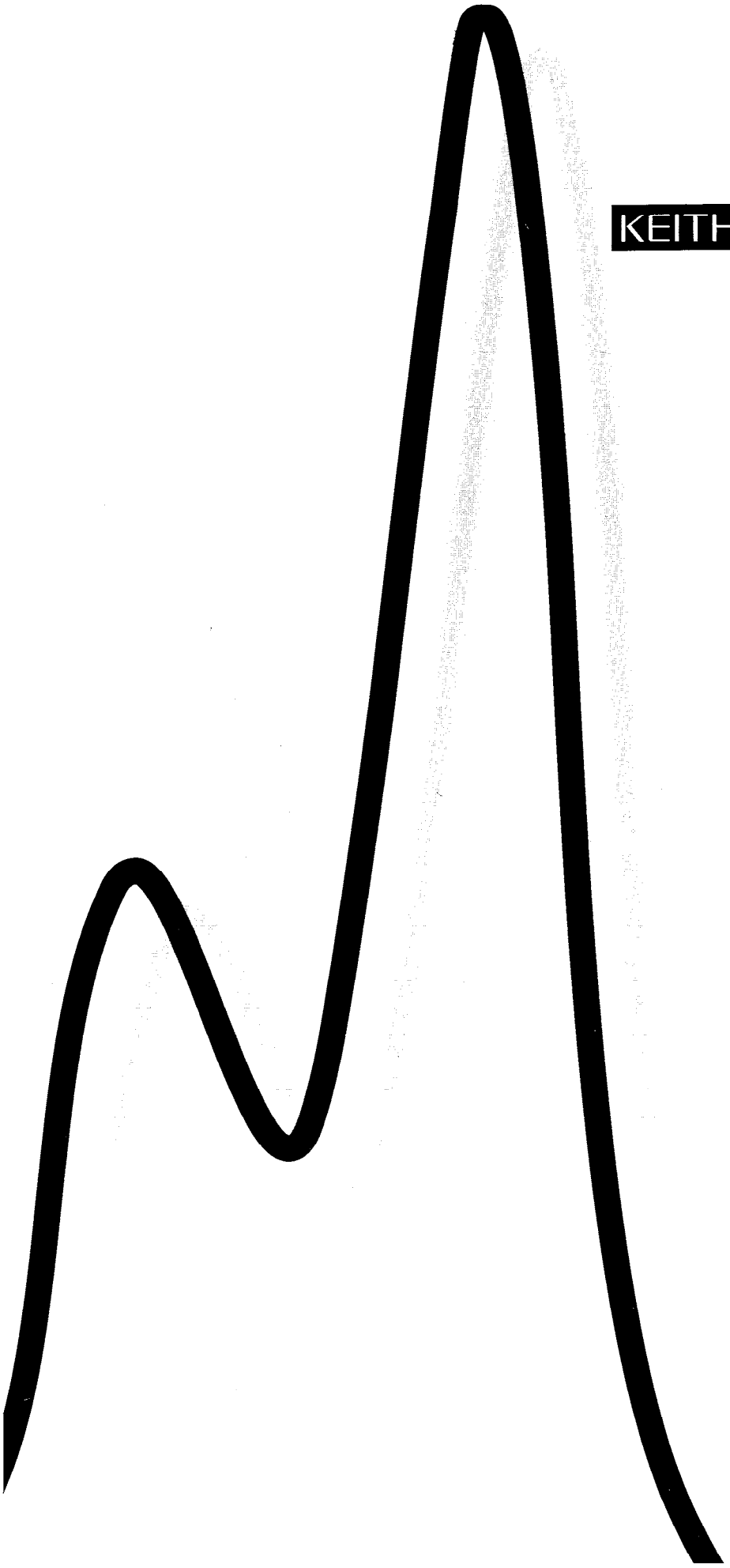
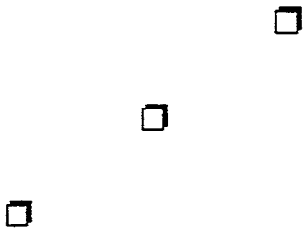


KEITHLEY METRABYTE

PCIP-SST

USER'S GUIDE

PCIP-SST



User Guide

for the

PCIP-SST

Function Generator

Board

Revision C - May 1992

Part Number: 24849

KEITHLEY DATA ACQUISITION - KEITHLEY METRABYTE/ASYST

440 MYLES STANDISH BLVD., Taunton, MA 02780
TEL. 508/880-3000, FAX 508/880-0179

Warranty Information

All products manufactured by Keithley MetraByte are warranted against defective materials and worksmanship for a period of one year from the date of delivery to the original purchaser. Any product that is found to be defective within the warranty period will, at the option of Keithley MetraByte, be repaired or replaced. This warranty does not apply to products damaged by improper use.

Warning

Keithley MetraByte assumes no liability for damages consequent to the use of this product. This product is not designed with components of a level of reliability suitable for use in life support or critical applications.

Disclaimer

Information furnished by Keithley MetraByte is believed to be accurate and reliable. However, the Keithley MetraByte Corporation assumes no responsibility for the use of such information nor for any infringements of patents or other rights of third parties that may result from its use. No license is granted by implication or otherwise under any patent rights of Keithley MetraByte Corporation.

Note:

Keithley MetraByte™ is a trademark of Keithley Data Acquisition.

Basic™ is a trademark of Dartmouth College.

IBM® is a registered trademark of International Business Machines Corporation.

PC, XT, AT, PS/2, and **Micro Channel Architecture®** are trademarks of International Business Machines Corporation.

Microsoft® is a registered trademark of Microsoft Corporation.

Turbo C® is a registered trademark of Borland International.

Contents

■ ■ ■

CHAPTER 1

INTRODUCTION

1.1 FUNCTIONAL DESCRIPTION

The PCIP-SST is a 1 Hz to 5 MHz function generator board that plugs directly into any I/O slot of an IBM PC/XT/AT, IBM PS/2 Model 25/30 or compatible. The Board offers the features and accuracy of a standard bench-top generator. However, instead of LEDs, LCDs or a vernier knob, the PCIP-SST uses the PC display screen as its control panel and its keyboard/mouse as its controls. Instead of dialing and setting controls to set the frequency, you simply give the PCIP-SST the desired frequency via keyboard or mouse.

During Board control, the PC display operates in a *pop-up* mode. When activated, the Pop Up display takes up one third of the screen, allowing up to three separate PCIP instrument displays on the screen at a time. When the SST is deactivated, the Pop Up disappears and the display returns to its original state.

PCIP-SST outputs are fully isolated from the computer and line power. Connections to the PCIP-SST use four banana-type plugs or alternatively two BNC connectors that plug directly into the banana plugs. One connector is for the output, while the second provides a TTL-compatible trigger.

1.2 FEATURES & APPLICATIONS

The following lists contain features and applications of the PCIP-SST board.

Features

- Extremely low cost: over 50% less than GPIB based bench instruments.
- Program with simple English words, no complex IEEE-488 commands.
- 1 Hz to 5 MHz operation.
- Sine, Square, and Triangle output waveforms.
- Plugs directly into any IBM PC/XT/AT or Compatible– No IEEE-488 or GPIB Interface required.
- 20 Volt Peak-Peak output with ±5 Volt offsets (in 2.44 mV Steps).
- Fully isolated from the host computer.
- Operates in Two Modes:

  1. As a standard bench instrument, with the display shown on the computer screen and either keyboard or mouse to select functions.

  2. Under program control from any computer language.

- All software included; no hidden software costs.

Applications

- Automatic Test.
- General Purpose Bench Instrumentation.
- Production Test.
- Frequency Analysis.
- Transfer Function Analysis.

1.3 OPERATING MODES

The PCIP-SST operates in either of two modes: *Programmed* Mode and *Standard* or *Bench-top Instrument* Mode. The Programmed Mode allows you to write programs (in any language) that control the board directly and is extremely useful when using the board in an automated test or experiment.

In the Bench-Top Instrument Mode, the board functions as a standard bench top instrument. The only difference is that the settings of the board are displayed on the computer screen, not on a variety of dials; and the output frequency, amplitude, DC offset, and wave-type are all set from the keyboard or via a mouse, and not by dials and buttons.

Programmed Mode

PCIP-SST software comes in the form of a "device driver." This device driver is loaded into the machine during the System configuration (CONFIG.SYS) each time you reboot or turn on your computer; or it may be loaded/unloaded by the LS.COM and US.COM programs. The PCIP-SST's device name is $SST. The program interface is language independent and to access the PCIP-SST simply "open" the Device called $SST. From there, commands are PRINTED to the board and INPUTTED from the board just as if the software were writing to or reading from a disk file. The PCIP-SST is programmed via a series of English-like commands. To generate a 673 Khz sine wave, with a 3.2 Volt peak to peak output, the command is "SET Sine Freq 673, amplitude 3.2".

PCIP-SST Standard Instrument Mode

You may pop all instruments up or down using *Hot Keys* , which may be defined by the user. While an instrument is popped up, you may change its parameters manually with either a keyboard or mouse. You may display up to three PCIP instruments simultaneously (except the PCIP-DLA and PCIP-SCOPE, since they require full screen for a single instrument).

1.4 THE DISTRIBUTION SOFTWARE

This manual refers to the PCIP-SST software as the *Distribution Software.* The Distribution Software contains driver files, utility files, calibration files, and programming example files. For a list of these files, with descriptions, refer to the ASCII file *FILES.DOC.*

1.5 SPECIFICATIONS

Waveforms: Sine, Square, Triangle.

Frequency Range: 1 Hz TO 5 MHz.

Harmonic Distortion: 1 Hz to 500 kHz 1st Harmonic Down 35 dB; 500 kHz TO 5 MHz 1ST Harmonic Down 30 dB.

Triangle Linearity: > 97% Linear (1 Hz to 5 Mhz).

Square Wave: Rise Time: 50 nS.
Fall Time: 50 nS.
(Overshoot Less Than 3%).

Amplitude Flatness: ±4% (1 hZ -- 5 MHz).

Amplitude Range: 20 V Peak to Peak (in 4096 Discrete Steps of 4.88 mV Increments).

DC Offset: -5 V TO +5 V (in 4096 Discrete Steps of 2.44 mV Increments).

Total Output: Cannot go beyond +10 V or -10 V including output waveform and offset.

