SI	0F	S I
^P	DLE	10	D L
^Q	DC1	11	D 1
^R	DC2	12	D 2
^S	DC3	13	D 3
^T	DC4	14	D 4
^U	NAK	15	N K

^V	SYN	16	S Y
^W	ETB	17	E B
^X	CAN	18	C N
^Y	EM	19	E M
^Z	SUB	1A	S B
^[	ESC	1B	E C
^\	FS	1C	F S
^]	GS	1D	G S
^^	RS	1E	R S
^_	US	1F	U S
DEL	DEL	7F	

7.   VIDEO GRAPHICS

This chapter discusses how to access the graphics features, what graphics features are available, and how to access the SuperMouse support functions.

The functions described in this chapter are specific to the TPC I. We recommend that you use the GSX-80 driver instead of these functions so that you can take advantage of the GSX-80 drivers for other devices that are supplied on the system diskette. Your programs will also be more portable. You can order a GSX-80 programmer's manual from TeleVideo through your computer store.

The GSX-80 driver for the TPC I calls the functions described in this chapter, but provides much more functionality than is available in the firmware.

INTRODUCTION TO GSX

This section provides you with information about how a graphics system is built when using GSX. It is intended for those who want to familiarize themselves with the basic capabilities of GSX. For more specific information about how to fit the graphics capabilities into your system, you should consult the GSX User's Guide.

GSX (Graphics System EXtension)

GSX-80 is the Graphic System eXtension for the CP/M family of operating systems. It incorporates graphics capability into the operating system and provides a host-independent and device-independent interface to your applications programs. Graphics primitives are provided for implementing graphics applications with reduced programming efforts. In addition, GSX-80 offers program portability by allowing an applications program to run on any CP/M system with the GSX-80 option. GSX-80 also promotes programmer interface to graphics which is compatible with one of the world's most widely used operating systems, CP/M.

GSX-80 is implemented as an integral part of your operating
system. Applications programs interface to GSX-80 through a
standard calling sequence similar to the BDOS conventions.
Drivers for specific graphics devices translate the standard GSX-
80 calls to the unique characteristics of the device. In this
way, GSX-80 provides device-independence since the peculiarities
of the graphics device are not visible to the applications
program. GSX-80 consists of several parts that work together to
give your system graphics capability:

*   The Graphics Device Operating System (GDOS)

*   The Graphic Input/Output System (GIOS)

*   The Gengraf Utility

Graphic Device Operating System

The Graphic Device Operating System(GDOS) contains the basic host
and device independent graphics functions that can be called by
your application program. GDOS provides a standard interface to
graphics which is constant regardless of specific devices or host
hardware, just as the BDOS standardizes disk interfaces. Your
applications program accesses GDOS through a mechanism analogous
to the normal BDOS systems calls.

GDOS loads at runtime with your graphics applications program,
so it consumes system memory space only when required, leaving
the normal Transient Program Area for non-graphic programs.

GDOS performs coordinate scaling so that your program can specify
points in a normalized coordinate space. It uses device specific
information to translate the normalized coordinates into the
corresponding values for your particular graphics device.

Multiple graphics devices can be supported under GSX-80 within a
single application. By referring to devices with a workstation
indentification number, graphics information can be sent to any
of the several resident devices. GDOS dynamically loads a
specific device driver when requested by the applications
program, overlaying the previous driver. This technique
minimizes memory size requirements since only one driver is
resident at any time.

Graphic Input/Output System

The Graphic Input/Output System(GIOS) is similar to the basic I/O
system or BIOS. It provides the device specific code required to
interface your particular graphics device to GDOS.

GIOS consists of a set of device drivers that communicate directly with the graphic devices through the appropriate host ports. A unique device driver is required for each different graphics device on your system. The term GIOS refers to the collection of available device drivers as well as the particular driver that is loaded into memory when required by your application. Although a single program can use several graphic devices, only one driver is loaded by GDOS at a time.

GIOS performs the graphics primitives of GSX-80 consistent with the inherent capabilities of your graphics device. In some cases a device driver will emulate standard GDOS capabilities which are not provided by the graphics device hardware. For example, some devices may require that dashed lines be simulated by a series of short vectors generated in the device driver.

GSX-80 is supplied with drivers for many of the most popular graphics devices for microcomputer systems.

A list of devices supported by the GSX supplied by TeleVideo for the TPC I, TS 803, TS 803H, TS 1603, and TS 1602 is supplied in this chapter. Some manufacturers of graphics devices supply drivers for their devices.

GENGRAF Utility

The GENGRAF utility is used to combine your applications program and the GSX loader into one executable .COM file. The GSX loader is a small program that loads the GDOS and GIOS into memory at run time and establishes the links between your applications program and GDOS. The GSX loader is attached to your applications program after it has been compiled/assembled and linked with the required external routines and libraries.

How GSX Fits Into Your System

Your applications program may be written in any language provided the GDOS protocol is observed. You may compile/assemble and link your application in the normal manner, yielding a .COM executable file. One additional step must be performed, however, before executing your graphics program: the GSX Loader must be attached to the front of your program so that it can prepare the operating system environment for your graphics application.

The GENGRAF utility (provided with the GSX-80 distribution) allows you to attach the loader to your program with one simple command:

GENGRAF<BAR><filename><Return>

For example, if your graphics application program were in an executable file named MYFILE.COM, then the following command string would attach the GSX Loader and place the result into file MYFILE.COM.

GENGRAF<BAR>MYFILE<Return>

The resulting MYFILE.COM file would be ready to run.

You should be aware of the total memory space available to your applications program in the TPA. This will be less for graphics applications than for normal programs because of the GDOS and device driver requirements.

GSX Supported Peripherals

TeleVideo supplies the following device drivers for their systems with graphic capabilities, like the TPC I, TS 803, TS 803H, TS 1602 and TS 1603:

GSX-80 supports:

*   Printers
            Epson MX-80 with Graftrax Plus
            Epson MX-80 type III
            Epson FX-80
            Epson-100

*   Plotters
            Hewlett-Packard 7470
            Hewlett-Packard 7220
            Houston Instruments HiPlot 7M
            Houston Instruments HiPlot 3M

*   CRT device drivers
            TPC I
            TS 803
            TS 803H

GSX-86 supports:

*   Printers
            Epson MX-80 with graftrax Plus
            Epson MX-80 type III
            Epson FX-80
            IDS MicroPrism 480
            IDS Prism 80/132 (monochrome)
            IDS Prism 80/132 (color)
            Okidata Microline 92
            Printronix MPV
            Printronix P300/P600

*   Plotters
            Hewlett-Packard 7470A
            Hewlett-Packard 7220
            Houston Instruments HiPlot DMP-3/4-443
            Houston Instruments HiPlot DMP-6/7
            Strobe Model 100

*   CRT device drivers
            TS 1602
            TS 1603 (plus graphic option)

GRAPHICS PRIMITIVES DRIVER

The EPROM firmware in the TPC I serves three functions:

        System boot and diagnostic routines
        Alphanumerics Generator/Terminal Attributes Emulator
        Graphics Primitives Driver

The Graphics Primitives Driver is the lowest level interface to the graphics memory. The Graphics Driver accesses the 20K segment of the 32K graphic memory block as a 640 x 240 pixel bit map through the CPU, mapping the graphics to be displayed by the video controller. The CPU updates the display on alternate cycles with video controller access.

The programmer can have programs access the display memory with calls to the Graphics Primitives Driver. The following sections describe the procedures by which calls can be made to the graphics driver.

Figure 7-1
Graphics Display

```
       (Y)
        ^
        |
        |---------------------  (639,  239)
        |                    |
        |      Screen        |
        |                    |
        |                    |
 (0,  0)  -------------------------->(X)
```

           X, Y coordinates in unit of pixel
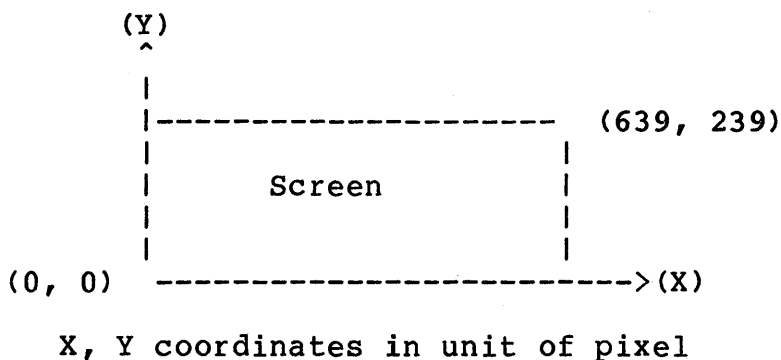
The TPC I is also equipped with an optical SuperMouse interface. The SuperMouse has eight software-support functions which are described in the SuperMouse Support Functions section of this chapter. These functions can be directly accessed by the user through a single-entry point. There are also six graphic hardware cursor functions.

CALLING THE GRAPHICS DRIVER

The following procedure enables you to access the graphics driver.

1.   Load register "C" with the desired graphics function identifier as listed in Table 7-1.

2.   Load the parameters as specified by Table 7-1 either into registers HL, and DE or into a **parameter array.**

NOTE!   The address of the parameter array mentioned above is contained in address OFF00h and OFF01h.

            OFF00h contains low byte of address.
            OFF01h contains high byte of address.

3.   Perform a restart instruction (RST 28h in Z80 or RST 5 in
      8080). The restart instructions direct the call to a
      location in the BIOS. It then moves to the graphics entry
      point in the EPROM where the graphics function is carried
      out.

Notice that the array for an extended parameter list is stored in
high RAM memory so that it is still available after the memory
bank switches from graphics mode to alpha mode and vice versa.

The following is an example of how to access the graphics driver
in Z80 code. A complete list of examples of how to access the
graphics driver is listed in the section of this chapter called
Sample Graphics Access Program Using 8080 Assembly Code.

Suggestion: it is very useful to create a macro for access to the
graphics output primitives.

```
grout      macro
           RST      28h        ;access graphics
           endm
```

This macro can be called each time the registers are set
appropriately for a particular graphics function call.

Example 1: To draw a line in Z80 code.

```
           CSEG
           .Z80
           ld       C,1        ;"initialize to graphics mode"
           grout               ;
           ld       C,2        ;clear graphic display
           grout
           ld       C, 11
           ld       HL, 1      ;set line style to solid.
           grout
           ld       C, 12
           ld       HL, 1      ;set line color to 1.
           grout
           ld       C, 14
           ld       HL, 1      ;set writing mode to replace.
           grout
           ld       C,4        ;"move to" 0,0
           ld       HL,0       ;
           ld       DE,0       ;
           grout               ;
           ld       C,5        ;"line to" 639,239
           ld       HL,639     ;
           ld       DE,239     ;
           grout               ;
           END
```

The preceding program draws a diagonal line across the bit map from
pixel 0,0 to pixel 639,239.

Example 2: To draw a graphics character string in Z80 code.

```
                CSEG
                .Z80
                ld          C,1
                grout
                ld          C,2             ; clear graphic display
                grout
                ld          C,13
                ld          HL,1            ; set character color to 1.
                grout
                ld          C,14
                ld          HL,1            ; set writing mode to replace.
                grout
                ld          HL,(0FF00h)     ; get pointer to string array
                ld          (HL),5          ; put length of string into array
                INC         HL              ;
                INC         HL              ; point at string
                ld          DE,string       ;
                ld          b,5             ;
loop:           ld          a,(DE)          ;
                ld          (HL),a          ; load characters onto parameter array.
                INC         DE
                INC         HL
                INC         HL              ; load one character per word.
                DJNZ        loop            ;
                ld          C,6             ; "draw string"
                grout                       ;

string:         Db          'Hello'
                END
```

See Figure 7-2 for the parameter array whose address is contained in FF00h and FF01h.

