PS 390 DOCUMENT SET

INTRODUCTION AND SUPPORT 1–6

EVANS & SUTHERLAND COMPUTER CORPORATION
P.O. Box 8700, 580 Arapeen Drive
Salt Lake City, Utah 84108

Section IS1
Reader's Guide

The PS 390 Document Set (E&S #901194–400) contains operation and programming information and is organized into five volumes. Each volume has been given a two-letter abbreviation, and all sections within each volume are sequentially numbered using the volume abbreviation.

The five volumes and their abbreviation are:

- Introduction and Support (IS)
- Graphics Tutorials (GT)
- Tools and Techniques (TT)
- Reference Materials (RM)
- Advanced Programming (AP)

The document set is designed and sequenced to be helpful both to the novice and to the experienced programmer. The document set contents are described by volume in the following sections.

1. Introduction and Support (IS)

The Introduction and Support (IS) volume provides general information about the PS 390 graphics system and its use. The IS volume should be read by all users.

IS1. Reader's Guide

This section contains abstracts of each volume and section in the PS 390 Document Set.

IS2. PS 390 System Overview

The System Overview section is a summary of system capabilities and an introduction to PS 390 terminology. It describes the basic PS 390 system configuration, including a discussion of interactive devices and the display. This section also covers information on the runtime firmware and the host-resident software.

IS3. Operation and Communication

Read this section before attempting to start up the machine, boot Graphics Firmware, operate the keyboard, or enter the tutorial commands.

Step-by-step instructions for starting and operating the PS 390 using a variety of host interfaces are presented in this section.

IS4. Maintenance and Services

This section supplies Software and Documentation Hotline numbers, along with descriptions of E&S maintenance plans and training programs.

IS5. PS 390 Site Preparation and Customer Support Guide

Place the Site Preparation and Customer Support Guide (E&S #901194–091) you received prior to receiving your system in this section. The guide includes information on installing the PS 390 and interactive devices.

IS6. PVT Guide

Place the *Customer Guide to PS 300 System Performance Verification Test* (E&S #901194–081) which was delivered with your software diskettes in this section. This guide accompanies the performance verification software and explains the PVT program. If you purchased the rendering option, the *Customer Guide to the PS 390 Rendering Performance Verification Test* (E&S #901194–086) was also delivered with the software diskettes.

2. Graphics Tutorials (GT)

The Graphics Tutorials (GT) volume is a self-paced tutorial that teaches graphics programming for the PS 390. It is designed as an introduction to users who have no experience in programming the PS 390. Programmers experienced with other graphics systems should read those sections which explain concepts and operations that are specific to the PS 390. This volume can also be used as a review of programming procedures where needed. Because of its length, this volume is separated into two parts: *GT 1–7* and *GT 8–16*.

GT1. Hands-On Experience

This section steps through simple operations to introduce the user to several basic PS 390 programming concepts. It is designed to give the user practical knowledge of basic system functions.

GT2. Graphics Principles

This section discusses basic graphics operations and principles, and how these operations are performed using the PS 390. Included is information covering polygonal rendering and shading capability.

The Graphics Principles section should be read by all graphics programmers. For novices, it introduces essential concepts taught in the remaining tutorial sections. For the advanced programmer who does not plan to study the tutorials, it is background information for the remaining volumes in the PS 390 Document Set.

GT3. PS 390 Tutorial Demonstrations

This section is designed to accompany the tutorial demonstration programs. These demonstrations are selected to illustrate and clarify concepts taught in the GT sections. The package should be used as a supplemental learning tool with each section. This package also contains graphics primitives for use in the rest of the tutorial sections.

GT4. Modeling

This section details how to design a display structure for a model, including designing an organizational hierarchy, designing a detailed display structure, and designing a complex model.

GT5. Command Language

This section explains the command language used create and modify display structures, create and modify function networks, and instruct the display processor and command interpreter.

GT6. Function Networks I

The first Function Networks section covers the basic knowledge you need to interact with a model created for display. It covers the PS 390 dials that allow you to rotate, scale, and translate the displayed model.

GT7. Function Networks II

This section is a continuation of Function Networks I. It covers using the PS 390 dials for multiple interactions, and using the function keys to select sets of operations.

GT8. Viewing Operations

This section discusses the two types of viewing transformations: line of sight and windowing. Also included are setting intensity ranges for depth cueing, setting displays, and setting color.

GT9. Conditional Referencing

Conditional referencing is a way to display selected branches of a display structure without displaying other branches. This section covers the three attribute settings for conditional referencing: conditional bit, level of detail, and rate settings.

GT10. Text Modeling and String Handling

Text is handled in the PS 390 the same as any other graphical item. This section covers the creation and manipulation of character strings using commands and the interactive devices. Included is how to use and create different fonts.

GT11. Picking

Picking allows the user to retrieve information about a selection (pick) made on displayed data. Information can include the name of the data node that the picked portion of the object is associated with, names of nodes along the branch of the display structure, or an index into the vector list. This section defines the elements involved in picking operations and explains how to perform picking operations.

GT12. PS 390 Video Output Control

This section explains how to control the video output of the PS 390. It describes the use of filters to implement antialiasing, setting intensity modulation, selecting the color of the screen cursor, selecting background/foreground color, and selecting a video timing format.

GT13. Polygonal Rendering

This section covers the commands used to define objects eligible for polygonal rendering. Included is how to perform hidden-line removal, backface removal, sectioning, cross-sectioning, and shading operations.

GT14. Raster Programming

This section explains how the PS 390 system can be used as an image buffer to display host-generated images. It includes information on run-length encoding of host data and using color lookup tables.

GT15. Sample Programs

This section contains code examples of various PS 390 programs that demonstrate programming techniques and applications. The programs are designed to be used in conjunction with the tutorials in this volume.

GT16. Glossary

The glossary contains definitions of terms used in all five volumes of the PS 390 Document Set.

3. Tools and Techniques (TT)

The Tools and Techniques (TT) volume describes programming aids for the PS 390. It includes information such as helpful hints, how to use the various graphics editors, and explanations of other utilities and applications.

TT1. Application Notes

This section is collection of useful samples and applications for PS 390 users. Contributions to the Notes come from PS 390 users inside and outside Evans & Sutherland.

TT2. Helpful Hints

This section contains task-oriented information such as defining break keys, using the SITE.DAT file, name suffixing, and CPK rendering. This section assumes a good working knowledge of the PS 390 and some programming experience.

TT3. Using the GSRs

This section is an introduction to using the graphic support routines (GSRs). The GSRs are a method of communicating graphics data to the PS 390. The GSRs can be used with FORTRAN, Pascal, or C programming languages.

TT4. Function Network Editor

This section describes NETEDIT, a program which allows the user to create a function network using a diagram on the PS 390 display rather than directly inputting commands to a file. Editing menus are used to build a network of "black boxes" which may contain more detailed representations of parts of the network. When the diagram is complete, an ASCII code file can be generated which contains the PS 390 commands needed to build the network.

TT5. Function Network Debugger

This section describes NETPROBE, a utility developed at Evans & Sutherland to be used as a guide for a user-written network debugging program.

TT6. Data Structure Editor

This section describes STRUCTEDIT, a graphical display structure editor for the PS 390 that allows the user to sketch out a display structure, then converts the diagram into ASCII PS 390 commands or a routine that can be included as part of a FORTRAN, Pascal, C, or LISP program.