Output Impedance: 50 Ohms.

Isolation from PC: 500 Volts.

Frequency Measurement Accuracy: ±0.2% in auto-calibrate/tracking mode.

Output Frequency Accuracy and Resolution: The output frequency of the PCIP-SST will be accurate to within ±0.5% typically, and ±3.5% worst case. However, in applications where frequency accuracy is critical, the on-board frequency counter can measure the output frequency to an accuracy of ±0.2%. The "TRACK" function can then be executed to set the output frequency as close to the desired output as the frequency resolution will allow (accuracy/resolution specifications are included below).

Output Frequency Resolution:

RANGE	MINIMUM RESOLUTION
1 Hz to 100 Hz	.03 Hz
100 Hz to 10 Khz	3 Hz
10 Khz to 500 Khz	150 Hz
500 Khz to 5 Mhz	1500 Hz

Output Frequency Accuracy: Standard Mode: ±% typical; ±4% worst case.

AutoCalibrate Mode: ±0.2% or 1 step of minimum resolution frequency (whichever is greater).

Frequency Accuracy Temperature Coefficients: Standard Mode: ±0.25% per Degree C.
Autocalibrate Mode: ±20 ppm / Degree C.

1.6 HOW TO USE THIS MANUAL

This manual assumes you are familiar with the PC and its operation. At times, you may find it necessary to refer to the PC operator's manual and a reference manual for the language used in custom application programs. Be sure to have these handy.

Chapter 2, *Installation*, describes unpacking and inspection procedures, setting the Base Address and memory configuration switches, installation of the Board, and connecting peripherals.

Chapter 3, *Getting Started*, tells you how to make copies of the PCIP-SST Distribution Software. Additionally, the chapter describes how to install the device drivers and modify your CONFIG.SYS file. This chapter also describes how to use the PCIP-SST's Pop-Up Menu to set the operating parameters.

Chapter 4, *The Pop-Up Display*, describes the use of the PCIP-SST in its Popped-Up Mode.

Chapter 5, *Programming*, describes the Language Independent Interface Commands. It also details the on-board registers. This chapter is for programmers who wish to create their own application programs.

Chapter 6, *Maintenance & Repair*, gives a procedure for calibrating the PCIP-SST. Warranty information and Return-to-Factory instructions are also provided.

The Appendices contain other useful information. Appendix A contains examples of how to access the PCIP-SST software driver from the Pascal, C, Turbo-Pascal, and Turbo-C programming languages.

■ ■ ■

CHAPTER 2

INSTALLATION

2.1 GENERAL

This chapter describes the installation of your PCIP-SST. The following information is provided: unpacking and inspection procedures, setting of the Base Address and memory-configuration switches, installing the Board, connecting the system, and backing up the Distribution Software.

The PCIP-SST is factory-calibrated and should be recalibrated as required; see Section 6.2 for calibration procedures.

2.2 UNPACKING & INSPECTION

After unpacking the Board from its outer shipping material, proceed as follows:

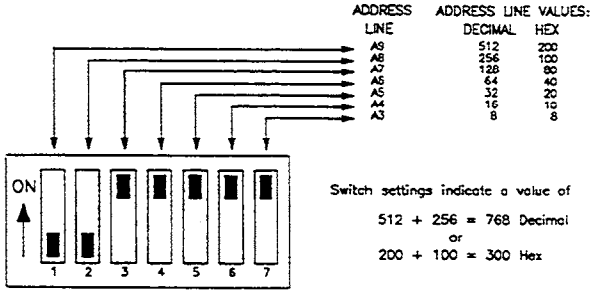
1. Remove the Board from its packing material while placing one hand firmly on a metal portion of the system chassis. This will prevent any damage to board components from static electricity.
2. After allowing a moment for static electricity discharge, carefully remove the Board from its anti-static wrapping.
3. Inspect the board for any possible damage. If any sign of damage is apparent, return the board to the factory as described in Section 5.4.

Check the contents of the entire Board package against the packing list. Report any missing items to the manufacturer.

2.3 DIP SWITCH SETTING

The Board's only DIP switch sets the Base Address. The PCIP-SST uses a block of 8 non-overlapping I/O addresses, and the Base Address Switch selects where in the PC's I/O space the board resides.

The Base Address Switch is a 7-position DIP switch located in the lower right-hand corner of the board. To set the appropriate base address, use a pen-tip to move the individual switches into the ON or OFF position. For example, to set a base address of 300h (768 decimal), move Switches 1 and 2 into the OFF position while leaving all other switches in the ON position. As a reference, Figure 2-1 shows the Base Address Switch set for 300h.



The factory setting for Base Address is 300h (see diagram); this Address is typically free in most computers. However, if more than one PCIP instrument is to be used, each must use its own distinct Base Address.

A new Base Address must be within the range 100 to 3FFh (512 to 1023 Decimal). Select a new Base Address from the following list of addresses reserved for standard IBM devices.

ADDRESS(HEX)	DEVICE	ADDRESS(HEX)	DEVICE
000-1FF	Internal system	378-37F	LPT1:
200-20F	Game	380-38C	SDLC comm.
210-217	Expansion unit	380-389	Binary comm.
220-24F	Reserved	3A0-3A9	Binary comm.
278-27F	Reserved	3B0-3BF	Mono dsp/LPT1:
2F0-2F7	LPT2:	3C0-3CF	Reserved
2F8-2FF	COM2:	3D0-3DF	Color graphics
300-31F	Prototype card	3E0-3E7	Reserved
320-32F	Hard disk	3F0-3F7	Floppy disk
		3F8-3FF	COM1:

This table covers the standard IBM I/O options (most compatibles are identical), but if you have other I/O peripherals (special hard disk drives, special graphics boards, prototype cards, etc.), they may be making use of I/O addresses not listed in the table.

Usually, a good choice is to put the PCIP-SST at Base Address &H300 or &H310 (the default is 300h). As an aid to setting the Base Address DIP switch, a graphical switch-position display program, DIPSW.EXE may be run from the DOS prompt. To run the installation aid program, type: **`DIPSW`**.

When you get the "Desired base address?" prompt, type in your choice in decimal or IBM &H— format and press [**Enter**]. The program will check to see if a PCIP-SST exists at that address and draw a picture of the correct positions of the toggles on the Base Address DIP switch.

2.4 BOARD INSTALLATION

This section provides general instructions for installing the PCIP-SST Board. For more detailed information regarding installation of peripheral boards, consult the documentation provided with your computer.

WARNING

DO NOT ATTEMPT TO INSERT OR REMOVE ANY ADAPTER BOARD WITH THE COMPUTER POWER ON! THIS COULD CAUSE DAMAGE TO YOUR COMPUTER!

To install the PCIP-SST Board, proceed as follows:

1. Turn off power to the PC and to all attached options.
2. Unplug the power cords of all attached options from the electrical outlets. Make a note of where all the cables and cords are attached to the rear of the system unit and disconnect.

3. Remove the cover of the PC. First, remove the five cover mounting screws on the rear panel of the computer. Then, slide the cover of the computer about 3/4 of the way forward. Tilt the cover upwards to remove.
4. Choose an available option slot. Make sure that there is an additional empty slot to its right. Loosen and remove the screw at the top of the blank adapter plate. Then slide the plate up and out to remove.
5. Hold the PCIP-SST in one hand. With the other hand, touch any metallic part of the PC/AT cabinet to safely discharge any static electricity from your body.
6. Set the desired Base Address as described in Section 2.3.
7. Align the gold edge connector with the edge socket and the back adapter place with the adapter plate screw. Gently press the board downward into the socket. Re-install the adapter plate screw.
8. Replace the computer cover. Tilt the cover up and slide it onto the system's base, making sure the front of the cover is under the rail along the front of the frame. Install the mounting screws.
9. Turn on power to the computer.

You are now ready to make any necessary system connections and install the software.

2.5 SYSTEM CONNECTIONS

Make all connections to the PCIP-SST through four banana plugs located on the rear edge of the card. Two banana-to-BNC adapters are supplied with each board. These plug directly into the PCIP-SST once it has been installed in your computer. The top BNC connector will connect to the SST's function output, while the lower connector connects the Sync/Trigger output.

2.6 BACKING UP THE DISTRIBUTION SOFTWARE

As soon as possible, make a back-up copy of your Distribution Software. For the back-up copy, be sure to have one (or more,as needed) formatted diskettes on hand.

First, place the Distribution Software diskette in your PC's A Drive and log to that drive by typing **`A:`** . Then, use one of the following instructions to copy the diskette files.

Copying With A Single Floppy Drive

If your PC has just one diskette drive (Drive A), type **`COPY *.* B:`** (in a single-drive PC, Drive A also serves as Drive B) and follow the instructions on the screen.

If you prefer to use the DOS *DISKCOPY* function, instead of *COPY*, you will type **`DISKCOPY A: A:`** and follow instructions on the screen. Note that the *source* will be the diskette containing your Distribution Software; the *target* will be the diskette you will copy to. The DISKCOPY alternative is faster, but requires access to DISKCOPY.COM, in your DOS files.

Copying With Dual Floppy Drives

If your PC has two diskette drives (Drive A and Drive B), type **`COPY *.* B:`** (same as above) and follow the instructions on the screen.

If you prefer to use the DOS *DISKCOPY* function, instead of *COPY*, you will type **`DISKCOPY A: B:`** and follow instructions on the screen. This alternative is faster, but requires access to DISKCOPY.COM, in your DOS files.

Copying To A Hard Drive

If you prefer to back up your Distribution Software to a hard drive, simply insert the diskette(s) into the A Drive and copy the entire contents into a the hard-drive directory of your making or choice.

■ ■ ■

CHAPTER 3

DRIVER SETUP

3.1 DRIVER INSTALLATION

After the Distribution Software has been copied to your hard disk, you will need to modify the system configuration (CONFIG.SYS) file to include the VI.SYS and SST.SYS device drivers, if the PCIP-SST is to be loaded during power-up. VI.SYS and SST.SYS may also be loaded/unloaded only when needed. The VI.SYS file contains support files for the pop-up screen and the command handler. The SST.SYS contains system files for the PCIP-SST.

There are several ways of installing the device driver files, VI.SYS and SST.SYS. The CONFIG.SYS file can be modified using EDLIN or any other standard word processor, or, if you aren't sure how to modify a CONFIG.SYS file, you can use the program SSTSETUP.EXE provided on the PCIP-SST Utility Disk. Procedures for both techniques follow.