Figure 7-2
Parameter Array Contents

```
 _____
| Character count #     |   = byte 1
|-----------------------|
| Not Used              |   = byte 2
|-----------------------|
| Character #1 (H)       |   = byte 3
|-----------------------|
| Not Used              |   = byte 4
|-----------------------|
| Character #2 (e)       |   = byte 5
|-----------------------|
| Not used              |   = byte 6
|-----------------------|
| Character #3 (l)       |   = byte 7
|-----------------------|
| Not Used              |   = byte 8
|-----------------------|
| Character #4 (l)       |   = byte 9
|-----------------------|
| Not Used              |   = byte 10
|-----------------------|
| Character #5 (o)       |   = byte 11
|_____|
```

GRAPHICS DRIVER FUNCTIONS

The TPC I Graphics Primitives Driver supports the following
functions. They are listed by function identification number
(passed in register C) in Table 7-1.

Table 7-1
Graphics Driver Functions

Function Number	Parameters	Description
0	C = 0	**Initialize to alphanumeric mode.** Does not clear display.
1	C = 1	**Initialize to graphics mode.** Does not clear display.
2	C = 2	**Clear** graphics screen (memory).
3	C = 3 array	**Polyline.** Draws a connected sequence of lines in the current line style, line color, and writing mode. The first point is taken as the current position. The last end-point of the sequence becomes the current position (CP).

The parameters are passed in an array.
The address of the array is contained in
locations 0FF00h and 0FF01h.
Maximum number of vertices (x,y) = 63
Parameter array contents:

      word 0 = count (n vertex points)

      word 1 = x1   ;move to vertex (x1, y1)
      word 2 = y1

      word 3 = x2   ;line to vertex (x2, y2)
      word 4 = y2

      word 5 = x3   ;line to vertex (x3, y3)
      word 6 = y3   ;

      (etc.)

4	C = 4 HL = x DE = y	**MoveTo.** Updates the current position. Used only with LineTo and DrawString.
5	C = 5 HL = x DE = y	**LineTo.** Draws a line from current position to x,y then updates the current position. The current linestyle, line color, and writing mode are used.
6	C = 6 (array)	**DrawString.** Draws character string starting at current position. It does not update the current position when done. The string is stored in an array whose address is contained at memory locations 0FF00h and 0FF01h. Character count max = 80.

    NOTE!   Address of parameter array is
                 contained in 0FF00h, 0FF01h.

Figure 7-3
DrawString Parameter Array Contents

```
 _____
| Word 1 = Character count |
|                                 |
| Word 2 = Character 1            |
|                                 |
| Word 3 = Character 2            |
|                                 |
|          "                      |
|                                 |
|          "                      |
|                                 |
| Word n = Character n            |
|_____|
```

NOTE!   A graphics character is formed
            by an 8-bit by 8-bit character
            cell. The graphics character
            string starts at the character
            cell's baseline at the current
            (X,Y) position. The current
            position can be changed by the
            MOVETO command.

Figure 7-4
8x8 Graphics Character Cell

BASE LINE
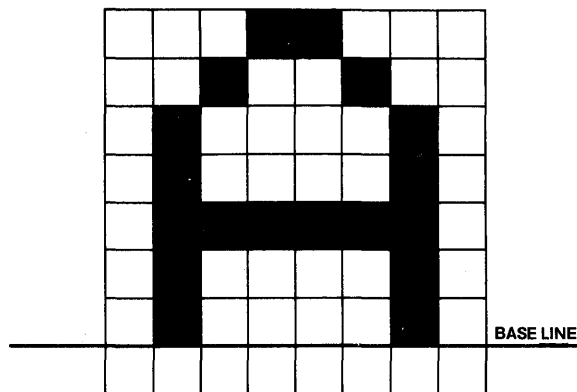
7   **Scan Out Horizontal Line.** This routine
           scans out a horizontal line with the
           current fill interior style, fill style
           index, and fill color. The end points
           of this horizontal scan line are passed
           in the array as shown in Figure 7-5.

Figure 7-5
Scan Out Parameter Array Contents

```
---------------------
|  Y - Coordinate  |  word 1
|------------------|
| X1 - Coordinate  |  word 2
|------------------|
| X2 - Coordinate  |  word 3
|_____|
```

NOTE!   Address of array is contained
                in 0FF00h and 0FF01h.

This routine internally adjusts the area
style (i.e. pattern or crosshatch)
according to the starting coordinate
(X,Y) passed to it from the user. This
routine is particularly useful for
scanning out the horizontal sections of
a polygon or bar fill.

NOTE!   X2 must be greater than or
                equal to X1.

8   C = 8   **Crosshair.** This function can be used
           HL = 0 or 1   to draw a cross-hair cursor of size
                         specified by the user. This function
                         overrides the current line style, line
                         color and writing mode with a solid line
                         style, white line color, and XOR writing
                         mode. The current attributes are
                         restored when exiting this function.

           Reg HL = 0   Use data passed in
                            parameter array.

               HL = 1   Use data from last entry
                            to this function (used
                            for erasing).

           NOTE!   The crosshair is drawn using
                           XOR writing mode, but the
                           current writing mode remains
                           the same.

The Parameters needed are passed in the **array** as follows:

Figure 7-6
Crosshair Parameter Array Contents

```
---------------------------------------------------------------
|Y Value of horizontal line of cross-hair             | = Word 1
|-----------------------------------------------------|
|X1 Value of left endpoint of horizontal line         | = Word 2
|-----------------------------------------------------|
|X2 Value of right endpoint of horizontal line        | = Word 3
|-----------------------------------------------------|
|X  Value of vertical line of cross-hair              | = Word 4
|-----------------------------------------------------|
|Y1 Value of bottom endpoint of vertical line         | = Word 5
|-----------------------------------------------------|
|Y2 Value of top endpoint of vertical line            | = Word 6
---------------------------------------------------------------
```

                    Crosshair Cursor

```
                    (X,Y2)
                       |
                       |
                       |
                       |
      (X1,Y)-----------------------------(X2,Y)
                       |
                       |
                       |
                       |
                    (X,Y1)
```
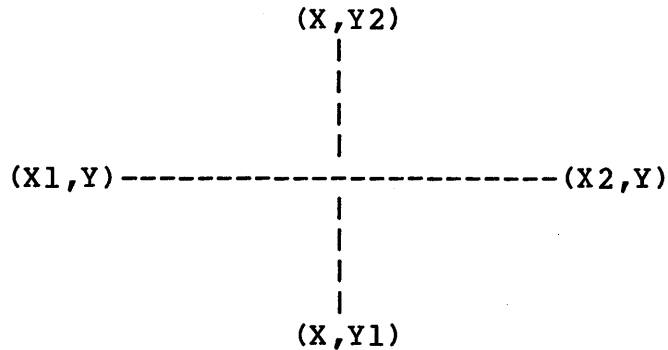
9   C = 9   This function fills a **BAR** with the
                           currently selected fill attributes:
                           fill interior style, fill-style index,
                           and the fill-color index. The lower-
                           left and upper-right corner coordinates
                           are passed in the parameter array as
                           in Figure 7-7.

Figure 7-7
Parameter Array Contents

```
------------------
| X1 - Coordinate  | Word 1
------------------
| Y1 - Coordinate  | Word 2
------------------
| X2 - Coordinate  | Word 3
------------------
| Y2 - Coordinate  | Word 4
------------------
```

```
                                    (X2,Y2)
         ------------------
         |                |
         |      BAR       |
         |                |
         ------------------
(X1,Y1)
```
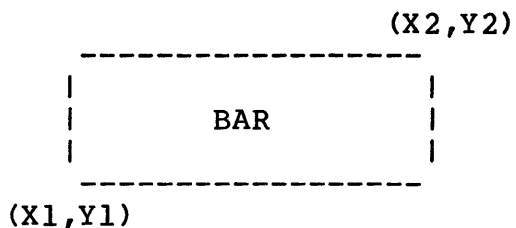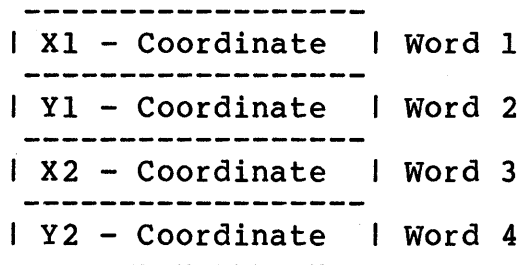
10	RESERVED	RESERVED
11	C = 11 HL = style	**Set LineStyle.** Valid parameters are 1 through 8:

> 1 - solid
> 2 - dashed
> 3 - dotted
> 4 - dash-dot
> 5 - long-dash
> 6 - short-dash
> 7 - dot-dot-dash
> 8 - long-dot

| 12 | C = 12
HL = color | **Set LineColor.** Valid parameters are 0 or 1: |

> 1 - white
> 0- black

| 13 | C = 13
HL = color | **Set Character Color.** Sets the current character color. Valid parameters are 0 or 1: |

> 1 - white
> 0- black

14   C = 14   **Set Write Mode.** Sets the current
                                writing mode for all output primitives.
                                Valid parameters are 1 through 8.

                                1 - Replace
                                2 - Logical <OR>
                                3 - <XOR>
                                4 - <NOT>
                                5 - Complement then Replace
                                6 - Complement then logical <OR>
                                7 - Complement then <XOR>
                                8 - Complement then <NOT>

                                Writing Mode Logic

                                1 - Plane = (Color Index) AND (Pattern)

                                2 - Plane = (Plane) OR [(Color Index)
                                    AND (Pattern)]

                                3 - Plane = (Plane) XOR [(Color Index)
                                    AND (Pattern)]

                                4 - Plane = (Plane) AND [(Color Index)
                                    AND (Pattern)]

                                5 - Plane = (Color Index) AND $\overline{(Pattern)}$

                                6 - Plane = (Plane) OR [(Color Index)

                                    AND $\overline{(Pattern)}$]

                                7 - Plane = (Plane) XOR [(Color Index)

                                    AND $\overline{(Pattern)}$]

                                8 - Plane = (Plane) AND [(Color Index)

                                    AND $\overline{(Pattern)}$]

                                NOTE!   Plane refers to the Graphics
                                               Display. The Color Index and
                                               Pattern corresponds to the
                                               respective output primitive
                                               selected such as text, line,
                                               or fill.