TT7. Character Font Editor

This section describes MAKEFONT, a program that allows the user to edit an existing character font or create a new one. It is an interactive, menu-driven program that displays characters in a 128- or 256-character font. Each character can be edited to create a new shape. Different fonts can be combined into a new font, and original fonts can be created from scratch.

TT8. ASCII-to-GSR Converter

This section explains how to run the ASCII-to-GSR conversion program. The program is a host-resident PS 390 option that allows you to combine ASCII programming with the faster data communication speeds available through the GSRs.

TT9. Transformed Data and Writeback

This section provides information on how to retrieve transformed data, such as a matrix or vector-list representation of transformation operations.

TT10. Crash Dump File

The crash dump file is created when a system crash occurs. This section explains how to read back the file and includes an example of a Pascal host program that writes the information from the PS 390 crash file into a host file.

4. Reference Materials (RM)

The Reference Materials (RM) volume provides reference information on PS 390 commands, functions, GSRs, utilities, and communication protocols. Also included is information on peripherals, interfaces, options, and a list of system errors. Because of its length, this volume is separated into two parts: *RM 1–4* and *RM 5–16.*

RM1. Command Summary

This is an alphabetically organized summary of the ASCII form of every PS 390 command. It contains the full command, the acceptable abbreviated form of the command, options, parameters, default values, and, where appropriate, notes and comments. If a command creates a node in a display structure, the inputs and acceptable data types for that node are shown.

RM2. Intrinsic Functions

This section describes in alphabetical order the PS 390 intrinsic user functions and intrinsic system functions. Each description gives a brief overview of the function and a diagram showing input queues, outputs, and the data types for all inputs and outputs. Notes and comments are included where appropriate.

RM3. Initial Function Instances

This section provides information about PS 390 initial function instances. Each description gives a brief overview of the initial function instance and a diagram showing input queues, outputs, and the data type for all inputs and outputs. Notes and comments are included where appropriate.

RM4. Graphics Support Routines

This is an alphabetically organized summary of the host-resident GSRs. FORTRAN, Pascal, and C routine names, parameters, and descriptions are provided. The corresponding PS 390 command and syntax is also included.

RM5. Host Communications

This section includes RS-232 and RS-449 interface specifications and pin connector definitions, PS 390 transmission protocol, port values and defaults, and the PS 390 system data reception functions.

RM6. Interfaces / Options

This section covers general information about PS 390/host interfaces and options. Its main purpose, however, is to provide a place for the insertion of Customer Installation and User Manuals for any interfaces and options ordered with the system (or subsequently). These manuals are typically shipped prior to E&S hardware installation so that customers can adequately prepare the site and their host computer.

RM7. Host Input Data Flow

This section covers information on the input data flow, including using routing bytes, routing byte definitions, and routing functions.

RM8. System Function Network

This section contains diagrams of the PS 390 system network. The diagrams show the logical paths of routing bytes and functions.

RM9. Initial Structures

This section describes initial data structures created by the runtime firmware. Configure mode is discussed and a "runtime" system is defined.

RM10. Terminal Emulator

Instructions are given in this section for changing the modes and features of the Terminal Emulator by either sending escape sequences from the host, entering PS 390 commands in the SITE.DAT file, or sending the appropriate ASCII characters to terminal emulator functions.

RM11. System Errors

This is a compendium of all error messages (informational, warning, recoverable, and fatal) that a user might encounter. Error messages are listed in numerical order. The text of the message is given with an indication of common causes of the error and, where appropriate, ways to correct it.

RM12. Diagnostic Utilities

This is a reference for the utility commands that are on the PS 390 diagnostic utility diskette.

RM13. Interactive Devices

This section provides hardware and data-transmission descriptions of the PS 390 interactive devices. Descriptions are included for the peripheral multiplexer, keyboard, data tablet, function buttons, control dials, mouse and display.

RM14. GSR Internals

This section describes the data formats expected by the PS 390 command interpreter. It is provided for advanced programmers to write their own GSRs.

RM15. Release Notes

A divider tab is provided for information supplied with future releases of software.

RM16. Index

This section is provided as a reference to all five volumes of the PS 390 Document Set.

5. Advanced Programming (AP)

This volume contains information for performing advanced programming. Included is information on PS 390 internal processing, mass memory structures, physical I/O, user-written functions, and data and operation node formats.

IS2. SYSTEM OVERVIEW

IS2. PS 390 SYSTEM OVERVIEW

CONTENTS

ILLUSTRATIONS

Section IS2
PS 390 System Overview

The PS 390 graphics system was developed to put the power of sophisticated computer graphics in the hands of users who are primarily designers involved in complex design and analysis tasks. The machine takes much of the burden of graphics processing and interactive-device handling away from the host computer and, more importantly, the user. The user is free to concentrate on the application task for which the PS 390 is being used. The PS 390's own high-level ASCII command language was designed to let the user program in a way that is closer to how a designer thinks. Memory is treated not as a set of addresses where data is stored, but as a collection of structured objects created and accessed by name. Minimum knowledge of matrix arithmetic is needed to manipulate graphical data since the system automatically performs all transformation processing and the necessary matrix concatenations. The result is a system which is several steps removed from traditional graphics machines, much as a high-level programming language differs from assembly languages.

1. Features of a High Performance Machine

The PS 390 allows the user to create, display, modify, and manipulate complex 2D and 3D wireframe models interactively. Using Evans & Sutherland's Shadowfax™ technology, the PS 390 combines the real-time interaction capability and line quality of a calligraphic system with the flicker-free images of a raster system. With the purchase of the Advanced 3D Visualization Option, high-quality shaded static images of models can be rendered locally on a full display or in selected portions of the display. The PS 390 also provides a variety of user-selectable shading styles and a range in the quality of the anti-aliased lines for tailoring speed, quality, and performance characteristics to specific applications. The following is a summary of important features of the PS 390 graphics system.

- Hierarchically Structured Models

  The PS 390 system allows the user to create three-dimensional objects as lines, polygons, and characters and to store them in the system's own mass memory. Models are created as hierarchical groupings of graphical data (points and lines or planes), mathematical operations (transformations) which are applied to the data, and attributes of the model such as color and intensity. Hierarchical structuring allows complex objects to be created from simpler parts. Individual components of a model can be used again and again. Changes can be made to individual parts of the structure without the need to recreate the whole structure.

- Control of 3D Models

  Using commands or interactive devices, the user can manipulate wireframe images of models, translating (moving), scaling, and rotating the images by any amount in any direction.

- Local Manipulation of Models

  The PS 390 controls the interactive manipulation of models locally. Values which are input from the various interactive devices (dials, keys, data tablet, buttons, and mouse) are sent through user-designed function networks to interaction points in the model's structure. The host computer never has to participate in handling the interactive devices.

- Real-Time Interaction

  Real-time interaction refers to the ability of an image to respond instantly to input from an interactive device. The PS 390 lets the user interact with displayed images of complex 3D models in real time within a dynamic viewport on its raster screen.

- Perspective Views

  The system is capable of displaying objects in perspective to enhance the illusion of three dimensions. In perspective views, lines which recede from the eye appear to converge. When an object viewed in perspective is manipulated on the screen (translated, rotated, or scaled) the system maintains a true perspective view of the object.

- Depth Cueing

  To further enhance the illusion of depth, the system performs depth cueing. This is a visual effect to allow the viewer to perceive three dimensions on a two-dimensional screen. A line in a picture which represents the depth dimension (into the screen) grows dimmer the "farther away" it is from the viewer.

- Text as a Graphical Item

  The system treats text as any other graphical item, allowing interactive translation, rotation, and scaling of characters and text strings. The PS 390 has a standard character font. Commands are available to create, modify, and use any other style or size of character font.