NOTE The ANSI.SYS driver is required with the VI.SYS driver. This is only to permit the use of the DOS CLS command. Be sure to install this driver in your CONFIG.SYS file, before the load of VI.SYS. The ANSI.SYS program is provided on your MS-DOS diskette.

Editing the CONFIG.SYS File

If you do not want to run the PCIP-SST Set-Up Program and are familiar with your CONFIG.SYS file, you may want to edit the existing CONFIG.SYS file with a standard word processor such as EDLIN. Start by bringing the CONFIG.SYS file into the editor.

STEP 1. Then, to install the VI.SYS device driver into the CONFIG.SYS file add the line

```
DEVICE = {PATH}VI.SYS {mono} /HK = x  /MK = m /SK = s
```

Where:

{MONO} Is an optional command that is used for computers with MonoChromatic displays. If {MONO} is not specified, COLOR is assumed.

HK designates the Help Key. Whenever the indicated key is pressed, the help screen will be displayed. (If the PCIP-SST is visible.) On-line help consists of a brief description of the PCIP-SST, a listing of current Key settings, and a listing of all commands which can be incorporated into your application programs.

x is the name of the key combination any of the legal shift keys (CTRL and/or ALT) plus 1 of the following character keys: A through Z, F1 through F10, 0 through 9, TAB, ESC, ?), spelled out. For example, "CTRL D" would indicate that the <Ctrl> and <D> keys must be pressed at the same time to bring up the help screen. The default setting for the help key parameter is ALT H.]

MK — designates the Mode Select Key. The POP-UP software can be operated either from the Keyboard, or a Mouse. Using the Mode Select key combination places the visible PCIP-SST into keyboard entry mode. Refer to the following section for more information.

m — is the name of the key combination any of the legal shift keys (CTRL and/or ALT) plus 1 of the following character keys: A through Z, F1 through F10, 0 through 9, TAB, ESC, ?). It must be spelled out, for example, "F1" assigns the function key F1 to be the Mode Select Key. The default Mode Select Key is <ALT M>.]

SK — defines the Instrument Select Key. If you have multiple MetraByte PCIP-SST or other PCIP boards installed in your computer, this key or key combination is used to toggle between the various instrument displays.

s — is the name of the key or key combination any of the legal shift keys (CTRL and/or ALT) plus 1 of the following character keys: A through Z, F1 through F10, 0 through 9, TAB, ESC, ?), spelled out. For example, "Alt I" would indicate that the <Alt> and <I> keys must be pressed simultaneously. The default Instrument Select Key is <Alt><Tab>.]

Example:

If this example line were to be placed in the CONFIG.SYS file, the default selections for: VI.SYS in the "ROOT" directory, Color Monitor, Help Key (<Alt><H>), Mode Select Key (<Alt><M>), and Instrument Select Key (<Alt><Tab>) are enabled.

```
DEVICE = C:VI.SYS /HK=ALT H /MK=ALT M /SK=ALT TAB
```

STEP 2. Then, to install the SST.SYS device driver add the line

```
DEVICE = {PATH}SST.SYS /PK=p  /BA=b
```

Note: This line must appear AFTER the line installing the VI.SYS, in the CONFIG.SYS file.

Where:

PK — designates the Pop-Up Menu Key. Whenever the indicated key or key combination is pressed, the instrument's Pop-Up Setup Menu will be displayed. Each type of instrument must be assigned a different Pop-Up Key.

p — is the name of the key combination. It must be spelled out. For example, "CTRL P" would indicate that the <Ctrl> and <P> keys must be pressed at the same time to bring up the help screen. The default setting for the Pop-Up Menu Key is <CTRL> <ESC>.]

BA defines the base address setting for the PCIP-SST. If this parameter is not given, the default base address of &H300 (768 decimal) will be used.

b is the value of the base address. Base address values may be given in either hex or decimal; however, if they are given in hex they must be preceded by an ampersand and an H (i.e., &H). Make certain that the base address you give has not been already assigned to another peripheral.]

CAUTION

Single keystrokes may be defined as Keys; however, it is suggested that only function keys be assigned in this manner. If the CONFIG.SYS file resides in the root directory, assigning a single keystroke will disable that key from being used in other software applications.]

STEP 3. Re-boot the PC. A screen similar to the one shown below should appear.

```
*******   VI.SYS loaded
    o Help Key is ALT H
    o Instrument Select Key is ALT TAB
    o Mode Select Key is ALT M

*******   SST.SYS loaded
    o Pop-Up Key is CTRL ESC
    o Base Address is 0300 hex
```

3.2 USING SSTSETUP.EXE TO MODIFY CONFIG.SYS

If you prefer not to modify CONFIG.SYS directly, you may use the PCIP-SST Setup Program. This is a self-explanatory program in the Distribution Software.

Before running PCIP-SST, be certain that the PCIP-SST is correctly installed. Note of the Base Address Switch setting; you will need this information later.

Then, log to the Distribution Software directory and type **SSTSETUP** .

The program will prompt you for the information required by the system driver (VI.SYS). Follow the instructions given in the following sections.

Selecting VI.SYS Options

Selecting The BOOT Drive

First, you will be asked to specify the drive where you want the CONFIG.SYS file to reside. Enter the appropriate Drive letter.

Selecting The PATH Name

You will be asked for the path name. This path name tells the CONFIG.SYS file the "directory" that contains the VI.SYS file. The following text represents a typical "PATH" set-up screen.

```
PCIP-SST Setup Program
Drive letter for CONFIG.SYS file : C
Full path name to VI.SYS:  C:\SST


Generate the path name to VI.SYS
The default path is ....  ( C:\ )
The path in the current CONFIG.SYS is ....  ( C:\SST\ )


Enter one of the following..

o  The complete path to VI.SYS followed by 'ENTER'
o  'ENTER' to use default path
o  'ESC' to exit Setup
```

Selecting the Monitor Type

Specify the type of monitor you are using. Type either **MONO** or **COLOR**.

Selecting the Help, Mode Select, and Instrument Select Keys

Whenever the Help Key is pressed and the instrument is visible, the help screen will be displayed. On-line help consists of a brief description of the PCIP-SST, a listing of current key settings, and a listing of all commands which can be incorporated into your application programs. The default setting for the help key is [Alt] [H].

The Mode Select Key is used to toggle between the mouse and keyboard setup modes. The default Mode Select Key is [Alt] [M].

If you have other PCIP boards installed in your computer, the Instrument Select Key is used to toggle between their setup menus. The currently selected instrument's menu will appear first in the sequence. The default Instrument Select Key is [Alt] [Tab].

Keys may be a single function key (F1 through F10) or a combination of keys (Ctrl or Alt with another valid key). Hot keys are specified by "spelling out" the key(s) used. For example,

Alt I means that the [Alt] and [I] keys must be pressed simultaneously.

F1 indicates the function key [F1].

Ctrl I is the [Ctrl] and [I] keys pressed at the same time.

If an invalid key or key combination is entered, the following error message will appear:

```
The Definition for the Key is not valid.
Strike any key to try another.
```

CAUTION

Single keystrokes may be defined as Keys; however, it is suggested that only function keys be assigned in this manner. If the CONFIG.SYS file resides in the root directory, assigning a single keystroke will disable that key from being used in other software applications.

Confirming the Selections

You will now be asked to confirm your selections. If you wish to change an entry, press [N] and the program will return to the Boot Drive Selection Screen. To return to DOS, press [Esc]. Otherwise, press [Enter]. A typical set-up screen is as follows:

```
PCIP-SST Setup Program
Drive letter for CONFIG.SYS file : C
Full path name to VI.SYS:  C:\SST

Declare type of monitor:  COLOR
HELP KEY definition: ALT ?
MODE SELECT KEY definition: F2
INSTRUMENT SELECT KEY definition: CTRL I

VI Setup CONFIRMATION
Enter one of the following..

o  'N' if above entries are NOT OK
o  'ENTER' to continue
o  'ESC' to exit setup
```

Setting up the SST.SYS Driver

Next, the program will ask you for the information required to setup the PCIP-SST driver. It first prompts you for the path name to the file SST.SYS. This will typically be the same path as used in the VI.SYS section.

Selecting the "HOT" Key

The program then asks you to select a Pop-Up Menu Key. Whenever the Pop-Up Menu Key is pressed, the instrument's Pop-Up Setup Menu will be displayed. It is suggested that each type of instrument be assigned a different Pop-Up Key. The default setting for the Pop-Up Menu Key is [Ctrl] [Esc].

Keys may be a single function key (F1 through F10) or a combination of keys (Ctrl or Alt with another valid key). Hot keys are specified by "spelling out" the key(s) used. For example:

Alt I means that the [Alt] and [I] keys must be pressed simultaneously.

F1 indicates the function key [F1].

Ctrl I is the [Ctrl] and [I] keys pressed at the same time.

If an invalid key or key combination is entered, the following error message will appear:

```
The Definition for the Hot Key is not valid.
Strike any key to try another.
```

CAUTION

Single keystrokes may be defined as Keys; however, it is suggested that only function keys be assigned in this manner. If the CONFIG.SYS file resides in the root directory, assigning a single keystroke will disable that key from being used in other software applications.

Selecting the Base Address

You now will be asked for the Base Address. Base Adress values may be given in either hex or decimal; however, if they are given in hex,they must be preceded by an ampersand and an H (&H). Make certain that the Bse Adress you give has not been already assigned to another peripheral and agrees with the Base Address sitch setting on the PCIP-SST. Selecting the Base Address, and setting the switches on the PCIP-SST is described in Capter 2.