15   C = 15   **Fill Interior Style.** This routine
sets the type of interior style to be
used in the horizontal scan routine.
The style selected is passed on the
first word of the parameter array:

```
          ------------
array     | Style  # |  word 1
          ------------
```

Table 7-2
Fill Interior Styles

Style #	Style
0	Hollow
1	Solid
2	Pattern
3	Cross-hatch

16   C = 16   **Fill-Style Index.** This routine sets the
style index of the currently-selected
interior style. The style index only
affects the pattern and cross-hatch
interior styles. The style index is
passed on the first word of the
parameter array.

```
          --------------
array     | Style Index |  word 1
          --------------
```

Table 7-3
Graphics Styles

Current Interior Style	Style Index	Comment
Hollow	None	Style Index has no effect.
Solid	None	Style Index has no effect.
Pattern	0 1 . . . 7	Low Intensity High Intensity

Cross-Hatch   0   Vertical
                   1   Horizontal
                   2   Diagonal 45 degrees
                   3   Diagonal -45 degrees
                   4   Vertical/Horizontal
                   5   Vertical/Diagonal 45 degrees
                   6   Vertical/Diagonal -45 degrees
                   7   Cross Diagonals

17   C = 17   **Fill Color Index.** This routine affects
                              the area style to be written into memory
                              in the following way:

                              1 = white
                              0 = black

                                      ----
                              array | Color Index | Word 1
                                      ----

18   RESERVED   RESERVED

19   C = 19   **GetPix.** Returns all of a scan line
          HL = y scan   of pixels. Returns the scan line in the
               line   array. Bytes in the array are packed
               value   with the most significant bit
               (0-239)   corresponding to the lower x value. The
                              bytes are loaded into the array in order
                              of increasing x (from left to right) on
                              the scan line.

SUPERMOUSE

The TPC I provides eight SuperMouse support functions which you
can access. The SETUPTPC program allows you to attach or detach
the SuperMouse from the system. Once the SuperMouse is attached,
the SuperMouse must be initialized (function 1) before support
functions (2-7) (see Table 7-4) become operational. The
SuperMouse opens a completely new dimension in human interface to
the computer. There are also six graphic hardware cursor support
functions which are discussed in the section, Mouse Support
Functions.

Figure 7-8
SuperMouse

RIGHT
KEY

MIDDLE
KEY

LEFT
KEY
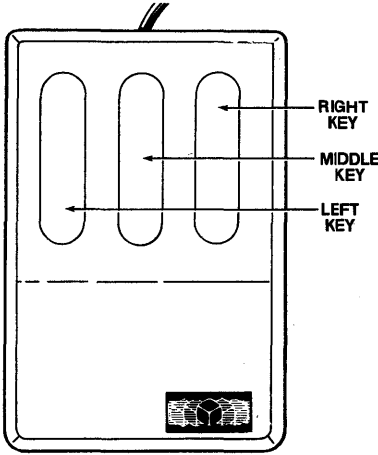
The SuperMouse is attached to the system through the SETUPTPC program.

SUPERMOUSE SUPPORT FUNCTIONS

You can access the SuperMouse functions by following these instructions:

1.   Load register "C" with the function number identifier (see Table 7-4).

2.   Input/Output parameters are passed in registers HL, DE, and A, as listed in the SuperMouse functions below.

3.   Perform a RESTART instruction.

>   RST 6   - for the 8080 instruction set.
>   RST 30h - for the Z80 instruction set.

The eight SuperMouse support functions and six hardware graphics cursor support functions that you may access are listed in Table 7-4.

Table 7-4
SuperMouse Functions

Function
Number   Function

C = 0   **INQUIRE.** Inquire if SuperMouse is available.
                        Inquire if hardware cursor available.

            USER: Enters nothing
            SYSTEM: Returns
                        Reg. A - 0   SuperMouse not available.
                               - 0FFh SuperMouse available.

            This function returns to the user a flag (Register A) which indicates if a SuperMouse is available or not. This flag is initially set by attaching or detaching the SuperMouse through the SETUPTPC program.

C = 1   **INITIALIZE.** This is to initialize the SuperMouse.

    USER: Enters

        Reg. - HL   Initial X-coordinate position
                      of SuperMouse (0 to 639).

        Reg. - DE   Initial Y-coordinate position
                      of SuperMouse (0 to 239).

    SYSTEM: Returns nothing

This function enables the SuperMouse interrupt, and sets the initial position of the SuperMouse.

NOTE!   When the SuperMouse is initialized, the default ASCII characters for the SuperMouse keys are implemented, and the SuperMouse key mode is set at 0.

             Mode 0 means that the pressing of the SuperMouse key is treated as a keyboard entry and that the SuperMouse stops tracking its movement until the key is released.

             The ASCII character associated with the SuperMouse key can be altered through SuperMouse function 6.

             The character placed in the keyboard queue because of the SuperMouse key depression can be retrieved through a BDOS call, such as BDOS function 6.

             Use a CP/M BDOS CALL to access the characters placed in the keyboard queue.

             For example: BDOS CALL FUNCTION 6

```
ld      c, 6        ;direct console
                     Input/Output
ld      e, 0ffh     ;set input flag.
call    5           ;BDOS CALL
                    ;BDOS returns character
                     in Register A or loads
                     Register A with 0 if no
                     character is available.
```

             The current status of the SuperMouse keys (up/down) can be inquired through SuperMouse function 3.

The default ASCII characters are as follows:

SuperMouse Key(left)   - ASCII character 32 (space)
SuperMouse Key(middle) - ASCII character 33 (!)
SuperMouse Key(right)   - ASCII character 34 (")

C = 2   **RANGE.** Inquires for the SuperMouse's coordinate range.

USER: Enters nothing
SYSTEM: Returns

Reg. - HL   SuperMouse's X-coordinate range
                 (640).

Reg. - DE   SuperMouse's Y-coordinate range
                 (240).

This function returns to the user the (X,Y) coordinate
range of the SuperMouse.

C = 3   **INPUT.** This returns the SuperMouse coordinates to the user.

USER: Enters nothing
SYSTEM: Returns

Reg.   - HL   SuperMouse's current X-coordinate
                  position.

Reg.   - DE   SuperMouse's current Y-coordinate
                  position.

Reg. A   - 0   No change in SuperMouse coordinate
                  position since last input or
                  no SuperMouse key has been pressed.

           - 0FFh New SuperMouse coordinate position
                  or SuperMouse key has been pressed
                  since last input.

Reg. B   SuperMouse Key Status. Contains
                  SuperMouse key status in bits 2, 1,
                  and 0 for the left, middle,
                  and right SuperMouse keys
                  respectively.

                  If bit returned is:

                  1 - key is depressed

                  0 - key is not depressed

C = 4   **TERMINATE.** This terminates the SuperMouse.

       USER: Enters nothing
       SYSTEM: Returns nothing

       This function terminates the SuperMouse interrupt.

C = 5   **SuperMouse Key Mode**

       USER: Enters

           Reg. A - 0   The tracking of movement by
                            the SuperMouse is discontinued
                            until the SuperMouse key is
                            released. The ASCII character
                            associated with the SuperMouse key
                            that is pressed is placed in
                            the keyboard queue (single
                            character per key depression).

                   - 1   The tracking of movement by
                            the SuperMouse continues even while
                            the SuperMouse key is pressed. The
                            ASCII character associated
                            with the SuperMouse key that is
                            pressed is not placed in the
                            keyboard queue.

       SYSTEM: Returns nothing.

       NOTE!   In both modes the status of the three SuperMouse
               keys can be inquired through SuperMouse function
               3.

C = 6   **SuperMouse Key Programming**

       USER: Enters

           Reg. H   Contains ASCII character to be
                            associated with left SuperMouse
                            key.

           Reg. L   Contains ASCII character to be
                            associated with middle SuperMouse
                            key.

           Reg. D   Contains ASCII character to be
                            associated with right SuperMouse
                              key.

       SYSTEM: Returns nothing.

C = 7   **Move.** This moves the SuperMouse.

   USER: Enters

      Reg. HL   New X-coordinate position of
             SuperMouse.

      Reg. DE   New Y-coordinate position of
             SuperMouse.

   SYSTEM: Returns nothing.

   NOTE! This function will clip the new coordinates
      to the screen resolution.

      X = 0-639
      Y = 0-239

C = 8   **Inquire cursor attributes and status.**

   To BIOS: Nothing

   From BIOS:

     Reg. ACC Bit

       LSB 0 = 1   Small crosshair/nonblinking
               supported
          1 = 1   Full screen
               crosshair/nonblinking
               supported
          2 = 1   Small crosshair/blinking
               supported
          3 = 1   Full-screen crosshair/blinking
               supported.
          4,5,6,7 Currently not used.

C = 9   **Initialize cursor attributes and status.**

   From BIOS: Nothing

   To BIOS:

      Reg. HL = Initial X-coordinate (screen
           coordinate)
        DE = Initial Y-coordinate (screen
           coordinate)

```
          Reg. Acc  Bit

          LSB:      0 = 1     Small crosshair/nonblinking
                    1 = 1     Full screen
                              crosshair/nonblinking
                    2 = 1     Small crosshair/blinking
                    3 = 1     Full-screen crosshair/blinking
                    4,5,6     Currently not used
                    7 = 0     Does not display cursor
                      = 1     Display cursor
```

C = 10   **Set cursor attributes.**

From BIOS: Nothing

To BIOS:

```
          Reg. Acc  Bit

          LSB:      0 = 1     Small crosshair/nonblinking
                    1 = 1     Full-screen
                              crosshair/nonblinking
                    2 = 1     Small crosshair/blinking
                    3 = 1     Full-screen crosshair/blinking
                    4,5,6     Currently not used
                    7 = 0     Does not display cursor
                      = 1     Display cursor
```

C = 11   **Move cursor.**

From BIOS: Nothing.

To BIOS:

```
          Reg. HL      =      Current X-coordinate of cursor
                              (screen coordinate 0-639)

               DE      =      Current Y-coordinate of cursor
                              (screen coordinate 0-239)
```

C = 13   **Terminate hardware cursor.**

From BIOS: Nothing.

To BIOS: Nothing.

Example SuperMouse Program

This program is a graphics doodle routine that utilizes the SuperMouse as the pen. Pressing any key on the SuperMouse or the keyboard terminates the program and returns the control to CP/M.

To create MSCDRAW.COM (executable file)

USER: 1.   Check that the following files reside on your
                   system diskette:

                   ASM.COM
                   LOAD.COM

             2.   Type in the program as shown below under the
                   filename mscdraw.asm.

             3.   Type

                   ASM MSCDRAW<Return>

             4.   Type

                   LOAD MSCDRAW<Return>

             5.   Attach SuperMouse to your system and initialize
                   the SuperMouse through the SETUPTPC program on
                   your diskette.

             6.   Type

                   MSCDRAW<Return>

             7.   Move the SuperMouse on its pad which in turn will
                   doodle on the graphics screen.

```
BDOS      EQU        0005H              ; BDOS entry point.
          ORG        100H
START:    MVI        C, 1               ; Initialize graphics display.
          RST        5
          MVI        C, 2               ; Clear Graphics screen.
          RST        5
          MVI        C, 11              ; Set solid line style.
          LXI        H, 1
          RST        5
          MVI        C, 12              ; Set to white line color.
          LXI        H, 1
          RST        5
          MVI        C, 14              ; Set to replace writing mode.
          LXI        H, 1
          RST        5
          MVI        C, 1               ; Initialize the SuperMouse position (0,
          LXI        H, 0
          LXI        D, 0
          RST        6
          MVI        C, 4               ; Move to (0,0).
          LXI        H, 0
          LXI        D, 0
          RST        5
          LXI        H, 0
          LXI        D, 0
```

```
                 CALL       DRAW               ; Draw dot at (0,0).
LOOP:            MVI        C, 3
                 RST        6                  ; Get SuperMouse's current position.
                 ANA        A                  ; If SuperMouse has moved then draw line
                 JZ         CHKKBD             ; Or else check for key press.
                 CALL       DRAW               ; Draw line to new position.
CHKKBD:          MVI        C, 6               ; CP/M direct console I/O Bdos call.
                 MVI        E, 0FFH            ; Set input flag.
                 CALL       BDOS
                 ANA        A                  ; If key has been pressed then quit
                 JZ         LOOP
                 MVI        C, 0               ; Initialize display to alpha mode.
                 RST        5
                 JMP        0                  ; Warm boot to operating system.
DRAW:            MVI        C, 5               ; Line to command.
                 RST        5
                 RET
                 END
```

GRAPHIC CHARACTER FONT

Table 7-5 is a map of the graphic characters associated with each
ASCII character. The font is stored in the EPROM. Each cell of
the font can be individually accessed through the DRAWSTRING
function.