- Use of Color

  In graphical renditions of complex three-dimensional structures, color can be an important asset in analyzing the design. The system gives the programmer the option of displaying parts of a model in different colors in a dynamic viewport on the display.

- Optional Advanced 3D Visualization of Solid Objects

  The PS 390 also allows users to create objects as polygons and to display hidden line–removed and sectioned views of polygonally defined wireframe objects. Smooth-shaded renderings of polygonal models can be displayed that take advantage of numerous attribute settings for color, multiple light sources, specularity, transparency, and polygon edge enhancement. In addition the PS 390 can be used as a frame buffer for the display of host-generated, run length–encoded images.

- Distributed Processing

  PS 390 systems allow high-performance interactive graphics in a distributed graphics environment. The high degree of local intelligence of the systems allows the PS 390 to store the graphical data base, perform all transformation and display processing, and handle data from the interactive devices without intervention from the host.

- Host Independence

  No specific size or make of host computer is dictated for the PS 390. The system communicates over a low-bandwidth interface to virtually any host that can accept RS-232-C and RS-449 asynchronous serial communications using START/STOP protocol. Other PS 390/host computer interfaces are available for high-speed data transfers to several models of DEC and IBM computers.

2. PS 390 System Configuration

The basic system configuration for the PS 390 is the control unit and display, various interactive devices, and a host computer.

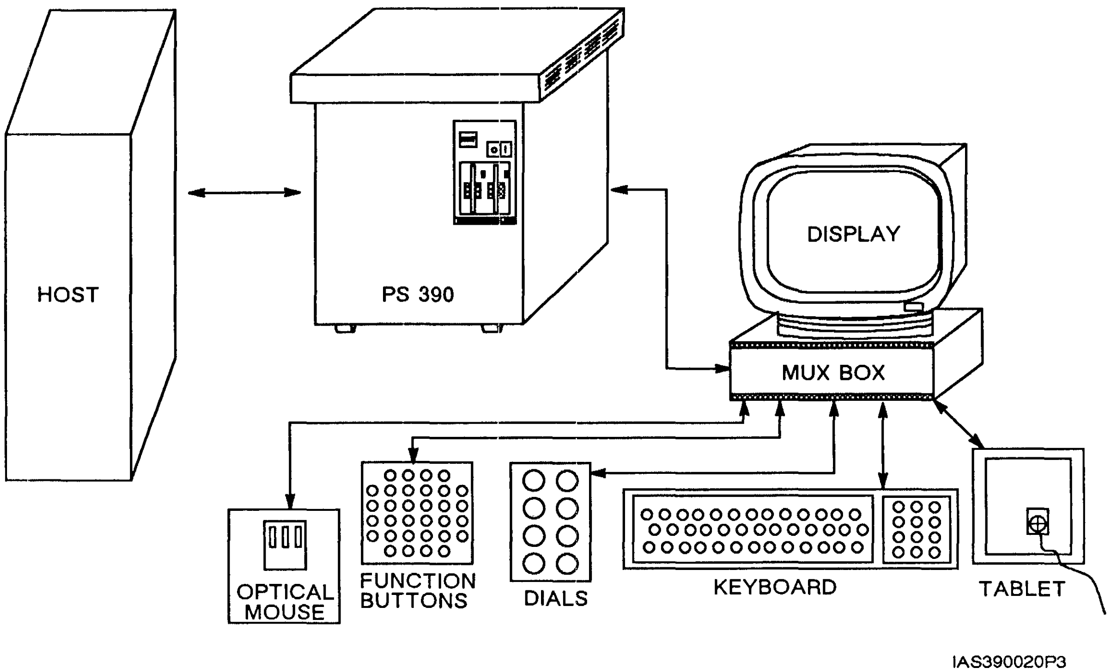

Figure 2-1. Basic System Configuration

2.1 The Control Unit

The PS 390 control unit is housed in the cabinet which contains the power supplies, the standard and optional circuit cards for the system, and the floppy diskette drives for loading the PS 390 firmware and data files.

The free-standing control unit is approximately 53 cm (21 in.) wide, 71 cm (28 in.) deep, 67 cm (26.5 in.) high, and weighs 55 kg (120 lbs.). The top holds over 250 pounds static weight or 180 pounds rolling load.

There are two external controls on the PS 390 control unit. One is the ON/OFF toggle switch, located at the top right of the front panel. A RESET switch is located just left of the ON/OFF switch. The RESET switch allows the system to be reset instead of powered off during a system lock or reboot.

The PS 390 floppy disk drives are located at the front of the unit near the upper right corner. The system uses double-sided, quad-density, 5-1/4 inch minifloppy diskettes capable of storing 737,280 formatted bytes on 160 tracks.

2.2 Port Configuration

The PS 390 supports four asynchronous RS-232 ports on the communications connector panel. Port 0 and Port 2 are physically present but not usable. See Figure 2-2.

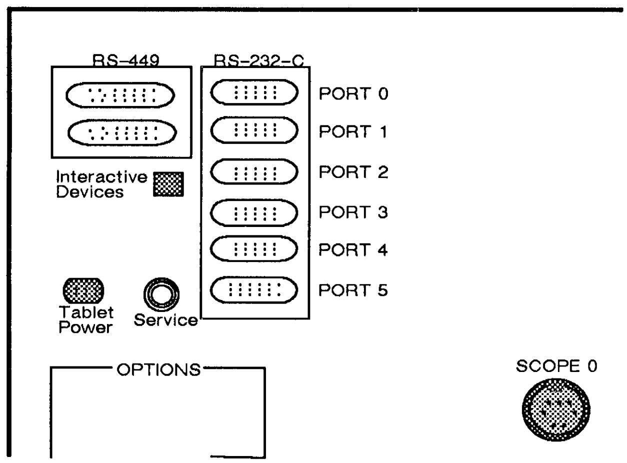

Figure 2-2. PS 390 Communication Ports

The port configuration for the PS 390 is as follows:

- Port 1 is the host port.
- Port 3 is the debug port, for diagnostic purposes.
- Port 4 may be used for special interface applications, including an alternate diagnostic port.
- Port 5 is the peripheral multiplexer port.

2.3 The Display

The PS 390 supports a color raster display. The PS 390 provides interactive manipulation of high quality anti-aliased wireframe images, as well as the rendering of shaded images in a variety of shading styles. Many combinations of viewports can be defined for the display of multiple views and renderings of objects.

2.4 Power Requirements

The following power requirements apply to the PS 390:

- 110V, single phase (± 10%), 47–63 Hz, 12 amps (max) for the control unit
- 115V, single phase (± 10%), 50–60 Hz, 1.5 amps (max) for the color raster display
- 90–130V, single phase, 47–63 Hz, 0.6 amps (max) for the peripheral multiplexer

2.5 Basic Card Set

There are five basic circuit cards: Joint Control Processor (JCP), Arithmetic Control Processor (ACP), Pipeline Subsystem (PLS), Raster Backend Bit-slice Processor (RBE/BP) and Raster Backend Video Controller (RBE/VC). The last four of these cards can be grouped into a functional unit called the display processor.

2.5.1 The Joint Control Processor

The JCP consists of two (optionally three) sections: control processor, mass memory, and interface sections. The JCP is based on a 68000 10-MHz microprocessor with a 24-bit address space. The JCP contains 512K bytes of local memory which is loaded with graphics firmware when the system is powered on. There is a local path to the JCP-resident mass memory which provides access from the 68000. The JCP also contains communications controllers for host and interactive-device handling and for bus-interface circuitry.