Confirming the Selections

You will now be asked to confirm your selections. If you wish to change an entry, press [N] and the program will return to the screen requesting a path name for the PCIP-SST driver. To return to DOS, press [Esc]. Otherwise, press [Enter].

```
PCIP-SST Setup Program
Full path name to SST.SYS:  C:\SST
POP UP KEY definition: ALT P
Base Address for the PCIP-SST:&H300

PCIP-SST Setup CONFIRMATION
Enter one of the following..

o  'N' if above entries are NOT OK
o  'ENTER' to continue
o  'ESC' to exit setup
```

Backing Up CONFIG.SYS

The program will back-up your present CONFIG.SYS file to the file CONFIG.BAK and write the new set-up information to the CONFIG.SYS File. The CONFIG.SYS file is displayed. An example is given below. If your system already has a CONFIG.BAK file, you will be asked if it should be deleted.

```
Backing old CONFIG.SYS file to CONFIG.BAK
and creating new CONFIG.SYS

DEVICE = C:\SYS\ANSI.SYS
DEVICE = C:\VI.SYS /HK=ALT H  /MK=ALT M  /SK=ALT TAB
DEVICE = C:\SST.SYS  /PK= CTRL F1  /BA = &H310
files = 20
buffers = 10


Backing old CONFIG.SYS file to CONFIG.BAK
and creating new CONFIG.SYS

Backup file already exists
Delete it? (Y/N)

DEVICE = C:\SYS\ANSI.SYS
DEVICE = C:\VI.SYS /HK=ALT H  /MK=ALT M  /SK=ALT TAB
DEVICE = C:\SST.SYS  /PK= CTRL F1  /BA = &H310
files = 20
buffers = 10
```

In order to activate the CONFIG.SYS file that was created, press the [Ctrl] [Alt] [Del] keys simultaneously to reboot the system.

3.3 INSTALLATION VIA THE DRIVER FILE LOAD/UNLOAD OPTION

If you prefer not to edit your CONFIG.SYS file and not to keep your PCIP-SST driver files permanently loaded in memory, you have the option of using the driver-File Load/Unload programs. These programs allow you to load and unload driver files for all PCIP products from the DOS prompt (typically, using a batch file), on an as-needed basis. The Load program is contained in the file LS.COM,; the Unload program is in the file US.COM. Both files are included in your PCIP-SST software package.

Driver File Load Option: LS.COM

The LS.COM program permits you to load your PCIP-SST driver files into memory one-at-a-time by typing the appropriate command at the DOS prompt. When you use LS.COM, you may load your PCIP-SST driver files at any time without editing your CONFIG.SYS file and without having to reboot your computer.

Typing **LS** [Enter] at the DOS prompt summons the LS Help screen. The LS Help screen contains a brief description of the LS.COM function, the LS.COM syntax, the LS.COM switch options, and sample LS.COM commands. Explanations for syntax and switch options are as follows:

Syntax `LS [Path\Filename {/Option 1/Option 2 /Option n}`

In which

Filename is VI.SYS or SST.SYS.

VI.SYS Options /HK=key/key combination that designates the Help Key.
/MK=key/key combination that designates the Mode Select Key.
/SK=key/key combination that designates the Instrument Select Key.

SST.SYS Options /PK=key/key combination that designates the Pop-Up Menu Key.
/BA=value of Base Address as hex (&H) or decimal.

Note that if options are not specified, their defaults are used.

Examples
```
LS VI.SYS /HK=ALT H /MK=ALT M /SK=ALT TAB
LS C:\SST\VI.SYS
LS C:\SST\SST.SYS /PK=CTRL F6 /BA=768
LS VI.SYS
```

Driver File Unload Option:US.COM

The US.COM program permits you to unload your PCIP-SST driver files from memory one-at-a-time by typing the appropriate command at the DOS prompt. When you use US.COM, you can unload you PCIP-SST driver files at any time, without editing your CONFIG.SYS file, and without having to reboot your computer.

Typing **US** [**Enter**] at the DOS prompt summons the US Help screen. The US Help screen identifies the program, gives the syntax for using the program, and gives a sample command. The syntax for a US.COM command is as follows:

Syntax `US {Path\Filename}`

Where

Filename is VI.SYS or SST.SYS.

Examples
```
US C:\SST\VI.SYS
US C:\SST.SYS
```

Notes on Using the Driver File Load/Unload Option

- VI.SYS loads into memory in two portions: a resident portion and a transient portion. The resident portion occupies about 500 bytes of memory and stays resident until the computer is rebooted or turned Off. The transient portion can be freely installed, and it can be uninstalled when SST.SYS is not installed (after SST.SYS has been uninstalled).
- You are advised to load VI.SYS as soon as possible after power-up and then unload it if you do not need it. This action locates the resident portion as low as possible in memory, avoiding the creation of memory "holes". (DOS has no mechanism for recovering fragmented memory.)

- When LS.COM loads a .SYS file, it displays an *Amount of Memory* statement to indicated the memory space taken by the file. The memory space stated by *Amount of Memory* is sometimes larger than the actual .SYS file size because the file will take up extra memory to be used as its working space.
- LS.COM can install some other manufacturer's device drivers as long as VI.SYS is installed beforehand. MSMOUSE.SYS, for example, seems to work properly when loaded with LS.COM. Moreover, any other manufacturer's drivers that appear to work after being loaded by LS.COM may not unload without disrupting certain interrupt vectors.
- Installing two instrument drivers will not work if a board for only one of the drivers is installed. Each instrument driver is dedicated to one adapter card at one Base Address. Also, installing two instrument drivers at the same Base Address is likely to prevent proper operation of the drivers and boards.
- If you use LS.COM without naming a path to the specified driver file, LS.COM first tries to load the file from the current working directory. If the driver file is not in the current working directory, LS.COM then looks through the path specified in the AUTOEXEC.BAT file. LS.COM loads the first file it finds in the specified path, displaying the file name and path as well as the load destination.

3.4 PCIPMOD.EXE: DRIVER CREATION UTILITY

General

If you have two or more PCIP-SST's, you may program each device independently only by having a separate driver that is unique to each device. Or if you have two or more of another type of PCIP device (PCIP-SCOPE, PCIP-CAL, etc.) you may program each device independently only by having a separate driver unique to each device.

PCIPMOD.EXE is a utility program that enables you to create separate drivers unique to each of two or more PCIP devices of the same type. This program first copies the original driver and then steps you through procedures for changing the name of the new driver and changing the device show-name (the name which appears in the upper-left corner of the device pop-up menu). As a result, you can load drivers for two or more PCIP devices of the same type, call their pop-up menus to the screen simultaneously, and control each device independently.

Devices Supported by PCIPMOD.EXE

The following list names the PCIP device types supported by the PCIPMODE.EXE utility:

SHOWNAME	DEVICE NAME	DEVICE TYPE
PCIP-SST	$SST	Sine/Square/Triangle Wave Source
PCIP-DMM	$DMM	Digital Multimeter
DAS-50	$DAS50	High-Speed Data Acquisition Interface
PCIP-CNTR	$COUNT	Frequency Counter/Timer
PCIP-CAL	$CAL	Voltage Calibrator
PCIP-SCAN	$SCAN	Reed-Relay Multiplexer
PCIP-SCOPE	$SCOPE	Sampling Oscilloscope
PCIP-AWFG	$AWFG	Arbitrary Waveform Generator
PCIP-RES	$RES	Programmable Resistor Network

Using PCIPMOD.EXE

1. To use PCIPMOD.EXE, log to its directory and type **PCIPMOD**. The program opens with a description of its function, as follows:

   This program will modify the ORIGINAL PCIP Driver's
   a. Name that is to be used to Open the device.
   b. Name that is displayed in the upper left corner of the instrument when the instrument is visible.

   Enter the name of the ORIGINAL driver to be modified. Please include the entire path name and drive where the device can now be found. For example... C:\SST\SST.SYS

2. To use PCIPMOD.EXE, log to its directory (entitled SST); your entry of a device will be **C:\SST\SST.SYS**.

3. The program now requests the name you wish to assign the driver being created, as follows:

   Enter the Name to be used for opening the device

   Use any name with up to eight letters. For example, if you are naming one of two PCIP-SST's you may wish to name the second $SST2. You therefore type **$SST2**.

4. The program next asks for the name that appears in the upper-left corner of the pop-up menu for the ORIGINAL driver, as follows:

   the current name used for the instrument popup is

   For your PCIP-SST, you type **PCIP-SST**.

5. Now the program asks for the name to be used with the driver being created as follows:

   Enter the new name to be used. This name can not be longer than 9 characters.

   For consistency with the device opening name of $SST2, you could choose PCIP-SST2 for the pop-up menu. You therefore type **PCIP-SST2**.

7. The program finally asks for the name and path to be used for the new driver. Using the choices you have made, type **C:\SST\SST2.SYS**. This completes the driver creation and naming process using PCIPMOD.EXE.

PCIPMOD.EXE Usage Notes

- If you have purchased two or more PCIP instruments of the same type, run PCIPMOD.EXE to create a unique driver for each so that you can program for independent operation.
- Loading the same PCIP device twice installs two devices whose names are identical (For example $SST and $SST). In this situation, DOS is unable to tell you which one you want to program. You should use PCIPMOD.EXE to create a unique driver for each device to be loaded.
- Using PCIPMOD.EXE allows you to easily distinguish between pop-up menus being displayed simultaneously for two or more PCIP instruments of the same type, because the program allows you to use a different show-name (the name appearing in the upper-left corner) for each menu.

- When you load two or more PCIP drivers for the same device type, you are urged to specify a different hot key for each menu so that you can bring menus to the screen simultaneously.

3.5 START-UP PREPARATIONS

You operate PCIP-SST either by manual control or by a software control program. If you use manual control, you conduct your operation solely via the keyboard/mouse entries (see Chapter 4). If you use a software control program, you rely on a program of commands to conduct your operations; you write the program according to instructions described n Chapter 5.

If you are a first-time user of PCIP-SST, you are advised to begin with manual control. Manual control is the fastest and easiest way to gain familiarity with the PCIP-SST.

■ ■ ■

CHAPTER 4

THE POP-UP DISPLAY

4.1 GENERAL

Once the PCIP-SST is installed you can "POP-UP" the display by typing in the "HOT" key sequence. The default HOT key sequence is <CTRL> ESC. To bring the display up on the screen simply hold down the <CTRL> key and then hit the ESC (escape) key. This should bring the PCIP-SST display up on the screen. To remove the PCIP-SST simply hit the same <CTRL> ESC sequence.

In this chapter there will be a number of multiple key, keyboard entries described (such as the [Ctrl] [Esc] sequence mentioned in the previous paragraph). For purposes of this chapter we will use the following conventions. A key inside [ ] signs signifies that the key should be held down for the entire sequence. A key that is not in brackets is pressed and released. For example, the keyboard entry [Alt] [H] says that the Alt key should be depressed and held down; the H key should be depressed and released; and then the Alt key should be Released.