Internally, this is done as follows:

1.   The ASCII character sent by the USER through the DRAWSTRING
      routine in Table 7-1 is used as an index into the graphics
      character font as shown in Table 7-5.

2.   From the font, the graphics routines can access the bytes
      that comprise the graphics character desired (as shown in
      Figure 7-4).

NOTE!   Sixteen area fill styles have also been stored in the
           FONT. These styles were stored in place of sixteen
           non-printable control characters (as shown in Table 7-
           5).

Table 7-5
Character Font

Contents	ASCII Code	Description
NULL	0	Control codes
SOH	1	
STX	2	

ETX   3

Fill Style Index 0   4   Half-tone patterns
Fill Style Index 1

Fill Style Index 2   5
Fill Style Index 3

Fill Style Index 4   6
Fill Style Index 5

Fill Style Index 6   7
Fill Style Index 7

                             Cross-hatch patterns

Fill Style Index 0   8   Vertical

Fill Style Index 1   9   Horizontal

Fill Style Index 2   10   Diagonal 45 degrees

Fill Style Index 3   11   Diagonal -45 degrees

Fill Style Index 4   12   Vertical/Horizontal

Fill Style Index 5   13   Vertical/Diagonal 45

Fill Style Index 6   14   Vertical/Diagonal -45

Fill Style Index 7   15   Cross diagonals

                             Marker symbols

        .   16   Dot

        +   17   Plus Sign

        *   18   Star

(octagon)   19   Octagon

        X   20   X

                      21

                      22

                      23

                      24

                      25

                      26

                                27

                                28

                                29

                                30

                                31

                        Printable ASCII characters

          SP   32

           !   33
           .
           .
           .
           .

           ~   126

    DEL RUB   127

SAMPLE GRAPHICS ACCESS PROGRAM USING 8080 ASSEMBLY CODE

```
          ORG       100H


main:     call      start     ;initialize graphics, clear screen.
          call      text      ;draw graphic text.
          call      line      ;draw a line, using moveto and lineto.
          call      plylin    ;draw using poly-line.
          call      fill      ;fill a square using horizontal scan.
          jmp       0         ;exit to CP/M
```

How to initialize to graphics mode and clear the graphics
display.

```
start:    mvi       c, 1      ;initialize to graphic mode.
          rst       5
          mvi       c, 2      ;clear graphics display.
          rst       5
          mvi       c, 14     ;set writing mode
          lxi       h, 1      ;to replace mode.
          rst       5
          ret
```

How to set the line style attributes and then draw a line from
(x1,y1) to (x2,y2).

```
line:      mvi        c, 11      ;set line style
           lxi        h, 1       ;solid line style.
           rst        5
           mvi        c, 12      ;set line color
           lxi        h, 1       ;set to white.
           rst        5
           mvi        c, 4       ;move to (x1,y1) = (400,50)
           lxi        h, 400     ;load x1 coordinate
           lxi        d, 50      ;load y1 coordinate
           rst        5
           mvi        c, 5       ;line to (x2,y2) = (500,50)
           lxi        h, 500     ;load x2 coordinate
           lxi        d, 50      ;load y2 coordinate
           rst        5
           ret
```

How to execute the polyline function.
```
      move to (x1,y1) = (100,120)
      line to (x2,y2) = (150,150)
      line to (x3,y3) = (200,120)
      line to (x4,y4) = (250,150)
```

```
plylin:    mvi        c, 3       ;polyline function
           lxi        h, 0FF00h
           mov        e, m
           inx        h
           mov        d, m
           xchg                  ;hl now points to the top of
                                 ;the parameter array.
           mvi        m, 4       ;load vertice's count.
           inx        h
           xra        a
           mov        m, a
           inx        h
           mvi        m, 100     ;load low byte of x1
           inx        h
           mvi        m, 0       ;load high byte of x1
           inx        h
           mvi        m, 120     ;load low  byte of y1
           inx        h
           mvi        m, 0       ;load high byte of y1
           inx        h
           mvi        m, 150     ;load low  byte of x2
           inx        h
           mvi        m, 0       ;load high byte of x2
           inx        h
           mvi        m, 150     ;load low  byte of y2
           inx        h
           mvi        m, 0       ;load high byte of y2
           inx        h
           mvi        m, 200     ;load low  byte of x3
           inx        h
```

```
        mvi       m, 0           ;load high byte of x3
        inx       h
        mvi       m, 120         ;load low  byte of y3
        inx       h
        mvi       m, 0           ;load high byte of y3
        inx       h
        mvi       m, 250         ;load low  byte of x4
        inx       h
        mvi       m, 0           ;load high byte of x4
        inx       h
        mvi       m, 150         ;load low  byte of y4
        inx       h
        mvi       m, 0           ;load high byte of y4
        rst       5              ;perform polyline.
        ret
```

How to draw a string of graphic text. For example, draw 'Hello'
starting at location (x=400,y=120).

```
text:   mvi       c, 4           ;move to command
        lxi       h, 400         ;load x coordinate
        lxi       d, 120         ;load y coordinate
        rst       5
        mvi       c, 13          ;set character color
        lxi       h, 1           ;set to white.
        rst       5
        lxi       h, 0FF00h
        mov       e, m
        inx       h
        mov       d, m
        xchg                     ;hl points to parameter array
        mvi       m, 5           ;load character count.
        mov       b, 5
        lxi       d, string
loop1:  inx       h
        inx       h
        ldax      d
        mov       m, a           ;store character in array.
        inx       d
        dcr       b              ;decrement character count
        jnz       loop1
        mvi       c, 6
        rst       5              ;draw string...
        ret

string: db        'Hello'    ;character string to be drawn.
```

How to fill a square with the selected fill attributes using the
horizontal scan routine (c=7). Square's corner points
(x1=100,y1=50), (x2=200,y2=50), (x3=200,y3=100), (x4=100,y4=100).

```
fill:       mvi         c, 15           ;Set fill interior style.
            lxi         h, 0FF00h
            mov         e, m
            inx         h
            mov         d, m
            xchg                        ;hl points to top of parameter table.
            mvi         m, 3            ;Select cross-hatch fill style
            rst         5
            mvi         c, 16           ;Set fill style index.
            lxi         h, 0FF00h
            mov         e, m
            inx         h
            mov         d, m
            xchg                        ;hl points at top of parameter table.
            mvi         m, 0            ;select vertical cross-hatch style.
            rst         5
            mvi         c, 17           ;Set fill color.
            lxi         h, 0FF00h
            mov         e, m
            inx         h
            mov         d, m
            xchg                        ;hl points to top of parameter table.
            mvi         m, 1            ;select color = white.
            rst         5
            lxi         h, 0FF00h
            mov         e, m
            inx         h
            mov         d, m
            xchg                        ;hl points to top of parameter table.
            mvi         m, 50           ;load y coordinate.
            inx         h
            mvi         m, 0
            inx         h
            mvi         m, 100          ;load x1 coordinate.
            inx         h
            mvi         m, 0
            inx         h
            mvi         m, 200          ;load x2 coordinate.
            inx         h
            mvi         m, 0
            mvi         b, 50           ;load scan line count.
loop2:      push        b               ;save count
            mvi         c, 7            ;select horizontal scan routine.
            rst         5
            pop         b
            lxi         h, 0FF00h
            mov         e, m
            inx         h
            mov         d, m
            xchg                        ;hl points to top of parameter table.
            inr         m
```

```
        dcr     b
        jnz     loop2       ;fill square from bottom to top.
        ret
        end
```

8. TROUBLESHOOTING

The information provided in this section might help you resolve
many operating problems without placing a service call. If your
system does not work properly after you have followed the
suggestions given here, place a service call to your computer
store or service center. See Chapter 9 of the User's Manual
called Care, Service, and Travel, for further information.

Table 8-1
Troubleshooting Procedures

Problem	Action
TPC I IS NOT OPERATIONAL AFTER POWER IS TURNED ON	Unplug power cord and plug it in again. Test for power by plugging something else in the same electrical outlet. Check the line fuse as described in the section, Changing the Fuse, in this chapter, and replace it if necessary. **If the fuse blows again, call your computer store.**
SYSTEM MESSAGE 'SYSTEM "BOOT" (X-REV.X) FROM FLOPPY DISK IN PROGRESS' APPEARS, BUT PROMPT DOES NOT APPEAR	Remove the diskette and turn the power off and on. Reinsert the diskette in the drive and turn the drive latch to the horizontal position. Use the master system diskette and then make a new working copy.
SCREEN DISPLAYS SYSTEM PROMPT BUT DOES NOT RESPOND TO OPERATOR COMMAND	Check to see that the appropriate diskette is in the active drive. Be sure the active drive latch is in the horizontal position. Make sure the keyboard is connected.
TPC I IS USED WITH MODEM AND DOES NOT RESPOND OR TRANSMIT TO NETWORK PROCESSOR	Unplug the modem and replug it on again. Turn its power on and off, and reconnect the modem to the TPC I RS-232 port.

TPC I APPEARS TO BE   Adjust contrast knob on front panel.
RUNNING BUT CURSOR DOES
NOT APPEAR

TPC I VIDEO DISPLAY   Reset the system by pressing <Ctrl>/
ATTRIBUTES DO NOT   <Alt>/<Del> simultaneously so the
CORRESPOND TO RECENTLY-   software can scan the new switch
CHANGED SWITCH SETTINGS   settings.

TPC I LOCKED UP (DOES   Turn the drive latches to the vertical
NOT RESPOND TO KEYBOARD)   position and turn the power off and
                                 back on. Turn the latches back to the
                                 horizontal position.

                                 Check keyboard connector.

TPC I SCREEN IS DEAD   Turn TPC I off and check the fuses.

PRINTER DOES NOT PRINT   Reconnect printer cable, being sure it
WHAT IS TYPED   is completely on the pin connector.

                                 Check printer cable configuration.

                                 Check for printer fault (out of paper
                                 or ribbon).

HARDWARE ERROR MESSAGE   Reset the system by turning the
APPEARS   power off and on.

ESCAPE AND CONTROL   Re-enter commands, paying particular
COMMANDS DO NOT FUNCTION   attention to whether upper- or lower-
AS ANTICIPATED   case characters, ones or zeroes,
                                 are needed.

CHANGING THE FUSE

The TPC I has one exterior fuse which is accessible from the
rear panel.

To check or change the exterior fuse, turn the power off, pull
the plug out of the power outlet, and unscrew the fuse holder (see
Figure 8-1).

The fuse is a small glass cylinder. If the thin wire inside the
fuse is still intact, the fuse should be functional (see Figure
8-2). If the thin strip is broken and/or if the glass is
slightly black, the fuse has blown and must be replaced (see
Figure 8-3). If the glass is totally black, do not replace the
fuse. The problem could be with the system or with the power
outlet. Call your computer store.