The JCP's mass-memory section has two megabytes of memory. Part of memory is taken up with system networks and structures. These are loaded from the firmware diskette when the system is booted. The rest of memory is available for storing data bases of models and function networks.

The JCP manages memory automatically. The user does not have to worry about buffering schemes, addresses, or garbage collection. The user accesses memory by the names of the structures that are created.

The JCP's optional interface section provides a location for an IBM 3278 interface. This option allows the PS 390 to communicate with an IBM 3274 control unit over a 56KB line.

The JCP performs the following tasks:

- Controls communications with the host.
- Provides the terminal emulator capability (DEC VT100 or IBM 3278).
- Processes commands and creates data structures and function networks in mass memory.
- Performs memory management.
- Handles interactive devices by executing user-designed function networks.
- Synchronizes updates so that complex interacting motion does not appear disjointed.

The JCP can be enhanced with General Purpose Interface Option (GPIO) cards. These handle high-speed communications with DEC and IBM hosts other than the IBM 3278 (which is a configurable option on the JCP itself).

2.5.2 The Display Processor

The display processor section of the PS 390 consists of four cards: ACP, PLS, RBE/BP and RBE/VC.

The display processor accesses data in mass memory and generates a picture for display on the screen. It traverses the structures of objects to be displayed, performs the transformations indicated, and generates the lines to be drawn on the screen.

The patented Shadowfax™ technology is contained within the display processor and is located on the RBE/BP and RBE/VC cards. Shadowfax™ technology consists of the custom VLSI circuitry which produces the calligraphic-quality (anti-aliased) lines on a raster display. Shadowfax VLSI subpixel processing mathematically divides each pixel into 64 subpixels. This provides addressability and image quality of 8192 by 6912 on the 1024 by 864 pixel display.

The display processor performs the following tasks:

- Reads and traverses structures in mass memory.
- Performs character generation.
- Does geometric character matrix concatenations. These only affect text, not three-dimensional data.

- Does 3x3 matrix concatenation affecting text and data
- Performs 3D translation concatenations
- Does 4x4 windowing concatenation, including perspective
- Performs all data transformation through matrices
- Performs clipping
- Performs viewport mapping
- Controls blinking
- Handles picking
- Performs digital-to-analog line conversion, including color, for circuitry in the PS 390 display
- Performs high-quality anti-aliasing of lines

2.5.3 Optional Cards

Currently, the optional cards supported for the PS 390 are the GPIO and Mass Memory (MM) cards.

The PS 390 runtime firmware supports up to two GPIO interfaces of differing types installed in the same system, as well as asynchronous communications. The default configuration is asynchronous but the user has the ability to configure any interface when the system is booted.

It is also possible to change the configuration without rebooting the PS 390 because the runtime software determines which of the interfaces are in the system and initializes them all. This is achieved through runtime identification of up to two GPIOs at the first two addresses assigned to GPIO interface cards. However, there are some limitations to the use of multiple GPIOs. First, there cannot be two of the same type GPIOs in the same system. Second, the IBM 3278 option is regarded as a GPIO interface, although it is included as an optional interface on the JCP rather than a separate interface card. This means that only one additional GPIO may be added when the IBM 3278 option is used.

The following are available GPIO options.

- Ethernet — Allows a PS 390 to communicate with a DEC VAX host via an Ethernet network.
- IBM 5080 — Allows a PS 390 to interface to an IBM host computer via an IBM 5088 control unit.

- Unibus Parallel Interface — offers 16-bit parallel direct memory access (DMA) communication between the PS 390 and a DEC VAX host. The interface is a bidirectional differential-driven data path. It can transfer data at a rate of one megabyte per second.
- IBM 3278 — Available as an optional interface section on the JCP card. It provides a location for an IBM 3278 interface. This option allows the PS 390 to communicate with an IBM 3274 control unit over a 56KB line.

The PS 390 mass memory can be expanded with two optional MM cards separate from the mass memory which exists on the JCP card. Each separate MM card contains one megabyte of mass memory. Total mass memory is expandable to four megabytes.

The JCP communicates with separate MM cards across a 16-bit path, creating structures in memory or sending data to function networks or interaction points in a model's structure. The display processor communicates with the mass memory over a 32-bit path to traverse the structures of models. This broad path ensures fast display access time. Figure 2-3 provides a simplified functional overview of the PS 390 graphics system.

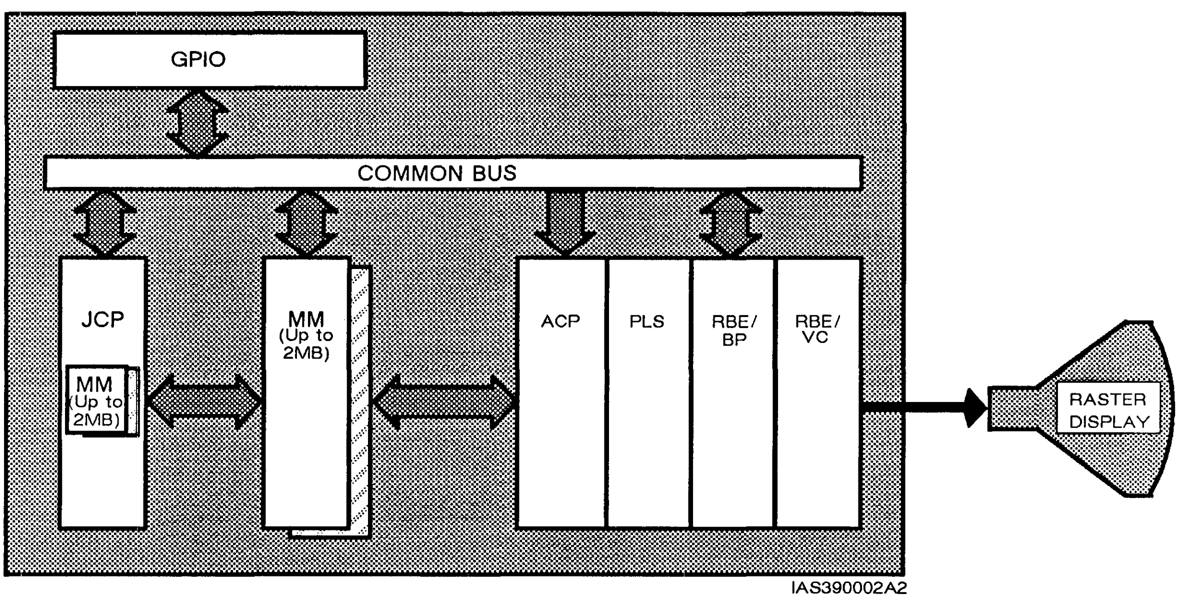

Figure 2-3. PS 390 Architectural Overview

2.6 Card Configuration

The following is the proper card configuration for the PS 390 as seen from the back of the control unit.

```
GPIO*(2)  MM*(2)  JCP  ACP  PLS  RBE/VC (blank)  RBE/BS
```

* optional cards (2). There are two available slots for these options.

2.7 Interactive Devices

Several optional interactive devices are available with the PS 390 systems. Two styles of interactive devices are available: the PS 300 style and the PS 390 style. The two styles cannot be mixed on the same system. They are programmable, easy-to-use devices which allow a PS 390 user to interact with the images displayed on the screen. All devices have local intelligence provided by a microprocessor. Details on the interactive devices are contained in Section *RM13*.