4.2 KEYBOARD/MOUSE ENTRY MODES

Once the PCIP-SST is "POPPED UP" there are two methods of setting the operating parameters (Output Frequency or Amplitude). The KEYBOARD mode allows the PCIP-SST to be controlled directly from the computer's Keyboard, while the MOUSE mode allows the SST's parameters to be set via a Mouse.

Mouse Mode

If your computer is equipped with a mouse, the mouse may be used to set the various operating parameters of the PCIP-SST board. Once the SST's display is popped up, the mouse will be enabled. As you move the mouse around you will see the rectangular "cursor" also moving. Selections are made by placing the cursor on the display such that it is touching the parameter you wish to change. All mouse operations are performed immediately. Any change made to the displays are automatically written to the PCIP-SST board. There is no need for any special [Enter] command.

NOTE that even in the "MOUSE" mode of operation, the on-screen help function is controlled via a keyboard command. Whenever the SST screen is being displayed, the help function can be run. Notice that in the upper right hand side of the PCIP-SST's display there is the word HELP, followed by a key sequence (default is HELP(ALT H)). To get to the HELP menu simply type the key sequence shown in the parenthesis following the word HELP. The default help sequence is [Alt] [H].

Setting Keyboard/Mouse Mode

The keyboard entry mode can be useful even in systems that have a mouse. For this reason the Keyboard mode can be selected via the mouse. On the very top line of the instrument display there are the words "Keyboard Entry." To select the Keyboard entry mode simply place the cursor on the word Keyboard, and press either of the mouse input keys.

Track

The TRACK command forces the PCIP-SST to perform a specific frequency calibration at the current setting. When the Track command is enacted the actual output frequency is monitored by the on-board frequency counter, and the output frequency is adjusted to bring the output frequency as close to the desired frequency as possible. To start a track operation place the cursor on the word TRACK (on the top line of the display) and press either mouse key. Issuing the TRACK command performs a single track function. If the frequency is changed, the TRACK command must be re-issued.

Calibrate

The PCIP-SST has a built-in self calibration function. It is recommended that the calibration function be run each time the computer is turned on (only if the PCIP-SST is to be used). For best results the calibration should be run whenever the PCIP-SST board will be exposed to a significant temperature change (as occurs when the computer is turned on). The calibration takes approximately 7 seconds. In addition the overall calibration of the board can be monitored by checking that the measured frequency is within 3% of the input frequency.

To force a board calibration put the cursor on the word Calibrate (on the top line of the PCIP-SST display), and press either mouse key.

The Meter Display Mode

The PCIP-SST includes an on-board frequency counter. This counter is constantly measuring the actual output frequency of the board. The Meter option allows this frequency measurement to be displayed on the screen. To display the measured frequency put the cursor on the word Meter (on the top line of the PCIP-SST display) and press either mouse key.

Setting the Frequency

To set the output frequency simply place the mouse cursor on each of the large numbers on the left hand side of the display. Once the cursor is touching one of the numbers, you can increment the number by hitting the left key of the mouse, or decrement the number by hitting the right key. Numbers will roll over (e.g. incrementing a 9 gives a 0) but no carry operations are performed. If an illegal value is entered, the frequency display will begin to flash on and off.

The Decimal point can be moved by placing the cursor at the desired decimal point location, and hitting either mouse key.]

Setting the Frequency Range

To set the frequency range, place the mouse cursor inside the box immediately to the right of the frequency. (This box will display Mhz, khz or Hz.) Press the mouse's left key if you want to increase the frequency or the right key to decrease the frequency.

Selecting the Output Waveform

In the center of the display is a large box with SINE, TRIANGLE, and SQUARE shown. To select the waveform simply put the mouse's cursor on the waveform desired and press either mouse key. The current selection will shown in a "REVERSED" typeface.

Setting the Output Amplitude

The output amplitude is displayed in the box labeled "Ampl Vpp." All amplitude selections are in the units of Volts, Peak to Peak. The amplitude display is in a fixed NN.NN volts format. To change the amplitude place the cursor on the digit you wish to change. Hit the left key of the mouse to decrease the value or the right key to increase the value. Numbers will roll over (e.g. incrementing a 9 gives a 0) but no carry operations are performed. If an illegal value is entered, the amplitude display will begin to flash on and off. Amplitude values must be less than or equal to 20.00 Volts.

Setting the DC Offset

The output DC offset is displayed in the box labeled "Offset." All amplitude selections are in the units of Volts, Peak to Peak. The offset display is in a fixed ±N.NN volts format. To change the amplitude place the cursor on the digit you wish to change. Hit the left key of the mouse to decrease the value or the right key to increase the value. To change the sign of the offset place the cursor over the current sign and hit either mouse key. Numbers will roll over (e.g. incrementing a 9 gives a 0) but no carry operations are performed. If an illegal value is entered, the offset display will begin to flash on and off. DC offsets must be set such that -5.00V $\leq$ OFFSET $\leq$ +5.00 Volt.

Note: The Total output of the PCIP-SST cannot go beyond ±10 volts. For proper operation note that

$$(0.5 * (\text{Amplitude})) + \text{Offset} \leq 10 \text{ volts}$$

The board will **not** be damaged by exceeding these limits. However if these limits are exceeded either the top or the bottom of the waveform will "clipped" off.

4.3 KEYBOARD MODE

As an alternative to the MOUSE input mode, all PCIP-SST parameters can be selected via the computer's keyboard. Once the PCIP-SST display is on the screen, the Keyboard mode can be selected by entering a **[Alt] [M]**. (Note that **[Alt] [M]** is the default value. If another key sequence has been selected use it instead.)

Immediately upon hitting the **[Alt] [M]** command, the software will enter the Keyboard Entry Mode. The Keyboard entry mode lets you change one parameter or figure at a time. The parameter that is currently being modified is always flashing, and in Reversed typeface. In addition, the bottom line of the display screen will contain HELP comments that describe what the legal keystrokes for that parameter are.

The actual functions of each of the blocks is explained in the following sections.

Track

The TRACK command forces the PCIP-SST to perform a specific frequency calibration at the current setting. When the Track command is enacted the actual output frequency is monitored by the on-board frequency counter, and the output frequency is adjusted to bring the output frequency as close to the desired frequency as possible. To start a track operation place the cursor on the word TRACK (on the top line of the display) and press the [Enter] key.

Calibrate

The PCIP-SST has a built-in self calibration function. It is recommended that the calibration function be run each time the computer is turned on (only if the PCIP-SST is to be used). For best results the calibration should be run whenever the PCIP-SST board will be exposed to a significant temperature change (as occurs when the computer is turned on). The calibration takes approximately 7 seconds. In addition the overall calibration of the board can be monitored by checking that the measured frequency is within 3% of the input frequency.

To force a board calibration put the cursor on the word Calibrate (on the top line of the PCIP-SST display), and press [Enter].

The Meter Display Mode

The PCIP-SST includes an on-board frequency counter. This counter is constantly measuring the actual output frequency of the board. The Meter option allows this frequency measurement to be displayed on the screen. To display the measured frequency put the cursor on the word Meter (on the top line of the PCIP-SST display) and press [Enter].

Setting the Frequency, Amplitude and Offset

To set the outputs of the PCIP-SST from the keyboard, first put the board in Keyboard Entry Mode (typically with an [Alt] [M] command). Then use the following procedure to set up the board.

1. Use the right/left arrow keys to move the cursor such that the words "Keyboard Entry" are flashing and in reverse video.
2. Hit the ENTER key.
3. The units (Hz, khz or Mhz) display will be flashing and reversed. Use the Up or Down arrow key to change the display to the desired units, and hit [Enter].
4. The Frequency digits will then be in reverse video. From the Keyboard type the desired output frequency. (You will need to type all four digits. The decimal point will always stay between the Most and second most significant digits.)
5. The Frequency digits should return to there regular color, and one of the Waveform selections will now be flashing and reversed. Use the Up and Down arrow keys to select the desired waveform type and hit [Enter].
6. The Amplitude digits will then be in reverse video. From the Keyboard type the desired output amplitude (in VOLTS PEAK TO PEAK). (You will need to type all four digits. The decimal point will always stay between the Second most and third most significant digits.)

7. The Offset digits will then be in reverse video. From the Keyboard type the desired DC offset. (You will need to type all three digits. The decimal point will always stay between the Most and second most significant digits.)
8. The board is complete set, hit the ESC key twice to leave the Keyboard mode.

Note: The Total output of the PCIP-SST cannot go beyond +/- 10 volts. For proper operation, note that

$(0.5 * (\text{Amplitude})) + \text{Offset} \leq 10 \text{ volts}$

The board will @p[not] be damaged by exceeding these limits. However if these limits are exceeded either the top or the bottom of the waveform will "clipped" off.

■ ■ ■

CHAPTER 5

PROGRAMMING

5.1 GENERAL

This chapter describes how to program the PCIP-SST. The chapter begins by describing the Independent Language Interface. A series of ASCII commands allows you to control the operation of the PCIP-SST from a program written in any of several languages. Each of the commands is described in detail. The description includes a synopsis of the command's function, its syntax, and a few examples illustrating its use.

The chapter also contains a short procedure for incorporating the commands into a program. Two sample programs, written in C and BASIC, are given. In addition, Appendix A provides example programs and a few helpful hints for users writing programs in Pascal, C, Turbo Pascal, and Turbo C.

For those who require more specific control over the PCIP-SST, register descriptions are also given. It should be noted, however, that these are only an outline of the registers' functions. We highly recommend that all users make use of the PCIP-SSTs device driver.

5.2 PCIP-SST DEVICE DRIVER

The PCIP-SST is provided with a language independent interface. To open the interface file, use the standard file open command for the language you are using and the filename $SST . Once the device file has been opened, commands are passed to the device in ASCII strings using whatever print or write instructions are available with the language. For example in BASIC commands are written to the board via the PRINT command, and read from the board with the INPUT command.

An example of opening the language interface file, and performing reads and writes in BASIC is provided under *Example 1* . *Example 2* provides a similar example using C. Appendix A also includes some helpful hints on how to Open, Write to, Read from, Trap errors, and close the Virtual instrument drivers from Pascal, C, Turbo-Pascal and Turbo-C.