Figure 8-1
Exterior Fuse Location

FUSE

Figure 8-2
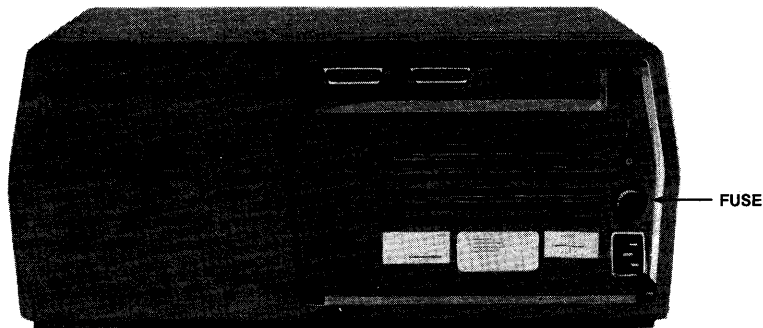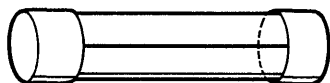Good Fuse

Figure 8-3
Blown Fuse
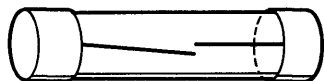
To replace the fuse, slip the old fuse from the fuse holder and
insert a new fuse. The fuse rating is shown on a label above the
fuse holder. See Appendix N for more information about fuses.
You can purchase new fuses from your computer store.

If the newly-replaced fuse blows out immediately, do not replace
it with another fuse. Call your computer store.

APPENDIX A
TPC I SPECIFICATIONS

MICROPROCESSOR/MEMORY

CPU	Z80A 8-bit microprocessor (processor speed 4 MegaHertz)
MEMORY	64 Kbyte dynamic RAM (expandable to 128 Kbytes) 32 Kbyte alpha and graphic display memory 8 Kbyte EPROM

SOFTWARE

OPERATING SYSTEM	CP/M 2.2
GRAPHICS DRIVER	GSX-80
APPLICATIONS SOFTWARE	TeleWrite executive word processor TeleCalc spreadsheet TeleChart business graphics

DISK DRIVES

TYPE	5 1/4-inch slim-line floppy disk drive (two maximum)
DISKETTES	Standard double-sided, double-density 5 1/4-inch floppy diskettes
	Soft sectored, 256 bytes/sector, 18 sectors/track, 40 tracks/side, 2 sides/ diskette, skew rate 6
STORAGE CAPACITY	368.6 kilobytes (formatted) 500 Kbytes per drive (unformatted)
TRANSFER RATE	250 kilobits/second
ACCESS TIME	84 milliseconds (average) 120 milliseconds (maximum)

INPUT/OUTPUT

I/O   One parallel port for a printer
                        One RJ11C port for SuperMouse connection
                        One RS-232C serial port for a modem
                        Asynchronous: Switch-selectable baud rates
                        for RS-232C serial port (75, 150, 300, 600,
                        1200, 2400, 4800, 9600)
                        Word structure: Eight data bits, one stop
                        bit

OPTIONAL   RS-422 port

POWER REQUIREMENTS

EXTERNAL
   U.S.   115 VAC (+/- 12 VAC)
   INTERNATIONAL   230 VAC (+/- 12 VAC)

POWER CONSUMPTION   1.30 amp maximum at 115 VAC
                        0.65 amp maximum at 230 VAC

POWER CORD   NEMA standard 5-15R, 3-prong receptacle
                        (US only)

ENCLOSURE

DIMENSIONS   Height: 8 inches (20.3 cm)
                        Width: 18 inches (45.7 cm)
                        Depth: 15 inches (38.1 cm)

COMPOSITION   Injection-molded plastic

ENVIRONMENT

OPERATING   50 to 85 degrees Fahrenheit
                        10 to 30 degrees Celsius

                        Maximum humidity 95 percent relative,
                        non-condensing

                        Maximum altitude 10,000 ft above sea level

NONOPERATING   32 to 120 degrees Fahrenheit
(SHIPPING)   0 to 50 degrees Celsius

CRT SPECIFICATIONS

SCREEN   9 inches measured diagonally
                        Yellow phosphor

DISPLAYED CHARACTER SET	96-character ASCIIcharacters
	32 control characters
	24 lines
	80 characters per line
	1920 characters per screen
	Video attributes:
	    Half intensity
	    Invisible
	    Blinking
	    Reverse video
	    Underline

CHARACTER FONT	8 x 10 dot matrix 7 x 7 resolution
REPEAT	20 cps auto-repeat
EDITING FEATURES	Typeover Clear screen to space or null Character insert and character delete Line insert and line delete Absolute cursor addressing Erase to end of line or field Line edit
CURSOR	Controls: Left, right, up, down, address Graphics mode: hardware controlled, full screen and single character, blinking or fixed Alphanumeric mode: full block and underline, blinking or fixed

GRAPHICS FEATURES

RESOLUTION	640 pixels horizontal x 240 pixels vertical
FIRMWARE	Built-in firmware for graphics

APPENDIX B
SUGGESTED REFERENCES

The following books are useful references for using the CP/M
operating system in addition to the basic concepts presented in this
manual. Ask at your computer store for the availability of these
books.

1.   Fernandez, Judi N. and Ruth Ashley, Using CP/M: A Self
      Teaching Guide, John Wiley and Sons, 605 Third Avenue, New
      York, NY 1980.

      This is an excellent self-teaching guide, highly recommended
      for the user unfamiliar with CP/M but perhaps familiar with
      computers. Assumes no prior knowledge about operating
      systems.

2.   Hogan, Thom, Osborne CP/M User Guide, 2nd ed.,
      Osborne/McGraw-Hill, 630 Bancroft Way, Dept. UB, Berkeley,
      CA 94710.

      A complete book for the beginning computer operator. Covers
      all aspects of computer operation and CP/M use. Organization
      and presentation are outstanding.

3.   Zaks, Rodney, The CP/M Handbook with MP/M, SyBex, 2344 Sixth
      Street, Berkeley, CA 94710 (415/848-8233).

      Giving very complete details of CP/M, this book gives more
      detail than many beginners will want.

4.   CP/M Summary Guide for Versions 1.4 and 2.X, edited by Bruce
      Brigham, The Rosetta Stone, Post Office Box 35, Glastonbury,
      CT 06025, 203/633-8490.

5.   Murtha, Stephen M. and Mitchell Waite, CP/M Primer, Howard
      W Sams & Co., Inc, 4300 West 62nd St., Indianapolis, IN 46268.

      An excellent introduction to both CP/M and microcomputers.

6.   Townsend, Carl, How to Get Started with CP/M, Dilithium
      Press, Post Office Box 606, Beaverton, OR 97075.

      A succinct discussion of CP/M. Also includes a listing of
      CP/M software suppliers.

7.   CP/M User Manual, Digital Research, P.O. Box 579, Pacific
     Grove, CA 93950.

     GSX-80 User Manual, Digital Research, P.O. Box 579, Pacific
     Grove, CA 95950.

     Published by Digital Research, these references are
     available through your computer store from TeleVideo.

8.   Mostek (for the STI chip), 1215 W. Crosby Rd., Carrollton,
     TX 75006.

TeleVideo welcomes comments from you about these books as well as
names of others that you find useful.

APPENDIX C
BUYING ADDITIONAL DISKETTES

SOURCES

Any TeleVideo Systems dealer or distributor
Retail Computer Center

SPECIFICATIONS

Any new diskettes should meet these specifications to ensure data
integrity:

Type   Floppy minidiskettes
Size   5 1/4-inch
Technology   Double-sided, double-density
Format   Soft-sectored
                Guaranteed for 48 tpi drives
                40 tracks per side

RECOMMENDED BRAND(S)

Brand	Part No.	Manufacturer
Dysan Diskettes 104/2D	800272 (40 tracks)	Dysan Corporation Santa Clara, CA

QUALITY

You should purchase the best quality diskettes available to ensure
data integrity. Diskettes are not that expensive to replace, but
the time spent getting data onto them is very expensive.

LIFE EXPECTANCY

Depending on the care and amount of use given your diskettes, they
may last from six months to two years. Many users automatically
phase out diskettes periodically. Factors such as number of disk
accesses, quality of diskette, environment, and care can
significantly affect their life expectancy.

One of the first signs of diskette wear is incorrect data. The care
with which you handle and store diskettes is probably the most
important single factor in life expectancy of diskettes.

APPENDIX D
CABLE SPECIFICATIONS

The cables that you use should be no more than 50 feet long.

Use of improper cables can result in noncompliance with FCC regulations.

Figure D-1
RS-232C
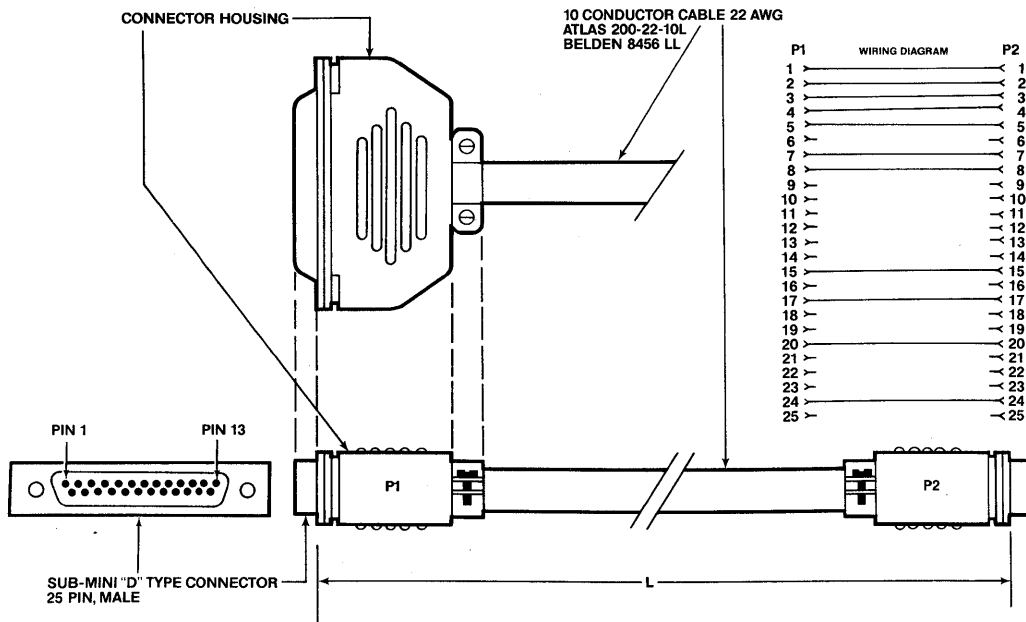
APPENDIX E
PORT ADDRESSES

Table E-1
Port Addresses

Description	Hex Address	Read/Write
DIP switch	00H	
STI port*	20H – 2FH	(modem port)

DART Channel Description

DART Channel A (Data)	30H	R/W (keyboard)
DART Channel A (Command/Status)	32H	R/W (keyboard)
DART Channel B (Data)	31H	R/W (speaker)
DART Channel B (Command/Status)	33H	R/W (mouse in)

SIO (Option Port)

Channel A (RS-422)

data	40H	R/W
command/status	42H	R/W

Channel B (not used)

data	41H	R/W
command/status	43H	R/W

Floppy Disk Drive Controller	90H	W

Bit No.	Description
D0	Ready (active high)
D1	Motor on (active low)
D2	Side select (active low)
D3	Double density (active low)
D4	Drive select 0 (active low)
D5	Drive select 1 (active low)
D6	Drive select 2 (active low)
D7	Drive select 3 (active low)

Floppy Disk Controller: Western Digital 1770

---

*See Appendix B for suggested references for the Mostek STI chip.
Floppy Disk Controller: (Western Digital 1793-02)