2.7.1 Peripheral Multiplexer

Interactive devices for the PS 390 are connected to the peripheral multiplexer (mux box) contained in a three-inch pedestal that supports the raster scope. All interactive device connections for the mouse, function buttons, control dials, keyboard and data tablet are clearly marked on the front panel of the mux box. Figure 2-4 shows the front view of the mux box.

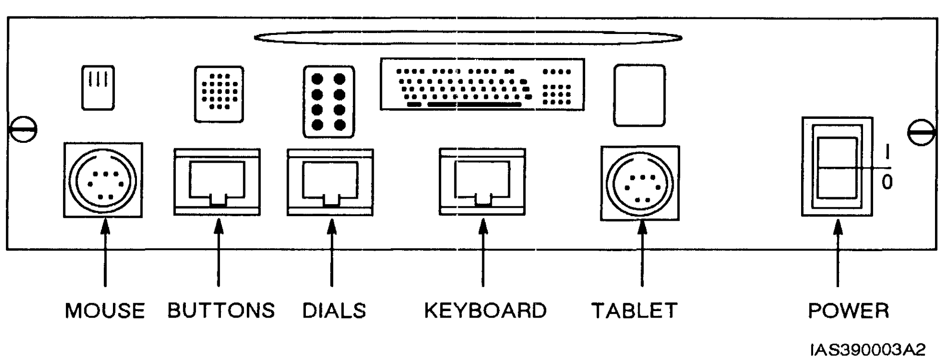

Figure 2-4. Front Panel for PS 300 Interactive Devices

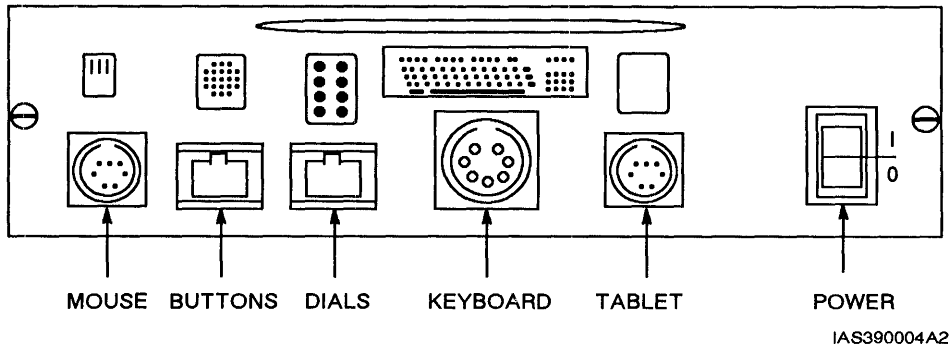

Figure 2-5. Front Panel for PS 390 Interactive Devices

The back panel of the mux box has an RS232-C connection and two external power connections. All cables and connections are clearly marked. To maintain EMI integrity, the screws on the RS232-C shielded cable must be tightly turned on the connection. Figure 2-6 shows the rear panel of the mux box.

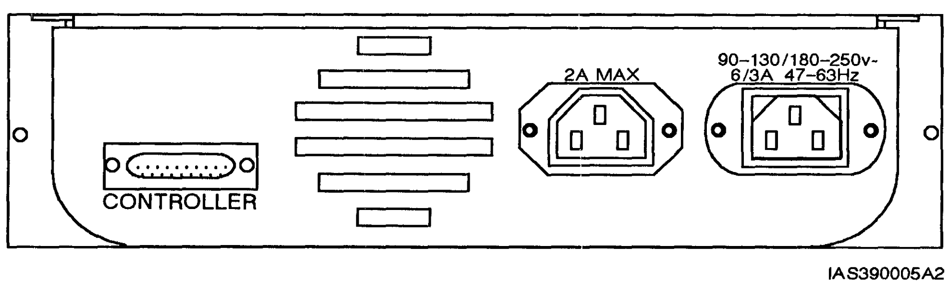

Figure 2-6. Peripheral Multiplexer Back Panel

2.7.2 Keyboards

The basic PS 390 keyboard includes all standard alphanumeric keys, a separate numeric keyblock, typical typewriter control keys, symbols, and a set of 12 function keys which can be programmed to perform interactive graphics functions.

The 8-character optional LEDs on keyboards are usually programmed to display prompts to aid the operator or labels that describe the operations being performed (e.g., ROTATE IN X).

2.7.3 Function Buttons

A unit containing 32 programmable function buttons is available with the PS 390. The function buttons are arranged with one row of four buttons, four rows of six buttons, and a final row of four buttons. Pressing a function button results in a user-specified action. The buttons are usually programmed to display different views of the same object or to switch between views of different objects.

2.7.4 Control Dials

A set of eight programmable control dials is available with the PS 390. The control dials are used to control size, position, and orientation of displayed objects, and for other programmed functions. The dials are mounted in two rows of four dials. Above each control dial is an 8-digit optional LED display which can be programmed to display the function of the dial or other information associated with the use of the dial.

2.7.5 Data Tablet

There are two data tablets available for the PS 390. One is a 6-inch square tablet, the other is a 12-inch square tablet. Both tablets have four-button "pucks." They are alike except for their active areas. Both tablets are normally used as an interactive pointing and positioning device to control the cursor on the display.

2.7.6 Optical Mouse

An optical mouse is available with the PS 390. The mouse sends X- and Y-axis position data to the JCP. The position information provided by the mouse is similar to that provided by the data tablets.

3. Host Communications

The PS 390 does not rely on a host to perform any of the graphics processing or handling of data from the interactive devices. In most applications, the host is used for analysis programs and for file storage. Since the PS 390 does not store commands or save files which create objects and function networks, most users create files on the host and transfer them to the PS 390.

3.1 Host Interfaces

The standard data communication interfaces to the PS 390 are asynchronous serial line RS-232-C or RS-449 using START/STOP protocol. These two interfaces are ideal for interfacing to most host computers in applications requiring a maximum data transfer rate of 19.2K baud. RS-232-C is recommended for distances up to 50 feet between the PS 390 and the host. For runs longer than 50 feet, RS-449 is recommended. A maximum cable length of 3,000 feet is available for 19.2K baud transfers.

In addition to the standard interfaces, different types of high-speed data transfer interfaces are available for selected IBM and DEC computers. These offer faster transfer rates and allow greater distances between the PS 390 and the host. Up to two of these interfaces can be configured in the PS 390, and the user is able to toggle between the two. This is described in detail in Section *RM6*.

3.2 Communication Lines

Asynchronous communication between the PS 390 and the host can take place over one or two lines.

In single-line communications, graphics commands and the terminal emulator for the host are multiplexed over one line. Control characters preceding and following the data are used to route information to the terminal emulator software or to the the PS 390 command interpreter.

In dual-line communications, one line is used to communicate graphics commands, and the other is used for terminal emulation. No multiplexing occurs with dual-line communication.

4. Distributed Media: PS 390 Firmware and Host Software

The distributed media that accompanies all PS 390 systems includes the PS 390 graphics firmware diskettes, other distributed system diskettes, and the PS 390 host-resident software that are distributed on magtape.

4.1 Other Diskettes Shipped With the System

The PS 390 system is shipped with several other system diskettes in addition to the PS 390 graphics firmware diskettes. These include:

- The PS 390 Performance Verification Test
- The PS 390 Rendering Performance Verification Test
- The PS 390 Demonstration Package diskettes
- The PS 390 Diagnostic Utility diskette

Guides to the Performance Verification Tests are shipped with the diskettes. Section *IS6* of this volume has been provided for placement of those guides. Documentation for running the PS 390 Demonstration Package is in Section *GT3*.