Example 1:

```
10 CLS
20 ON ERROR GOTO 330
30 OPEN "$SST" FOR OUTPUT AS #1
40 PRINT #1, "CLEAR"
50 OPEN "$SST" FOR INPUT AS #2
60 '
70 'Clear area on screen for command
80 '
90      LOCATE 11,1:PRINT"                          "
100     LOCATE 12,1:PRINT"                          "
110     LOCATE 13,1:PRINT"                          "
120  LOCATE 11,1
130     '
140  INPUT "Enter command ", A$
150  '
```

```
160  'determine if it's a read command
170     C$ = MID$(A$,1,2)
180     IF (C$ = "RE" OR C$ = "Re" OR C$ = "rE" OR C$ = "re") THEN MODE=1
ELSE MODE=0
190  '
200  PRINT #1,A$
210  IF MODE=0 THEN GOTO 90
220  '
230  'if READ then retrieve data
240  '
250  INPUT #2,A$
260  PRINT "The Value returned was...", A$
270  PRINT "Strike any key to continue"
280  IF INKEY$="" GOTO 280
290  GOTO 90
300  '
310  'Error Handler
320  '
330  BEEP
340  PRINT ERR
350  INPUT #2,EN: LINE INPUT #2, A$: LINE INPUT #2, B$
360  PRINT "SST Error number ";EN
370  PRINT "ERROR - ",B$
380  PRINT "On command line of ...";A$
390  PRINT "Strike any key to continue"
400  IF INKEY$="" GOTO 400
410  CLS
420  RESUME 90
```

Example 2:

```
FILE *VI
char *ErrMess1[200], ErrMess2;
int  ErrNum;

main();
{
if ((VI = fopen("$SST","r+"))==NULL)
   printf("Could not find device PCIP-SST");
if (!(fputs("Set FR 1234 AMPL 1.5 OFF 3.2",VI)GetError;

...
...
...

}
void  GetError()
\\* retrieve error number and error message from device*\\
{
rewind(VI);
fscanf(VI,"%d",ErrNum):                \\* get error number\\
fgets(ErrMess1,80,VI);                 \\* get error message 1\\
fgets(ErrMess2,80,VI);                 \\* get error message 2\\
rewind(VI);
printf("ERROR Number %d", ErrNum);
printf("ERROR - %s", ErrMess2);
printf("On command line of ...%s", ErrMess1);
}
```

The PCIP-SST Command Structure

Once the Device Driver has been opened, reading and writing to the PCIP-SST is performed by a number of simple ASCII commands. There are three types of commands available:

SET commands are for setting output wave type, frequency, amplitude, and DC offset. Examples of valid SET commands include:

`SET FORM SINE`	This command selects a sine wave output.
`SET frequency 12340`	This command sets the output Frequency to 12,340 Hz.
`SET amplitude 2.6`	Sets the output amplitude to 2.6 Volts, Peak-to-Peak.

READ commands are used to read data from the PCIP-SST. Examples of valid READ commands include the following:

`READ frequency`	Reads current measured frequency.
`READ amplitude`	Reads current amplitude setting.

After the READ command is sent, the value or data to be read is retrieved via an INPUT command (in BASIC), or a standard FILE read command in other languages.

Single Word commands are used for a variety of set-up and control applications. Examples of these include the SHOW command which POPS-UP the PCIP-SST display, the HIDE command which removes the SST's display and the CALIBRATE command which forces a board calibration.

List Of Commands

The following list displays each of the available PCIP-SST commands.

CAlibrate	REad OFfset
CLear	{SEt} {FOrm} [SIne, TRiangle, SQuare]
HIde	{SEt} FRequency [F]
LOck	{SEt} AMPLITUDE [A]
MEter [ON, OFF]	{SEt} OFfset [O]
REad {FOrm}	SHow
REad FRequency	TRack
REad AMPLITUDE	UNlock

Nomenclature Rules

This section presents nomenclature rules for the command descriptions.

1. Anything appearing in curly brackets, (i.e, {}) is optional. Don't enter the curly brackets.
2. Anything appearing in square brackets indicates a mandatory choice. The square brackets should not be entered.
3. Uppercase defines the command's key letters which are the minimum set to be used. In your commands you may use upper case, lower case, or a combination of both, case is ignored. When entering commands you must use the first two letters. You may optionally go on to spell the entire word, or may just spell out the next few letters. However, if you do continue to spell out the word, any misspellings, or abbreviations other than just dropping off the last few letters will cause an error.
4. Variables denoting parameters will be in italics.
5. Multiple commands may appear on one command line.
6. The following are examples of legal and illegal input lines.

Legal:

Set Freq 500
Set AMPL 2.5
Set Freq 500 Ampl 2.5 Commands may be combined within a single command.

LOck Set FRequency 4000
lock SET FREQUENCY 4000 Note that case does not matter.

Set Frequency 5000
Set FR 5000 Note that only the first two letters are required for abbreviation.

Illegal:

Set FRequesy 300 Words must be correctly spelled
Set Freq 6e6 6e6 is not a valid frequency variable.

Command Descriptions

CAlibrate

The CAlibrate command forces the board to perform an immediate calibration. This calibration is performed by comparing the output frequency with a known on-board crystal oscillator. The calibration will take approximately 7 seconds. **NOTE:** The calibrate mode only effects the output frequency. It does not change either the amplitude or offset settings. These can only be calibrated via the STEST.EXE program provided on the PCIP-SST utility disk. Examples of the use of the Calibrate command are:

CAlibrate
CaLiBr
CA
ca

CLear

The CLEAR command is used to clear the input string buffer of the driver. It is recommended that a CLEAR command should be sent after opening the SST driver for output. Examples of valid clear commands are:

```
CLEAR
clear
CL
Cl
```

HIde

The HIDE command removes the PCIP-SST display from the computer screen. (The HIDE command is the opposite of the SHOW command). Examples of valid HIDE commands are:

```
HIDE
HIde
hi
HI
```

LOck

When the LOCK command is executed, the parameters of the PCIP-SST can no longer be controlled via the mouse or keyboard. (See also the UNLOCK command). Examples of using the LOCK command are:

```
LOCK
LoCk
LO
loc
```

MEter [ON, OFF]

The PCIP-SST is equipped with a built-in frequency counter. This frequency counter provides a high accuracy (+/- 0.2%) measurement of the actual output frequency. Though the counter is always running, the Meter command allows the user to select whether this counter is displayed on the PCIP-SST screen or not. Examples of using the meter command are:

```
Meter On
METER OFF
ME on
met off
```

REad FOrm

The READ FORM command allows the program to read the type waveform currently selected. The command returns (through an INPUT instruction) either a 0 (for Sine), a 1 (for triangle) or a 2 (for Square wave). Examples of the READ FORM command are shown below:

```
RE FO
READ FO
re form
```

REad FRequency

The READ FREQUENCY command allows the program (through an INPUT instruction) to actually measure the output frequency with the on-board frequency counter. Examples of this command are:

```
READ FREQ
read fr
Re Fr
```

REad AMplitude

The READ AMPLITUDE command allows (through an INPUT instruction) the program to read the output amplitude that is currently selected. Examples of the READ AMPLITUDE command are:

```
read amplitude
re am
read ampl
```

REad OFfset

The READ OFFSET command allows the program to read (through an INPUT instruction) the current DC offset. Examples of the READ OFFSET command are:

```
READ OFF
read offset
re of
RE of
```

{SEt} {FOrm} [SIne, TRiangle, SQuare]

The SET FORM command is used to select the type of output waveform. Note that {SET} is an optional word. Valid examples of using the SET FORM command are:

```
Set Form Sine
SEt FORM TRI
form square
fo Tr
Se Fo Sq
TR
sq
```

{SEt} FRequency F

The SET FREQUENCY command allows the output frequency to be set. The variable F is in the units of Hertz (HZ) or cycles per second, and must be between 1 (Values between 0 and 1 are automatically set to 1 Hz) and 5e6. Acceptable examples of setting the frequency are shown below:

```
Set Freq 400.1
Set Freq 4e6
set freq 4000000
set freq 38.7
Set Frequency 500
SE FR 3000
FR 3e3
```

{SEt} AMplitude A

The Set AMPLITUDE command is used to set the output amplitude. The variable A is an amplitude between 0 and 20. @B[Note:] The Total output of the PCIP-SST cannot go beyond +/- 10 volts. For proper operation note that

$$(0.5 * (\text{Amplitude})) + \text{Offset} \leq 10 \text{ volts}$$

Examples of valid SET AMPLITUDE commands are shown below:

```
Set Amplitude 3.65
SE AMPL 2.456
SE AM 0.567
SEt AMPL 3.45
AMPL 3.45
am 2.345
```

{SEt} OFfset O

The SET OFFSET command sets the DC offset of the output waveform. The offset voltage (represented by the variable 0) can be from -5 V to +5 V. **Note:** The Total output of the PCIP-SST cannot go beyond +/- 10 volts. For proper operation note that

$$(0.5 * (\text{Amplitude})) + \text{Offset} \leq 10 \text{ volts}$$

Examples of valid SET OFFSET commands are shown below:

```
Set OFFSET 2.34
Set Off -4.33
set off -.004
se off 3.43
OFFset - 3
OF +2.45
```

SHow

The SHOW command forces the PCIP-SST's display to appear on the screen. When in the SHOW mode, the instrument display will always show the actual or current values. As various parameters are changed by the software the screen will automatically be updated. The SHOW command is the opposite of the HIDE command. Examples of valid Show commands are:

```
SHOW
SH
show
```

TRack

The TRACK command forces the PCIP-SST to perform a specific frequency calibration at the current setting. When the Track command is enacted the actual output frequency is monitored by the on-board frequency counter, and the output frequency is adjusted to bring the output frequency as close to the desired frequency as possible. Track is a one-time operation. If the frequency is modified, the Track command must be re-issued to Track the new frequency. Examples of the TRACK command are:

```
Track
track
trA
Tr
```

UNLock

When the UNLOCK command is executed, the parameters of the PCIP-SST screen can be changed via the mouse or keyboard. (See also the LOCK command). Examples of the UNLOCK command are:

```
UNlock
UNL
unl
```

Associated Error Messages

151 Unknown character encountered
152 Space expected
153 Illegal function requested
154 Bad or illegal FREQUENCY
155 Bad or illegal OFFSET
156 Bad or illegal AMPLITUDE
157 Illegal word following METER

Other Programming Notes

The VI.SYS driver maintains all the instruments' visual characteristics. In doing this, the VI.SYS prevents any screen writes to the instrument area by intercepting and interpreting all INT 10 h calls that would result in affecting the visible instruments. Any program that uses DOS or INT 10's to

display information will work without interference to the visible instruments. If, however, a program is being used that performs direct screen I/O (i.e., writes directly to the display's memory) it is impossible to prevent the visible screen from being overwritten, one may redraw it by doing a pop-down followed by a pop-up. The instrument itself will be unaffected.