---

command/status	80H	R/W
track register	81H	R/W
sector register	82H	W
data register	83H	R/W

Parallel Data Port	70H	W
Parallel Control Port	60H	R/W

Command

DO	Data strobe (-)
D1	Auto feed (-)
D2	Init (-)
D3	Select in (-)

Status

D3	Error (-)
D4	Select
D5	Out of paper
D6	Acknowledge (-)
D7	Busy

System Control Port 1	10H	W

DO	LED 0 (active low)
D1	LED 1 (active low)

System Control Port 2	11H	W

DO	LED 2 (active low)
D1	LED 3 (active low)

System Control Port 3	12H	W

DO	RS 422 enable (active low)
D1	RS 422 WAIT enable (active low)

System Control Port 4	13H	W

DO = 0 and D1 = 0	Enable memory bank 0
DO = 1 and D1 = 0	Enable memory bank 1
DO = 0 and D1 = 1	Enable memory bank 2

Control Register for Alpha or Graphics Mode Selection

                                      C4H   W

```
Bit 0 = 0        alpha mode
        1        graphics mode

Bit 1 = 0        page 1 (alpha mode only)
        1        page 2 (alpha mode only)

Bit 2 = 0        alpha memory access (round off)
        1        graphics memory access (normal CPU address)

Bit 3 = 0        clears the screen
       =1        normal screen
```

Table E-2
CRT Controller Control Register Port

Description		Read/Write
R0	Horizontal total characters number	W
R1	Horizontal displayed characters number	W
R2	Position of horizontal sync pulse	W
R3	Pulse width of horizontal sync pulse	W
R4	Vertical total character number	W
R5	Vertical total adjust	W
R6	Vertical displayed character number	W
R7	Position of vertical sync pulse	W
R8	Mode control	W

```
        BIT 3 = 0        for video display RAM addressing
                         straight binary
```

R9	Maximumscan line address	W
R10	Cursor start raster	W

```
        bit 6,5 - 00     no blinking
                  01     no cursor
                  10     blink at 1/16 field rate
                  11     blink at 1/32 field rate
```

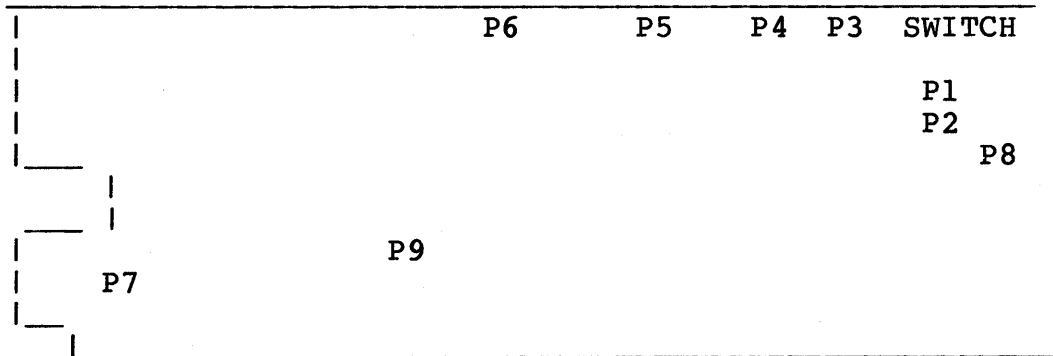|		
R11	Cursor end raster	W
R12	Start address high byte	R/W
R13	Start address low byte	R/W
R14	Cursor position high byte	R/W
R15	Cursor position low byte	R/W

CONNECTOR AND SWITCH DESCRIPTION

The positions of all the connectors on the TPC I board are shown
in Figure E-1. The connector functions are listed in Table E-3.

Table E-3
Connector Functions

Connector	Function
P1	Power
P2	Video
P3	Keyboard
P4	SuperMouse
P5	Parallel
P6	RS-232C (modem)
P7	Floppy Disk
P8	Composite video (not used)
P9	Option board (for RS-422 board)

Figure E-1
Board Connectors

```
|                                    P6        P5      P4  P3   SWITCH  |
|                                                                      |
|                                                             P1       |
|                                                             P2       |
|___                                                               P8  |
|   |                                                                  |
 ___|                                                                  |
|                            P9                                        |
|    P7                                                                |
|___                                                                   |
    |_____
```

The following tables describe the pin assignment of each
connector.

Table E-4
RS-232C Terminal Interface - P6

Pin Number	Description
1	Frame ground
2	Transmit data
3	Receive data
4	Request to send
5	Clear to send
7	Signal ground
8	Data carrier detect
20	Data terminal ready

Table E-5
RS-422 User Connector (Optional) - P9

Pin Number	Description
1	Ground
2	TXD +
3	RXD +
4	RTS +
5	CTS +
6	TXCK -
7	RXCK -
8	Ground
9	TXD -
10	RXD -
11	RTS -
12	CTS -
13	TXCK +
14	RXCK+
15	Test

Table E-6
Power Connector - P1

Pin Number	Description
1	- 12 V
2	Unused
3	Ground
4	+ 5 V
5	+ 12 V

Table E-7
Video Connector - P2

Pin Number	Description
1	Hsync
2	Unused
3	Ground
4	Video
5	Vsync

Table E-8
Floppy Disk Drive Connector - P7

Pin Number	Description
6	Drive select 3
8	Index/sector
10	Drive select 0
12	Drive select 1
14	Drive select 2
16	Motor on
18	Direction select
20	Step
22	Composite write data
24	Write enable
26	Track 0
28	Write protected
30	Composite read data
32	Side select

All the odd number pins are ground.

Table E-9
Keyboard Connector - P3

Pin Number	Description
1	Frame Ground
2	+ 12V
3	Ground
4	Receive data
5	Clock
6	Reset -

Table E-10
SuperMouse Connector - P4

Pin Number	Description
1	No connect
2	No connect
3	Receive data
4	+5 V
5	Ground

Table E-11
Parallel Port - P5

Pin Number	Description
1	Strobe -
2	Data bit 0 +
3	Data bit 1 +
4	Data bit 2 +
5	Data bit 3 +
6	Data bit 4 +
7	Data bit 5 +
8	Data bit 6 +
9	Data bit 7 +
10	Acknowledge -
11	Busy +
12	Out of paper +
13	Select +
14	Auto feed -
15	Error -
16	Initialize -
17	Select input -
18-25	Ground

APPENDIX F
COMMUNICATIONS PORT DRIVER

TPC I MODEM PORT BASED ON MOSTEK 3801

The following information provides an example of how to program
the MOSTEK 3801 STI UART/TIMER chip. This specific example is
more oriented for asynchronous communications, but could be
easily adapted for synchronous communications.

3801 STI REGISTER ADDRESSES

```
IDR       EQU       20H       ;INDIRECT DATA REG
GPIP      EQU       21H       ;GENERAL PURPOSE I/O REG
PVR       EQU       28H       ;POINTER/VECTOR REG
TABCR     EQU       29H       ;TIMER A AND B CONTROL REG
TADR      EQU       2BH       ;TIMER A DATA REG
UCR       EQU       2CH       ;USART CONTROL REG
RSR       EQU       2DH       ;RECEIVER STATUS REG
TSR       EQU       2EH       ;TRANSMITTER STATUS REG
UDR       EQU       2FH       ;USART DATA REGISTER
```

COMMUNICATIONS PORT INITIALIZATION

Application calls the routine at absolute address 0200H to
initialize the communications port. This might be done when the
program is first loaded or whenever you want to raise or lower
DTR. On entry to the communications port initialization routine,
registers B through E specify how the communications port is to
be set up, as listed in Table F-1.

Table F-1
Regs B Through E

B = BAUD RATE	C = PARITY
0 = 110	0 = NONE
1 = 300	1 = ODD
2 = 600	2 = EVEN
3 = 1200	D = DATA BITS
4 = 2400	0 = 8 BITS/CHAR
5 = 4800	1 = 7 BITS/CHAR
6 = 9600	E = DTR STATUS
7 = 19200	ZERO = DROP
	NON-ZERO = ASSERT

Whenever the communications port initialization is done, any
pending input characters should be discarded.

```
COMIN:      PUSH D              ;SAVE D AND E
            MVI D,0
            IN TABCR            ;READ CURRENT TIMER
            STA TABD            ;SAVE FOR LATER
            LXI H,BRBTB         ;GET BIT PATTERN FOR CORRECT BAUD RATE
            MOV E,B
            DAD D
            MOV A,M
            STA TADRD           ;BAUD RATE
            POP D
            MVI B,0
            LXI H,PTYTB
            DAD B
            MOV A,M             ;GET PARITY
            LXI H,DTBTB
            MOV C,D
            DAD B
            ORA M               ;COMBINE WITH DATA BITS
            ORI 88H             ;DIVIDE BY 16 AND 1 START/STOP BIT
            STA UCRD
            LXI H,DTRD          ;INITIAL DTR STATUS
            MOV A,E
            ORA A
            JZ CMIN6
            MVI A,20H           ;ASSERT DTR
CMIN6:      ORA M
            STA DDRD
            MVI A,6
            OUT PVR
            LDA DDRD            ;SET DDR
            OUT IDR
            SUB A
            OUT GPIP            ;SET HANDSHAKE LINES
            MVI A,7
            OUT PVR
            MVI A,80H
            OUT IDR             ;RESET CHANNEL A
            MVI A,0
            OUT TABCR           ;STOP A AND B
            LDA TADRD
            OUT TADR            ;SET BAUD RATE
            LDA TABD
            OUT TABCR           ;RESTART
            LDA UCRD
            OUT UCR             ;SET FORMAT
            MVI A,01H
            OUT RSR             ;TURN ON RXD
            MVI A,05H
            OUT TSR             ;TURN ON TXD
            SUB A
            RET

TADRD:      DB 0                ;BAUD RATE STORAGE
TABD:       DB 0                ;A AND B CONTROL WORD STORAGE
```

```
UCRD:      DB 0              ;FORMAT STORAGE
DDRD:      DB 0              ;DTR AND RTS BITS
```

Table F-2
Baud Rate Bits

```
BRBTB:     DB 64             ;150
           DB 32             ;300
           DB 16             ;600
           DB 8              ;1200
           DB 4              ;2400
           DB 2              ;4800
           DB 1              ;9600
```

Table F-3
Parity Bits

```
PTYTB:     DB 0              ;NONE
           DB 04H            ;ODD
           DB 06H            ;EVEN
```

Table F-4
Character Length Bits

```
DTBTB:     DB 0              ;8 DATA BITS
           DB 20H            ;7 DATA BITS

DTRD:      DB 04H            ;RTS HI DTR LOW
```

RELEASE COMMUNICATIONS PORT

Applications call the routine at absolute address 0210H to
release communications port, that is, to restore it to its
original condition. This routine could be called immediately
before applications returns to CP/M.

```
COMRL:     RET
```

GET COMMUNICATIONS PORT INPUT STATUS

Applications call the routine at absolute address 0220H whenever
it wishes to determine if there is an incoming byte on the
communications port available to be read from the UART. On
return from this routine, register A should be zero if no byte is
available and should be non-zero if a byte is available.

```
COMIS:     IN RSR            ;READ STATUS REGISTER
           ANI 80H           ;LOOK ONLY AT RELEVANT STATUS BIT
           RET
```

READ COMMUNICATIONS PORT INCOMING BYTE

Applications call the routine at absolute address 0230H to read
the incoming data byte into register A (note that all 8 bits are
read). This routine is not called unless a previous call to the
routine at addr 0220H has indicated that there is a byte
available!

```
COMIB:    IN UDR          ;INPUT BYTE
          RET
```