The Diagnostic Utility diskette contains Utility Commands that are used to copy diskettes and to delete files on diskettes. Refer to Section *RM12* for the definition and use of the diagnostic utilities.

4.2 Host Resident Software

Along with the system diskettes, E&S distributes files on magtape that will be loaded onto the host system. These files contain various applications and utilities that are used for PS 390/host communication. The magtape can contain the following files:

- A README file that describes the files on the magtape.
- PS 390/Host communication tests.
- The PS 390 Graphics Support Routines source code.
- A SITE.DAT file for setting up dual-line configuration of the terminal emulator.
- Instructions for installing the cross-compatibility software for both IBM and DEC operating systems.
- A file with the execs or jobstreams to be used as examples for installation procedures for both IBM and DEC operating systems.
- Three programming utilities (For VAX/VMS users only): NETEDIT, NETPROBE, and MAKEFONT.
- The E&S supplied files that are used in developing and transporting user-written functions.

4.3 The Graphics Support Routines

The PS 390 Graphics Support Routines (GSRs) are a set of software routines developed and supported by Evans & Sutherland that allow for faster graphics transactions on the PS 390 System.

The GSRs are distributed as either callable FORTRAN, Pascal, or UNIX/C routines. It is the responsibility of the user to load, compile, and link the routines with their application program. Installation instructions for the GSRs are provided in Section *TT3*.

The source code for the GSRs contains all files necessary for the customer to compile and link the GSRs, once the tape is loaded on the host. The file names and descriptions are included in the installation instructions.

The GSRs are supported in DEC VAX/VMS, DEC/UNIX, IBM VM/SP CMS, and IBM OS/VS2-TSO environments.

The documentation for the GSRs is provided in Section *RM4*.

5. The PS 390 Command Language

The PS 390 has its own command language which has two equivalent representations: ASCII and binary. Both forms can be used over most communications lines.

The ASCII form of the language consists of English-like commands which closely reflect typical operations that are performed in computer graphics applications. For example, the ROTATE command rotates an object through any angle around any axis; the VECTOR_LIST command creates an object as a list of vectors; the DISPLAY command produces an image of an object on the screen, and so on. The ASCII commands were designed to simplify graphics programming for users who are primarily structural designers and analysts, not experienced graphics programmers.

Using PS 390 commands, the user builds structures which represent the objects and models to display and interact with. Because of the PS 390 "naming convention," however, the user never has to treat the data which defines an object as addresses in memory. Instead, memory is treated as a collection of objects, each created with a name and accessed by that name using the system's commands.

Objects are named groupings of graphical data, mathematical operations called transformations which are applied to the data, and attributes such as color and level-of-detail. Once created, a single named definition of a graphical object can be used again and again like a template to build an endless number of other objects. It can be included in other structures by a simple reference to its name.

For example, a scientist modeling a molecule of water could first create Atom as a primitive object from which the hydrogen and oxygen atoms can be built. By scaling the original atom, objects named *Hydrogen_Atom* and *Oxygen_Atom* can be created. These objects are created as named entities and worked with as named entities. To create and position an additional hydrogen atom, a translation can be applied to *Hydrogen_Atom* to give *Hydrogen_1*. Another translate operation applied to *Hydrogen_Atom* will yield *Hydrogen_2*. An object named *Water_Molecule* can then be created. Its structure uses one instance of *Oxygen_Atom*, one instance of *Hydrogen_1*, and one instance of *Hydrogen_2*, each correctly positioned to simulate the structure of the molecule. Whenever the scientist accesses *Water_Molecule* to change its definition, to display it on the screen, or rotate it using a dial, it is accessed by name.

Both the form of the PS 390 command language and the treatment of graphical data as named entities in memory reflect the design of the PS 390 as a tool for designers and analysts, not just graphics programmers.

Commands fall into the following categories:

- Function
- General
- Modeling
- Rendering
- Structure
- Viewing

Function commands connect and disconnect inputs and outputs of functions, set the inputs of functions to be constant, and send data to inputs of functions.

General commands display and remove objects from the screen, alter the structures of objects by including, prefixing, or following named objects with other named objects, and build function networks which allow the user to interact with the model by connecting interactive devices to places in the structure of the object.

Modeling commands create primitive objects as vector lists, curves, and polygons, and perform three-dimensional modeling transformations (rotate, translate, scale) on objects and two-dimensional transformations on characters.

Rendering commands declare polygon objects to be surfaces or solids so that rendering operations may be performed on them, and specify the characteristics of polygons (such as hue, saturation and intensity) used in the creation of shaded renderings.

Structure commands are used for creating the structure of objects, naming objects, explicitly and implicitly referencing objects, and conditionally referencing other objects. They create display structures which are the structured grouping of graphical data, transformations, and attributes which define an object.

Viewing commands create different views of objects by letting the user specify a line of sight, orthographic views, perspective views, and viewports, and set and change appearance attributes of a model such as the color or the intensity of the lines displayed.

5.1 Using the Command Language

Each command directs the machine to perform a graphical operation which would require several statements in a conventional programming language such as FORTRAN or Pascal. Once a command is entered, the command itself no longer exists. Instead, the command is interpreted and an action is taken or an entity is created in mass memory. PS 390 commands can be entered in command mode or downloaded in a file from the host system. Commands entered in command mode are immediately interpreted by the PS 390. As the PS 390 does not store data entities between power-up cycles, when the system is turned off memory is cleared and existing entities are lost.

The terminal emulator mode is provided to allow the PS 390 to log on to the host system. In terminal emulator mode, the PS 390 keyboard and display can emulate either a DEC VT100 terminal or an IBM 3278 terminal.

Commands can be entered into a file on the host system and then downloaded to the PS 390. This method uses the host as a storage device for data files.

PS 390 commands are transmitted in ASCII or in binary format. The PS 390 has its own command interpreter which checks the syntax of commands and puts them into effect. When commands are sent in ASCII, they pass through a system function that parses and packages the data into binary data packets that are accepted by the command interpreter.

The GSRs are the binary form of the PS 390 command language, and allow for more efficient communication between the PS 390 and the host system. These routines prepackage all commands into binary packets on the host and provide much faster throughput for downloading PS 390 command files from the host. They are available in FORTRAN and Pascal for DEC VAX/VMS systems and IBM systems, and in C for DEC/UNIX systems.

5.2 Structures are Hierarchical

The structures that are created in memory are structured groupings of the graphical data which define shapes and the transformations which are performed on the data. These structures form a set of instructions which the display hardware processes to generate a picture on the screen. Elements in the structure are organized hierarchically. Implicit in this hierarchical organization are the operations and transformations that should be applied down any one path.

In a hierarchical structure, each element is used as a reference to all elements below it. So in the structure for Water_Molecule:

Atom references the vector list which defines a basic atom.

Hydrogen_Atom and *Oxygen_Atom* reference scaling operations that are applied to *Atom*.

Hydrogen_1 and *Hydrogen_2* reference translation operations that are applied to *Hydrogen_Atom*.

Water_Molecule references the two scale operations applied to *Atom*, and the translations applied to the scaling operations applied to *Atom*.

This type of structure is typical of the structures of objects you create when programming the PS 390. It is referred to as a display structure. Each element in the structure is called a node. A node is either a data node, such as *Sphere*; an operation node, such as *Hydrogen_Atom* (scale) or *Hydrogen_1* (translate); or an instance node, such as *Water_Molecule*, which groups other elements under a single name. Display structures can be drawn as diagrams which represent the hierarchical structuring of data and operations which define an object. A display structure for *Water_Molecule* is shown in Figure 2-7.