Most BASICs will have no problems, although it has been noted that some basics do direct screen I/O when scrolling the screen up. If this is the case, one could pop the instrument down before the scroll would occur.

MS QuickC, will not change to the user's screen unless there has been a write to the screen. If the user programs the instrument to pop-up, using the SHOW command, before any normal screen I/O has occurred, the instrument will pop-up over QuickC'S SCREEN and not the user's screen.

BASIC files Specification

If BASICA is loaded without any command line switches, there will be only 3 files permitted to be open at one time. Since the instrument require two files to be open (1 for output and 1 for input) it may be necessary for your program to have more than 3 files. To do this, enter:

`BASIC \fn` where n is the number of files (up to 16).]

Microsoft C

If the PCIP-SST is opened for reading and writing as below:

```
FILE *SST;
SST = fopen( "$SST", "r+");
```

It is important to execute a rewind command between an input and output or an output and input command to the PCIP-SST. In addition, in Microsoft C, it is necessary to issue an **fflush ( )** command after the **fprint( )** to ensure that the command is flushed from DOS' buffer.

5.3 REGISTER DESCRIPTIONS

MetraByte strongly recommends that all programmers wishing the use the PCIP-SST make use of the SST.SYS driver that is provided with the board. However if for some reason the user wishes to program the board directly via the I/O bus, we provide the following description of the I/O registers and their functions. The following diagram shows the overall I/O register mapping of the PCIP-SST.

ADDRESS	READ	WRITE
BASE +0	FREQ COUNT 0	FREQ COUNT 0
BASE +1	FREQ COUNT 1	FREQ COUNT 1
BASE +2	FREQ COUNT 2	FREQ COUNT 2
BASE +3	------	8254 CONTROL
BASE +4	STATUS	CONTROL
BASE +5	------	OFFSET / GAIN
BASE +6	------	HIGH BYTE FREQ
BASE +7	------	LOW BYTE FREQ

Frequency Measurements

Base +0 through base +3 correspond to registers within an 8254 programmable counter timer. The three counters are used to perform the actual frequency measurements. Counter 0's input is connected directly to the output frequency of the board. Counter 2's input is connected directly to a precision 10 MHz crystal controlled oscillator. Both Counters 0 and 2 are used as pre-scalers for counter 1. There are two modes of counter operation. Which is used is dependant on the input frequency.

Frequency Mode

At frequencies above 1 MHz, the Frequency mode is used. In this mode the scaled output of Counter 0 (the unknown frequency) is connected to the CLOCK INput of Counter 1. The scaled output of Counter 2 (the 10 MHz prescaler) is then connected to Counter 1's gate. In this mode we count the number of pulses generated by our unknown frequency for a fixed period of time set by our 10 Mhz prescaler. A block diagram of this is shown in the following diagram.

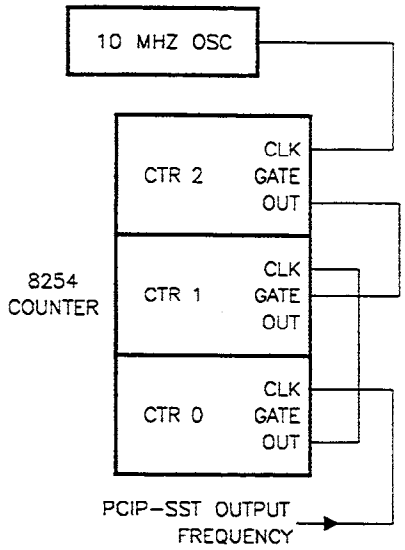


Figure 5-1. Frequency Mode Counter Configuration

Period Mode

At frequencies less than 1 Mhz the frequency measurement is performed in a Period measurement mode. In this case the roles of the 10 MHz input and the PCIP-SST output frequency are reversed. A scaled version of the 10 MHz signal is connected to the clock input of counter 1, while a prescaled version of the unknown is connected to the counter 1 GATE. This configuration is shown in figure 5-5.]@group[

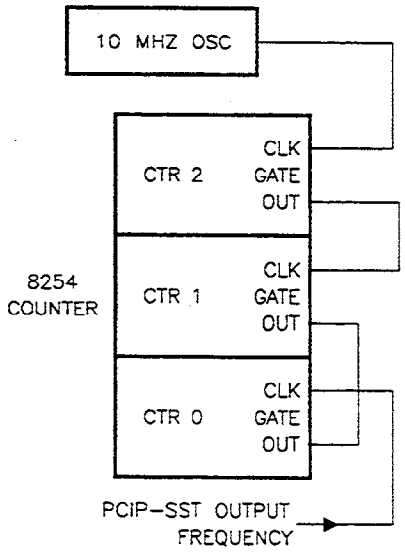


Figure 5-2. Period Mode Counter Configuration

The actual mode selected is set by bit 0 of the BASE ADDRESS + 4, Control Register. A 0 selects period measurement, while a 1 selects frequency measurement.

In addition, Bit 1 of the BASE + 4 Control Register is used as an enable/disable control bit. When Bit 1 is 0, the counters are enabled, when Bit 1 is 1, the counters are disabled. Also, Bit 7 of the BASE ADDRESS + 4 Status Register tells whether a frequency measurement is in progress, or complete. With this information it should be possible to write a frequency measurement routine for the PCIP-SST. However, it will be a very complex programming task, and once again we urge you to use the device driver software before undertaking this task.

Register Descriptions

BASE ADDRESS + 0

Frequency Counter 0 of the 8254 counter. Counter 0 provides a 16 bit programmable divider for the frequency output of PCIP-SST

BASE ADDRESS + 1

Frequency Counter 1 of the 8254 counter. Counter 1 provides a 16 bit counter that is used to accumulate either clock ticks or waveform cycles depending on mode of operation (frequency or period). For further details please refer to the previous section on frequency measurement.

BASE ADDRESS + 2

Frequency Counter 2 of the 8254 counter. Counter 2 provides a 16-bit divider/prescaler for the 10 MHz clock output.

BASE ADDRESS + 3

The 8254 Control register.

BASE ADDRESS + 4

Control/Status register.

STATUS — Bit 7 is used as measurement cycle indicator. When Bit 7 = 1, the frequency measurement is done and data is available in the frequency counter (at BASE + 1). When Bit 7 = 0, the frequency measurement is in process.

Bit 0-6 not used.

CONTROL — Bits 7 and 6 are used to select the waveform type.

BITS 7 6
- 0 0 – TRIANGLE WAVE
- 0 1 – SQUARE WAVE
- 1 0 – SINE WAVE

Bits 5 and 4 are used to select the output frequency range.

BITS 5 4
- 0 0 –RANGE 1, 1 HZ TO 100 HZ
- 0 1 –RANGE 2, 100 HZ TO 10 kHZ
- 1 0 –RANGE 3, 10 kHZ TO 500 kHZ
- 1 1 –RANGE 4, 500 kHZ TO 5 MHZ

Bits 3 and 2 are not used.

Bit 1 is used to enable/disable the frequency measurement circuit.

BIT 1 = 0 measurement circuit enabled
BIT 1 = 1 measurement circuit disabled

Bit 0 is used to select period or frequency measurement.

BIT 0 == 1 FREQUENCY MEASUREMENT
BIT 0 == 0 PERIOD MEASUREMENT

BASE ADDRESS + 5

OFFSET/GAIN REGISTER

BITS 0 1 2 3 data bits for offset and gain values

BITS 6 5 4	address pointer bits
0 0 0	LOW NIBBLE gain value
0 0 1	MID NIBBLE gain value
0 1 0	HIGH NIBBLE gain value
0 1 1	12 BIT gain value loaded
1 0 0	LOW NIBBLE offset value
1 0 1	MID NIBBLE offset value
1 1 0	HIGH NIBBLE offset value
1 1 1	12 BIT offset value loaded

Bit 7 – write bit: must be toggled TO send each nibbled across opto barrier. Data is loaded on the low to high transition of bit 7

BASE ADDRESS + 6

The output frequency is controlled by a 12-bit D/A converter. Writing to the Base Address + 6 writes the 8 Most Significant Bits of this D/A. **NOTE:** There is no fixed scale factor converting D/A output to frequency. The scale factor to use must be determined by performing a number of tests based on writing various 12-bit values to the D/A, and then measuring the corresponding output frequency. When using the PCIP-SST device driver, this is done automatically.

BASE ADDRESS + 7

Base address + 7 writes the low nibble of 12 bit frequency control D/A converter. The data is in bits 0 through 3, while bits 4 through 7 are Unused.

■ ■ ■

CHAPTER 6

CALIBRATION & REPAIRS

6.1 GENERAL

This chapter provides instructions for calibrating the PCIP-SST. The return to factory procedure is also discussed in the event that your unit should require repair.

6.2 CALIBRATION

The Distribution Software contains the program STEST.EXE, a user-friendly calibration and test program. The STEST.EXE program leads you through each step of the calibration procedure.

1. Allow the PCIP-SST to warm-up for at least 1 hour.
2. To invoke the Calibration Program, at the DOS prompt, type **STEST**.

6.3 WARRANTY INFORMATION

All products manufactured by Keithley MetraByte are warranted against defective materials and workmanship for a period of one year from the date of delivery to the original purchaser. Any product that is found to be defective within the warranty period will, at the option of Keithley MetraByte, be repaired or replaced. This warranty does not apply to products damaged by improper use.

WARNING

Keithley MetraByte Corporation assumes no liability to damages consequent to the use of this product. This product is not designed with components of a level of reliability suitable for use in life support or critical applications.

6.4 RETURN-TO-FACTORY INFORMATION

Before returning any equipment to the factory for repair, you must first call the Technical Support Department at (508) 880-3000. They will try to diagnose and solve your problem over the phone. If they ascertain that the unit needs to be returned to the factory for repair, they will issue a Return Material Authorization (RMA) number. Note that if the board is to be repaired under warranty, the Technical Support Department will need your invoice number and the date the board was purchased. Please reference the RMA number on any correspondence regarding the board.