GET COMMUNICATIONS PORT OUTPUT STATUS

Applications call the routine at absolute address 0240H whenever
it wishes to determine if the communications port UART is ready
to accept another byte to be transmitted. On return from this
routine, register A should be zero if the UART is not ready and
should be non-zero if the UART is ready.

```
COMOS:    IN TSR          ;READ STATUS REGISTER
          ANI 80H         ;LOOK ONLY AT RELEVANT STATUS BIT
          RET
```

OUTPUT BYTE TO COMMUNICATIONS PORT

Applications call the routine starting at absolute address 0250H
whenever it wishes to output a byte via the communications port
UART. The byte itself should be supplied in register A. Note
that all 8 bits are set. Applications will not call this routine
unless communications port readiness to accept a byte has been
indicated by calling the status routine at 0240H above.

```
COMOB:    OUT UDR         ;OUTPUT BYTE
          RET


COMEN:    RET             ;ENABLE COMM PORT

COMDS:    RET             ;DISABLE COMM PORT
```

SEND BREAK

Applications call the routine at absolute location 0280H whenever
it wishes to transmit a break condition for approximately one
sec.

```
COMBS:    MVI A,09H       ;SEND BREAK
          MVI B,255
COMB2:    OUT TSR
          DCR B
          JNZ COMB2
          MVI A,01H       ;STOP
          OUT TSR
          RET
```

CHECK CARRIER DETECT STATUS

Applications call the routine at absolute address 0290H whenever
it wishes to determine carrier detect (EIA PIN 8) status on the
communications port. On return, register A is zero if carrier is
not asserted, non-zero if carrier is asserted.

```
COMCS:     IN GPIP          ;READ STATUS REGISTER
           ANI 01H          ;LOOK ONLY AT RELEVANT BIT
           XRI 01H          ;READY IF BIT 0 = 0
           RET
```

APPENDIX G
CHANGING THE DEFAULT LST: DEVICE

The TPC I arrives with the default LST: device set to UL1:. The
following procedure allows you to change the default LST: device
to LPT:.

Using the CP/M SYSGEN and DDT programs, you change a byte in CP/M
and save the modified CP/M on the system area of a disk. Follow
this procedure. User responses are shown in bold print.

USER: 1.   Boot the system.

            2.   Enter

                  SYSGEN\<Return>

SYSTEM: 3.   Displays

                  SYSGEN Vx.x
                  (c) 1982 TeleVideo Systems, Inc.

                  SOURCE DRIVE NAME(OR RETURN TO SKIP)

USER: 4.   Enter

                  A

SYSTEM: 5.   Displays

                  SOURCE ON A:, THEN TYPE RETURN

USER: 6.   Press

                  \<Return>

SYSTEM: 7.   Displays

                 FUNCTION COMPLETE
                 DESTINATION DRIVE NAME(OR RETURN TO REBOOT)

USER: 8.   Press

                 <Return>

SYSTEM: 9.   Displays

                 A>

USER: 10.   Enter

                 SAVE 60 CPMXX.COM<Return>

                 (where CPMXX.COM represents the filename of the
                 modified CP/M)

SYSTEM: 11.   Displays

                 A>

USER: 12.   Enter

                 DDT<BAR>CPMXX.COM<Return>

SYSTEM: 13.   Displays

                 DDT VERS X.Y
                 NEXT   PC
                 3D00 0100
                 -

USER: 14.   Enter

                 S1FDA<Return>

SYSTEM: 15.   Displays

                 1FDA D5

USER: 16.   Enter

                 95<Return>

                 (where D5 is the code to enable the UL1 device and
                 95 changes it to LPT. Table G-1 lists the
                 available device codes.)

Table G-1
Available Device Codes

Code	Device
15	TTY: Serial printer, pin 20 for DTR
55	CRT: Serial device using XOn/XOFF protocol through modem port
95	LPT: Serial device using ETX/ACK protocol through modem port
D5	UL1: Parallel printer port

NOTE!   In the above procedure, we have selected device LPT.

SYSTEM: 17.   Displays

1FDB 0D

(this is the next hexadecimal address)

USER: 18.   Enter

.<Return>

(BE SURE TO ENTER A PERIOD BEFORE <Return>. This is entered because you do not want to change this address.)

SYSTEM: 19.   Displays

—

20.   Enter

^C

SYSTEM: 21.   Displays

A>

USER: 22.   To generate the modified system, run SYSGEN again. Enter

SYSGEN<Return>

SYSTEM: 23.   Displays

                  SYSGEN Vx.x
                  (c) 1982 TeleVideo Systems, Inc.

                  SOURCE DRIVE NAME(OR RETURN TO SKIP)

USER: 24.   Press

                  <Return>

SYSTEM: 25.   Displays

                  DESTINATION DRIVE NAME(OR RETURN TO REBOOT)

USER: 26.   Press

                  A

SYSTEM: 27.   Displays

                  DESTINATION ON A:, THEN TYPE RETURN

USER: 28.   Press

                  <Return>

SYSTEM: 29.   Displays

                  FUNCTION COMPLETE
                  DESTINATION DRIVE NAME(OR RETURN TO REBOOT)

USER: 30.   Press

                  <Return>

SYSTEM: 31.   Displays

                  A>

Now you can test your new system by booting the system from the
drive containing the new version. The modifications will not take
effect until the system is booted on the new customized CP/M.

Crossover cable must be used to connect between modem port and
printer port when ETX/ACK protocol printer is used. Refer to
Appendix H for cable configuration.

APPENDIX H
DEFAULT DEVICE ASSIGNMENT

CON: = CRT:

RDR: = PTR:

PUN: = PTP:

LST: = UL1:

UL1: device uses Centronics interface as the response to the printer busy status. Types of printers supported: Epson Centronic parallel interface printers.

A Centronic parallel printer cable must be used with a 25-pin D-shell on the computer end.

Optional device assignment:

1.   LST: = TTY:

TTY: device uses DTR protocol for a modem port. Crossover cable must be used.

2.   LST: = CRT:

CRT: device uses X-ON/X-OFF protocol for a modem port. Crossover cable must be used to connect from the modem port to the printer device.

3.   LST: = LPT:

LPT: device uses ETX/ACK protocol for a modem port. Types of printers supported are NEC Spinwriter and Diablo daisywheel. Crossover cable must be used.

Table H-1
Crossover Cable Configurations

System RS-232C

Modem Port Pin #	Printer Pin #
1	1
2	3
3	2
4	5
5	20
7	7
8	4
20	6 and 8

APPENDIX I
ASCII CODE CHART

Figure I-1
ASCII Code Chart

Bits	7	6	5			0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
	4	3	2	1	Column → / ↓ Row	0	1	2	3	4	5	6	7
	0	0	0	0	0	NUL	DLE	SP	0	@	P	`	p
	0	0	0	1	1	SOH	DC1	!	1	A	Q	a	q
	0	0	1	0	2	STX	DC2	"	2	B	R	b	r
	0	0	1	1	3	ETX	DC3	#	3	C	S	c	s
	0	1	0	0	4	EOT	DC4	$	4	D	T	d	t
	0	1	0	1	5	ENQ	NAK	%	5	E	U	e	u
	0	1	1	0	6	ACK	SYN↓	&	6	F	V	f	v
	0	1	1	1	7	BEL	ETB	'	7	G	W	g	w
	1	0	0	0	8	BS←	CAN	(	8	H	X	h	x
	1	0	0	1	9	SKIP HT	EM	)	9	I	Y	i	y
	1	0	1	0	10 (a)	LF	SUB	★	:	J	Z	j	z
	1	0	1	1	11 (b)	VT↑	ESC	+	;	K	[	k	{
	1	1	0	0	12 (c)	FF→	FS	,	<	L	\	l		
	1	1	0	1	13 (d)	CR	GS	−	=	M	]	m	}
	1	1	1	0	14 (e)	SO	HOME RS	.	>	N	^	n	~
	1	1	1	1	15 (f)	SI	NEW LINE US	/	?	O	—	o	DEL RUB

ASCII Code Table
Abbreviations For Control Characters

NUL	null	FF	form feed	CAN	cancel
SOH	start of heading	CR	carriage return	EM	end of medium
STX	start of text	SO	shift out	SUB	substitute
ETX	end of text	SI	shift in	ESC	escape
EOT	end of transmission	DLE	data link escape	FS	file separator
ENQ	enquiry	DC1	device control 1	GS	group separator
ACK	acknowledge	DC2	device control 2	RS	record separator
BEL	bell	DC3	device control 3	US	unit separator
BS	backspace	DC4	device control 4	SP	space
HT	horizontal tabulation	NAK	negative acknowledge	DEL	delete
LF	linefeed	SYN	synchronous idle		
VT	vertical tabulation	ETB	end of transmission block		

APPENDIX J
SYSTEM DISKETTE FILE LIST

This is a list of the files that were on the TPC I system
diskette when it was initially released. TeleVideo reserves the
right to change these files without notification to the customer.
If you have any questions about the files on your diskette,
contact your computer store.

Table J-1
System Diskette Files

Filename	Description
MOVCPM.COM*	Program to relocate the CCP and BDOS to generate a CPMxx.COM file.
DDT.COM*	Debug program (Dynamic Debug Tool) for the assembly language.
SYSGEN.COM*	CP/M utility program to copy system tracks 0 and 1.
COPYDISK.COM	Utility program to copy an entire diskette.
GENGRAF.COM*	Utility to combine applications program and GSX loader into executable .command file.
DD7220.PRL*	GSX graphics driver to support a plotter (Hewlett Packard 7220).
DDMX80.PRL*	GSX graphics drive to support a printer (Epson MX80).
DDTS803.PRL	GSX graphics driver to support the TPC I.
PIP.COM*	CP/M utility program to copy files.
ED.COM*	CP/M text editing program.
LOAD.COM*	Program to generate an executable program (.command file) from the HEX file generated by the assembler.
SETUPTPC.COM	Utility program to reconfigure the system parameters.
DD7470.PRL*	GSX graphics driver to support a plotter (Hewlett Packard 7470A).

GSX.SYS*   Graphics System Extension file (GSX loader).

SURF1   Graphics data for demostration program.

STAT.COM*   CP/M utility program. For capabilities of
                      the STAT command, see Chapter 3.

SUBMIT.COM*   CP/M utility program for submitting CP/M
                      commands in a batch mode. Works with
                      XSUB.COM.

DUMP.COM*   Program to display a file in HEX numbers.

TOD.COM   Time of Day program.

DDHI3M.PRL*   GSX graphics driver to support a plotter
                      (Houston Instruments DMP 3/4).

ASSIGN.SYS   Graphics file that works with the
                      demonstration program for determining the
                      output device of the demo.

DEMOTXT2.BIN   Data file used by the system demonstration
                      program.

ASM.COM*   8080 assembler program. Generates a HEX file
                      which can be used by the load program.

XSUB.COM*   Works with the SUBMIT file submitting CP/M
                      commands.

FORMAT.COM   Utility program to format a diskette.

DDHI7M.PRL*   GSX graphics driver to support a plotter
                      (Houston Instruments DMP 6/7).

GSXREAD.ME   Explanation about GSX (Graphics System
                      EXtension) and the TPC I demonstration
                      program.

TPCIDEMO.COM   System demonstration program.

DEMOSCRN   Contains data needed by the demonstration
                      program.