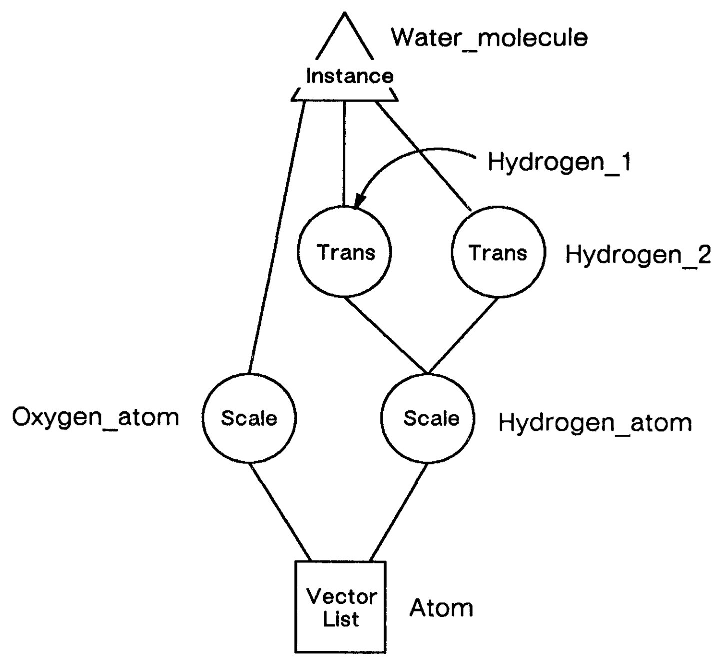

Figure 2-7. Display Structure for Water_Molecule

Data nodes are shown as squares, operation nodes as circles, and instance nodes as triangles. Nodes are connected by branches. The branches in the display structure determine what paths the display processor will take when it traverses the object's structure in memory. The content of each node in the structure determines either the operations the display processor will perform as it travels a branch or the data that will be transformed.

PS 390 display structures are more properly described as acyclic directed graphs. In a directed graph, an entity may have more than one ancestor. In PS 390 display structures, a data node may be the terminal node for more than one branch, as is the case for Atom in Water_Molecule. An acyclic graph does not allow recursion; that is, an element cannot refer back to itself. In PS 390 display structures, no path out of a node can be a loop back to that node.

6. Local Processing

The PS 390 was designed to perform locally all transformation processing, display processing, and interaction with the picture. Traditionally, the host computer has performed most of these operations.

6.1 Transformation Processing

The system performs most of the transformation and display processing involved in modeling and viewing objects without intervention from the user. This greatly reduces the graphics programming task and frees the programmer to concentrate on the design and analysis aspects of the application for which the PS 390 is being used. The main aspects of the transformation processing which the PS 390 performs are outlined below.

- Traversing Structures in Memory

  The programmer creates objects in memory as display structures with named nodes. The nodes define both the graphical data that determine primitive shapes and the transformations that are applied to the data. The programmer issues a single command to display a picture of the object. Then, during each refresh and update cycle, the system traverses the structure, performs all necessary transformations, and produces an image on the screen. The programmer does not have to issue commands for every step in this process.

- Matrix Concatenation

  All graphical transformations are applied to objects through a matrix called a transformation matrix. The display structure indicates the type of transformations and the order in which they are to be performed. The PS 390 performs the concatenation of matrices that this involves, so the programmer does not have to do this explicitly.

- Transformation and Clipping of the Data

  When the programmer manipulates an image on the screen using interactive devices, the system itself performs all the calculations needed to draw the image at each new position on the screen. The system traverses the display structure which defines the object, updating the image as new data is received from the interactive devices. If parts of the image move out of screen boundaries, the image is clipped automatically without any intervention from the programmer.

- Perspective Viewing

  Much of the "realism" of computer graphics is imparted by displaying views of objects in true perspective projection. The PS 390 produces perspective views of objects and maintains a true perspective view as the objects are in motion. The programmer creates a perspective view with a sequence of viewing commands. The processing which maintains the perspective view while the object is interactively manipulated is performed without intervention from the programmer.

- Viewport Mapping

  The PS 390 automatically performs viewport mapping. A viewport is an area of the screen in which pictures are displayed. The programmer defines a window, an area of world coordinate space that will be displayed on the screen. No matter what size of viewport the programmer specifies, the system maps the contents of the window to the viewport, ensuring that the view which the programmer wants will fit in the specified area of the screen.

  The PS 390 allows multiple dynamic and static viewports. Dynamic viewports allow interactive manipulation of objects displayed in them; static viewports allow objects to be displayed with various kinds of shading operations applied.

- Generating Characters

  The PS 390 supplies a standard character font of the characters, numerals, and marks and symbols that are usually found on typewriter keyboards. The programmer does not have to generate the vectors that compose a character in the standard font, although the PS 390 does allow the programmer to create, modify, and use other character fonts.

6.2 Display Processing

The main aspects of display processing which the PS 390 performs are outlined below.

- Digital to Analog Data Conversion

  Part of the display processing functions of virtually all CRT graphics devices is the conversion of digital data defining the vectors that need to be drawn to analog signals that drive the CRT. With the PS 390, this process is performed in the display processing hardware and requires no intervention from the programmer. The conversion of data from digital to analog happens very late in the processing cycle. The system keeps data in digital form through most of the display processing to enhance the processing speed and to allow the update rate to equal the refresh rate.

- Color Specification

  To generate colors, the programmer simply creates "color" nodes in the display structure, indicating the desired colors. The hardware itself does all the processing needed to display those colors.

- Blinking

  The PS 390 performs blinking as an integral part of display processing. The programmer enters a command with values for the number of frames to display the image and the number of frames to blank the display. The machine itself determines which lines to draw and which not to draw at any time.

- Picking

  Picking allows a graphics system user to select any portion of a randomly positioned three-dimensional image and get back information about the part of the object's structure it represents. The PS 390 performs this task with a minimum of intervention from the programmer.

6.3 Interactive Device Handling

The PS 390 lets users manipulate views of objects with all the interactive devices. These include:

- Keyboard with Function Keys
- Function Buttons
- Data Tablet
- Control Dials
- Optical Mouse

The devices are programmed through user-defined software known as function networks. These networks accept data from devices, manipulate the data as programmed in the network, and send new values to interaction points in a display structure of an object. The user programs the interactive devices by designing networks which manipulate the image as desired. The devices themselves convert analog data (such as the amount a dial is turned) into digital form.

6.4 Interacting With the Picture

Interaction with an object means changing the image on the screen. Typically, objects can be rotated or translated in any direction, and scaled by any amount along any axis. The PS 390 allows real-time interaction through the interactive devices. Real time means that when the programmer turns a dial that has been programmed to rotate an object, the delay between the dial turning and the object rotating is imperceptible.

Interaction is achieved by changing the values in an operation node or data node in the display structure. If new values are sent to a scale operation node, for example, the object will appear to grow smaller or larger on the screen. Interaction points in the structure accept new values from function networks connected to interactive devices.

6.5 Functions and Function Networks

The path between a device and an interaction node in a structure is a function network, created by the user to customize input from the interactive devices. A network is composed of individual functions, each function being thought of as a "black box" with inputs and outputs, as shown below.