When returning the Board for repair, please include the following information:

1. A brief description of the problem.

2. Your name, address, and telephone number.
3. The invoice number and the date when the board was purchased.
4. Repackage the board in its original anti-static wrapping (and handle it with ground protection) and ship it back to:

   Repair Department
   MetraByte Corporation
   440 Myles Standish Boulevard
   Taunton, Massachusetts 02780

   FAX:(508)880-0179
   Telephone:(508)880-3000
   Telex:503989

Be sure to reference your RMA number on the outside of the package!

■ ■ ■

APPENDIX A

PROGRAMMING HINTS

This section provides a few programming hints for using the PCIP-DMM device driver with high-level languages such as Microsoft C (4.0 and higher), TURBO C (1.0, 1.5), Microsoft PASCAL (3.0. 3.3 and 4.0), and TURBO PASCAL (4.0). Each section gives specifics regarding opening the device, I/O operations, and handling error strings. An example program is given for each language.

A.1 MICROSOFT C (VERSIONS 4.0, 5.0, 5.1)

Keep the following rules in mind when creating application programs in Microsoft C.

Opening a Device

Use the **fopen();** function with the **r+** file mode parameter.

I/O Operations

1. In versions 5.0 and higher, before switching between read and write operations, call the **Rewind();** function.
2. Strings are OUTPUT using the **fprintf();** function.
3. Use the **fgets();** to INPUT strings.
4. After each **fprintf();**, be sure to call **fflush()**. This will flush the output buffer and write an output string immediately. Otherwise, the output strings will be buffered and the device may fail due to overflow.

Error Strings

1. When **fflush();** is called, an EOF will be returned if a write error occurred.
2. The driver can return any of three diagnostic error strings. These are terminated by a CR,LF or newline sequence.
3. Note that the **fscanf();** function will not execute correctly due to the whitespace (blanks, line-feeds) contained in the returned diagnostic error strings. Use **fgets();** instead. This function treats the CR,LF sequence as string delimiters.

Examples

```
main()                                  /*Opening the Device for I/O*/
{
  FILE *SST;
  if ((SST = fopen("$SST", "r+")==NULL)
  {
     printf("Error Opening Device  \n");
     exit(-1)
  }
  .
  . .
}


OutCommand(CommandString)               /* A Function That Outputs */
char *CommandString;                    /* A String of Commands to */
{                                       /* a Device called SST */
  fprintf(SST, "%s   \n", CommandString);
  if (fflush(SST) == EOF) GetError();
}


#define MaxErrorSize 255                /* longest possible error string*/
CalInputString(StringPointer)           /* A Function That Inputs */
char * StringPointer;                   /* A String of ASCII characters*/
{                                       /*From Device SST */
rewind(SST);                            /*(5.0 and above)/
                                        /* Prepares SST for input*/
if(fgets(StringPointer, MaxErrorSize,SST) == NULL)
{
  printf("Error - End of Input detected in Device SST) \n");
  exit(-1);
}
  rewind(SST);                          /*(5.0 and above)*/
                                        /*Prepares SST for output*/
}
```

A.2 TURBO C (VERSIONS 1.0)

The following suggestions apply to application programs written in TURBO C.

Opening a Device

Use the **fopen();** function with the **r+** file mode parameter.

I/O Operations

1. Don't use **rewind()** to switch from Input to Output mode. Instead, use **fclose()** and then re-open the file or else, use Microsoft C.
2. Strings are OUTPUT using the **fprintf();** function.
3. Use the **fgets();** to INPUT strings.
4. After each **fprintf();**, be sure to call **fflush()**. This will flush the output buffer and write an output string immediately. Otherwise, the output strings will be buffered and the device may fail due to overflow.

Error Strings

1. When **fflush();** is called, an EOF will be returned if a write error occurred.
2. The driver can return any of three diagnostic error strings. These are terminated by a CR,LF or a newline sequence.
3. Note that the **fscanf();** function will not execute correctly due to the whitespace (blanks, line-feeds) contained in the returned diagnostic error strings. Use **fgets();** instead. This function treats the CR,LF sequence as string delimiters.

Examples

```
main()                                  /*Opening the Device for I/O*/
{
  FILE *SST;
  if ((SST = fopen("$SST", "r+")==NULL)
  {
     printf("Error Opening Device  \n");
     exit(-1)
  }
  .... other code here....}


OutCommand(CommandString)               /* A Function That Outputs */
char *CommandString;                    /* A String of Commands to */
{                                       /* a Device called SST */
  fprintf(SST, "%s   \n", CommandString);
  if (fflush(SST) == EOF) GetError();
}


#define MaxErrorSize 255                /* longest possible error string*/
CalInputString(StringPointer)           /* A Function That Inputs */
char * StringPointer;                   /* A String of ASCII characters*/
{                                       /* From Device SST */

  fclose(SST);
  if ((SST = fopen("$SST", "r+")==NULL)
{
     printf("Error Opening Device  \n");
     exit(-1)
  }
if(fgets (StringPointer, MaxErrorSize,SST)==NULL)
{    printf("Error - End of Input in Device SST");
     exit(-1);}
  fclose (SST); fopen(SST, "r+");

}
```

A.3 MICROSOFT PASCAL (VERSION 3.0, 3.3, 4.0)

Rules for creating programs in Microsoft PASCAL are described below.

Opening a Device

1. One file variable, "x" should be ASSIGNed for both input and output.

2. Set the MODE field for file "x" to DIRECT.
3. Declare file "x" globally as VAR X : Text(127);

I/O Operations

1. Use Writeln to write commands to the device rather than write.
2. If an error occurs after an I/O operation has been executed, the ERRS field for file "x" will be non-zero.
3. After an error occurs, the ERRS field for file "x" must be cleared.
4. To detect runtime errors, the TRAP field for file "x" must be set.
5. You must use a SEEK(X,1); to rewind the file before any read or write is performed.

Error Strings

1. The driver can return any of three diagnostic error strings. These are terminated by a CR,LF.
2. Use Readln to read diagnostic strings, not read.
3. You must use a SEEK(X,1); to rewind the file before any read or write is performed.

Examples

```
FUNCTION InputString(var Str:LSTring):integer;  (* Returns 1 if error occurs*)
var Result    :integer    ;
BEGIN   X.ERRS  :=0;        (*Input A String From The Device*)
        X.TRAP  :=TRUE;
        Seek(X,1);
        Readln(X,STR)  ;
        IF (X.ERRS <> 0) Then InputString := 1 else InputString :=0;

END;


PROCEDURE GetError ;        (*Fetch the Three Diagnostic Strings*)
BEGIN                       (*That the Device Driver Returns)
     clearscreen;
     Seek (X,1);            (*Rewinds File and Flushes Previous Contents*)
     X.ERRS :=0;            (*Clear I/O Error Flag Before All File Ops*)
     X.TRAP :=TRUE;         (*Trap Errors Instead of Exit to DOS*)
     readln(X,ErrNum);
     readln(X,AStr);
     readln(X,BStr);
     writeln(char(7), (*BELL*) 'Driver Error Has Occurred !!');
     writeln('              1. Device Error Number  =>  'ErrNum);
     writeln('              2. Error  =>  ', BStr);
     writeln('              3. On Command Line of  =>  ', Astr);
END;


PROCEDURE  OutputCommand (STR:String[128]);
BEGIN
  writeln(X,Str);
  if(X.ERRS <> 0) Then GetError; (*Call Error Handling Routine*)
END;


BEGIN                       (*This would be the beginning of the main program*)
  Assign(X,'$SST')          (*Open a Device Driver called $SST for Input *)
  Assign(X,'$SST')          (*Open a Device Driver called $SST for Output*)
  Reset(X);                 (*Input, PASCAL has no read/write text files *)
  Rewrite(X);               (*Output, see above*)
```

A.4 TURBO PASCAL (VERSION 4.0)

The following suggestions apply to application programs written in TURBO PASCAL.

Opening a Device

1. One file variable should be ASSIGNed for input and another for output.
2. The INPUT file variable should be RESET.
3. REWRITE the OUTPUT file variable.

I/O Operations

1. Use Writeln to write commands to the device rather than write.
2. If an error occurs after an I/O operation has been executed, the IOResult will be non-zero.

Error Strings

1. The driver can return any of three diagnostic error strings. These are terminated by a CR,LF.
2. Use Readln to read diagnostic strings, not read.

Examples

```
FUNCTION DeviceInputString(var Str:String[255]):integer;
BEGIN

  Readln(SSTIN,Str);
if  (IOResult <> 0) then DeviceInputString := 1     (*Error in Read*)
else     DeviceInputString := 0;
END;


PROCEDURE GetError ;          (*Fetch the Three Diagnostic Strings*)
BEGIN                         (*That the Device Driver Returns)

  readln(DEVIN,ErrNum);
  readln(DEVIN,AStr);
  readln(DEVIN,BStr);
  writeln(char(7), (*BELL*) 'Driver Error Has Occurred !!');
  writeln('                 1. Device Error Number  =>  'ErrNum);
  writeln('                 2. Error  =>  ', BStr);
  writeln('                 3. On Command Line of  =>  ', Astr);
END;


PROCEDURE  OutputCommand (STR:String[128]);
BEGIN
  writeln(DEVOUT,Str);
  if(IOResult <> 0) Then GetError; (*Call Error Handling Routine*)
END;


BEGIN                         (*This would be the beginning of the main program*)
  Assign(DEVIN,'$SST')        (*Open a Device Driver called $SST for Input *)
  Assign(DEVOUT,'$SST')       (*Open a Device Driver called $SST for Output*)
  Reset(DEVIN);               (*Input, PASCAL has no read/write text files *)
  Rewrite(DEVOUT);            (*Output, see above*)
```

■ ■ ■

Specification Errata

The 1st harmonic distortion specification for the PCIP-SST is incorrect as stated in the user's guide for the board. This specification has been corrected as follows:

1st Harmonic Distortion	New Specification
1 Hz to 499 kHz	Down ≥ 30 dB
500 kHz to 5 MHz	Down ≥ 25 dB

Part #24685, Rev A August, 1996

KEITHLEY METRABYTE

440 Myles Standish Boulevard
Taunton, MA 02780
508-880-3000