* Digital Research, Inc.(DRI) files

APPENDIX K
PROGRAMMER'S QUICK REFERENCE GUIDE

This information can also be found on the easy-to-use reference
card that came with the TPC I this Manual.

The spaces shown in the escape codes are for clarity and should
not be entered.

Table K-1
Escape Sequence List

Function	Command
MONITOR	
Monitor mode ON	ESC U
Monitor mode OFF	ESC X or
	ESC u
DISABLE/ENABLE KEYBOARD	
Disable keyboard	ESC #
Enable keyboard	ESC "
CURSOR	
Home	^^
Carriage return	^M
Linefeed/cursor down	^J or ^V
Cursor up	^K or ESC j
Backspace/cursor left	^H
Cursor right	^L
Cursor off	ESC . 0
Blinking block cursor	ESC . 1
Steady block cursor	ESC . 2
Blinking underline cursor	ESC . 3
Steady underline cursor	ESC . 4
KEYCLICK AND BELL	
Keyclick on	ESC >
Keyclick off	ESC <
Ring bell	^G

VISUAL ATTRIBUTES

Default video (green on black)	ESC G 0
Invisible normal video	ESC G 1
Blink	ESC G 2
Invisible blink	ESC G 3
Reverse video (reverse of default)	ESC G 4
Invisible reverse	ESC G 5
Reverse and blink	ESC G 6
Invisible reverse and blink	ESC G 7
Underline	ESC G 8
Invisible underline	ESC G 9
Underline and blink	ESC G :
Invisible underline and blink	ESC G ;
Reverse and underline	ESC G <
Invisible reverse and underline	ESC G =
Reverse and blink and underline	ESC G >
Invisible reverse and blink and underline	ESC G ?
Half Intensity ON	ESC)
Half Intensity OFF	ESC (

ADDRESS CURSOR

Address cursor (row column)	ESC = r c

TAB

Set (column) tab	ESC 1
Tab	^I
Back tab	ESC I
Clear tab	ESC 2
Clear all tabs	ESC 3

EDITING TEXT

Character insert	ESC Q
Character delete	ESC W
Line insert	ESC E
Line delete	ESC R
Erase to end of line	ESC T
Erase to end of line with nulls	ESC t
Erase to end of page	ESC Y
Erase to end of page with nulls	ESC y

CLEAR

Clear all to spaces	ESC + or ^Z
Clear all to nulls	ESC :
Clear all to half- intensity spaces	ESC ,
Clear all to nulls and reset half intensity	ESC *

FUNCTION KEYS

Program function keys   <Shift>/<Esc> | p1 p2 message ^Y
(p1=number of function key
 p2=1
 ^Y= termination character)

Table K-2
Switch Setting List

Section	Setting		Function
1	closed	(down)	Baud rate
2	closed	(down)	Baud rate
3	closed	(down)	Baud rate
4*	closed	(down)	TPC I
5*	closed	(down)	Local
	open	(up)	Remote
6*	closed	(down)	Not used
7*	closed	(down)	60 Hz
	open	(up)	50 Hz
8	closed	(down)	Black on yellow screen
	open	(up)	Yellow on black screen

* required settings

Table K-3
Baud Rate List

Switch Section			Baud Rate
1	2	3	
C	C	C	9,600
O	C	C	4,800
C	O	C	2,400
O	O	C	1,200
C	C	O	600
O	C	O	300
C	O	O	150
O	O	O	75

C=closed
O=open

APPENDIX L
TPC I DEMONSTRATION PROGRAM

The Digital Research GSX (Graphics System eXtension) gives programs the ability to easily display images on a number of different graphics devices. It provides a standard interface so that different graphics devices look the same to applications programs.

We have included a demonstration program to show some of the capabilities of the GSX on the TPC I. To run the program, you need the following files on your logged disk:

TPCIDEMO.COM	Demonstration program itself
assign.sys	Explained below
demotxt2.bin	Data for text fonts
surf1	Data for a curved surface
ddts803.prl	Software to talk to the TPC I
demoscrn	Contains data needed by demonstration program

NOTE!   Plotters must be connected to the RS-232 port and printers must be connected to the printer port.

To run the program, enter

        TPCIDEMO<Return>

The program asks you if you want the demonstration displayed on the CRT, a printer, or a plotter. To see the demonstration on the TPC I screen, follow these instructions.

USER: Enter

            1<Return>

SYSTEM: Displays

            reading in data

After about 30 seconds, the program will start displaying a series of different screens giving information about the TPC I. The program runs unattended, and continues to run until you press the D key. The program then finishes drawing the current page and returns to CP/M.

If you wish to look at any single page, press any key other than D while the page is being drawn. That page stays on the screen until you press any other key.

Hardcopy Options

For the following printers, you can see the screens from the demonstration on the printer by choosing option number 21 when the demonstration program starts running. The Epson must be plugged into the TPC I printer port, located on the back panel of the main unit.

       Epson MX-80 printer with Graftrax-Plus
       Epson MX-80 type 3
       Epson FX-80
       Epson MX-100

For the following plotter, you must plug it into the RS-232 port of the TPC I, and select number 11 when the program starts. If you do not get output on the plotter, you can use the SETUPTPC Program to check the baud rates and data formats and make sure that the PUN: logical device is set up to be the UR1: physical device.

       Hewlett Packard 7470 plotter

How the Assign.sys File Works

If you have a printer or plotter and wish to see output on it, you must make sure that software support is provided for that device. See Tables L-1 and L-2 for the printers and plotters that are supported.

Table L-1
Supported Printers

Printer Name	File for Device
Epson MX-80 with Graftrax Plus*	ddmx80.prl
Epson MX-80 type 3	ddmx80.prl
Epson FX-80	ddmx80.prl
Epson MX-100	ddmx80.prl

Table L-2
Supported Plotters

Plotter Name	File for Device
Hewlett-Packard 7470*	dd7470.prl
Hewlett-Packard 7220	dd7220.prl
Houston Instruments HiPlot 7M	ddhi7m.prl
Houston Instruments HiPlot 3M	ddhi3m.prl

* The Epson MX-80 printer with Graftrax-Plus and the Hewlett-Packard 7470 plotter have been tested by TeleVideo to be fully compatible to the TPC I. The other drivers have not been tested by TeleVideo System.

The file named ASSIGN.SYS determines which device will get output. The file comes set up to talk to the TPC I screen, the Epson MX-80 printer or the Hewlett-Packard 7470 plotter, and should look like:

```
21 @:ddmx80
01 @:ddts803
11 @:dd7470
```

The dd portion of the filename stands for device driver. The @ (at-sign) stands for the default drive. For example, if you are running from drive A, CP/M takes the "@" to mean "A." The format of the file must be as shown: two digits specifying the device number, followed by a space, the disk letter and file name. GSX assumes a **.prl** extension, so you should not include the extension in specification of the filename.

Note that the numbers at the beginning of each line correspond to the numbers asked for by the demonstration program. When you specify a number to the demonstration program, it tells the GSX to look for that number in the assign.sys file. The GSX uses the file name associated with that number to send graphics commands to the device associated with the file.

The GSX assumes that device 1 (screen driver - DDTS803.PRL) is the default console, so 1 should always be set to the TPC I. Devices 1-10 are CRTs, 11-20 are plotters and 21-30 are printers. The ASSIGN.SYS file that comes with the system uses 1, 11 and 21.

GSX assumes that the printer has been set up as the LST: device and the plotter as the RDR and PUN: device. You can use the SETUPTPC Program to make sure that the TPC I is set up correctly.

Changing the ASSIGN.SYS File

If you have a printer other than the Epson MX-80, or a plotter other than the HP 7470, you must edit the ASSIGN.SYS file using a text editor. For example, the Houston Instruments HiPlot 7M uses the DDHI7M.PRL file. You should change the third line in the file to read

```
11 @:ddhi7m
```

and save the new version of the file. The largest driver should be on the first line in the ASSIGN.SYS file, so use the stat program to check the driver sizes if you change the ASSIGN.SYS file. You must plug the H1 plotter into the RS-232 port on the back of the TPC I, and check the baud rates and data formats.

APPENDIX M
CONFIGURING FOR 230 VOLTS

To reconfigure the TPC I for 230 volts, you must change the fuse, change a switch setting on the back of the unit, and change the voltage switch on the back of the unit. See Chapter 2, Setting Up Your TPC I, in the User's Manual for information about the switch settings.

INSERTING THE FUSE FOR 230 VOLTS

The TPC I is shipped with the fuse installed to correspond to 115 volts. You need to acquire a 230 volt fuse that corresponds to the requirements in Table M-1.

Table M-1
Fuse Table

Amperes	Maximum Fuse Voltage	Power Standards
1.5	250VAC	115V U.S.
.75	250VAC	230V International

To change the fuse to correspond to 230VAC, be sure that the power is turned off and the power plug is not connected.

Unscrew the fuse holder from the back panel of unit and remove the 1.5A fuse. Place one end of the .75V fuse into the holder and insert the fuse into the unit. Press gently while screwing the holder back into place. See Figure M-1.

Figure M-1
Fuse
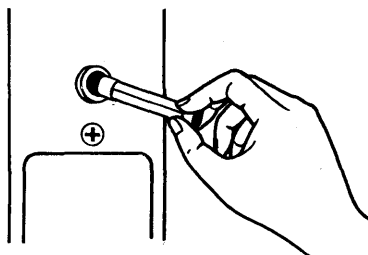
Changing the Voltage Switch to 230 Volts

The system is configured at the factory for 115 VAC.

To change the voltage configuration to 230 VAC you need to switch the voltage switch on the back panel of the system.

The voltage switch is locked into place by a bar that is removable. To remove the bar and switch from 115 VAC to 230 VAC, follow these directions:

1.   Using a Phillips screw driver, remove the screw from the bottom of the bar that spans the 230V side of the switch. See Figure M-2.

Figure M-2
Voltage Switch
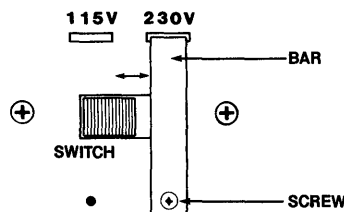
2.   Carefully remove the bar.

3.   Firmly push the switch to the 230V position (to the right).

4.   Match the screw hole of the bar with that under the 115V position and fit the top horizontal section carefully into the slot above the 115V position.

5.   Insert and tighten screw.

STOP!   Contact your dealer if you are not sure that your power requirements match that of the unit. **Trying to operate the unit with the wrong power configuration can seriously damage the system.**

TPC I DOCUMENTATION COMMENT CARD

TeleVideo invites your comments regarding the TPC I User's Manual
and the TPC I System Reference Manual. Your suggestions will
help us to develop documentation that will enhance the value of
TeleVideo computers to our customers.

You may wish to comment on the manuals' organization, clarity,
accuracy, and completeness. if you find discrepancies, please
list them by page, figure, or table number. If you need more
space, please attach a separate sheet.

User's Manual. Is the manual clear and complete? If not, please
comment, giving the date of publication (listed on the lower
right hand back cover) and giving specific references.

_____

_____

_____

_____

_____

System Reference Manual. Is the manual clear and complete? If
not, please comment, giving the date of publication (listed on
the lower right hand back cover) and giving specific references.

_____

_____

_____

_____

_____

What is the title/occupation of the primary user?

Check the description that applies to you:
_____End User   _____Dealer   ____Distributor   ____OEM

Name_____ Phone_____/____-_____

Title_____

Company_____

Address_____

City_____ State_____ Zip_____