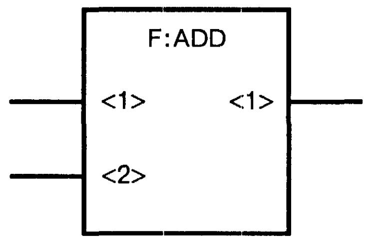

Figure 2-8. Representation of a Function

Each function performs a mathematical, logical, routing, or selecting operation, such as add, multiply, and, or, route, or choose. The programmer combines functions into networks which manipulate the data from interactive devices usually to produce an effect on a displayed image.

Function identifiers are of the form **F:name** and have numbered inputs and outputs which accept only certain data types. There are nine categories of functions available with the PS 390. They are as follows.

- Arithmetic and Logical

  These functions perform all arithmetic operations (add, divide, subtract, multiply, square root, sine, and cosine) and logical operations (and, or, exclusive-or, and complement).

- Character Transformation

  These functions are used to position, rotate, and scale text interactively.

- Comparison

  These functions test whether values are greater than, less than, equal to, not equal to, greater than or equal to, and less than or equal to other values.

- Data Conversion

  These functions change matrices into rows, rows into scalar elements, and real numbers to integers or vectors. Data can be output in decimal or exponential format.

- Data Input and Output

  These functions set up and control the interactive devices (dials, function keys, function buttons, data tablet, and keyboard) and output values to the optional LED labels which several of the devices have. They also handle communications with the host.

- Data Selection and Manipulation

  These functions are used to switch functions selectively, choose outputs, and route data.

- Object Transformation

  These functions connect to modeling operation nodes in display structures to rotate, translate, and scale objects interactively.

- Viewing Transformation

  These functions connect to viewing operation nodes in display structures to change line-of-sight, window size, and viewing angle, interactively.

- Miscellaneous

  Other functions set up and control clocking, timing, and synchronizing operations.

Most functions are general purpose but have been designed specifically for graphics operations. For example, an arithmetic function such as "multiply" not only accepts two scalar numbers as inputs and outputs their product, it also performs matrix multiplication, accepting two matrices and outputting a concatenated matrix.

Using the set of "master" functions as templates, the programmer creates uniquely named instances of functions and connects them to form a network using PS 390 commands. A function in the network accepts data from an interactive device or another function, performs its specific operation, and outputs the result to another function in a network or to an interaction point in a display structure. The flow of data through a function network is shown in Figure 2-9.

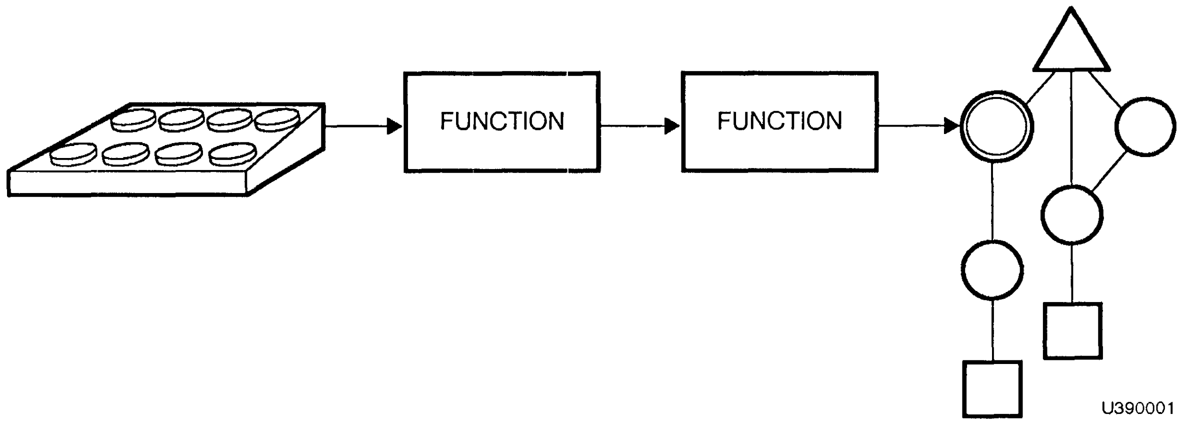

Figure 2-9. The Flow of Data Through a Network

Function networks differ from programs written in conventional programming languages to handle data from interactive devices in that function networks are data driven. That is, networks are dormant until a function receives data at its input queues. Then the function becomes active, processes the data, passes on the output, and becomes dormant again. In this way, the computer does not have to spend time polling the interactive devices to see if any activity has occurred.

IS3. OPERATION AND COMMUNICATION

CONTENTS

ILLUSTRATIONS

Section IS3
Operation and Communication

This section instructs a PS 390 user in how to operate the PS 390 Graphics System. When different types of host systems have operational differences, the operation is documented in separate subsections, one subsection for each host system. For example, some keyboard operations are different depending on whether the host is a DEC or an IBM system.

The first part, *Booting*, gives instructions for booting the PS 390.

The second part, *Confidence Testing*, provides a general discussion of the power-up confidence tests.

The third part, *Graphics Firmware*, gives a specific description of the Graphics Firmware usually loaded at boot time. Included in the description are: the CONFIG.DAT file, the SITE.DAT file, and methods that are available to conform PS 390 firmware or software to specific site needs.

The fourth part, *Interactive Devices*, includes a brief description of the interactive devices that can be used with the PS 390 Systems: keyboard, control dials, data tablet, function buttons, and optical mouse.

The fifth part, *Operation*, deals with the general operation of the PS 390 display and keyboard, and the features available with the PS 390 terminal emulator package.

The sixth part, *Communication*, discusses PS 390 host communication and PS 390 local communication, including the E&S supported software package that uses host-resident routines to prepare data prior to sending it to the PS 390.

1. Booting the PS 390 Graphics System

Please read the following information carefully before installing any diskette in the floppy diskette drives or turning on the PS 390 control unit and display. PS 390 firmware requires two floppy diskettes. These diskettes are booted from the two drives on the front of the control unit.

1.1 Preparation

Booting the PS 390 is a simple procedure. Before starting the booting process, please check the following.

1. Make sure that the control unit, display station, and peripheral multiplexer are turned off. The "ON-OFF" power switch for the control unit is located on the front panel of the control unit on the upper-right side. This is a toggle switch. The "ON-OFF" power control for the display station is the lower pushbutton switch located on the right side of the display. "ON-OFF" power for the peripheral multiplexer is a toggle switch located on the far right side of the front panel of the mux box. The control unit has an orange power-on indicator behind the ON-OFF switch; the display station has a green power-on indicator on the bottom front of the display.
2. Check that the control unit, display station, and peripheral multiplexer power cords are plugged into active wall outlets of adequate capacity.
3. There should be either a keyboard with LED displays or an auxiliary ASCII terminal available to monitor the progress of the PS 390 confidence tests during the power-up sequence if necessary, and to allow commands to be entered once the PS 390 is booted.
4. The peripheral multiplexer must be connected to port 5 on the control unit.
5. The display AC cable should be connected to the peripheral multiplexer. The video cable from the display must be connected to the control unit.
6. The interactive devices should be connected to the peripheral multiplexer.

1.2 Booting The System

1. Install the appropriate PS 390 system diskettes in the disk drives. For basic PS 390 graphics operations, use the current version of the PS 390 graphics firmware diskettes (Not the DEMO diskettes). To install the diskettes:

   - Push down the latch on the left-hand drive (drive 1), and with the write-protect slot up (usually covered with foil) and the manufacturer's or E&S label facing right, install the firmware labeled "DISK A" in the drive.

- Push the latch up to a horizontal position when the diskette is in place. The latch must be closed to activate the drive.
- The right-hand drive is labeled as drive 2. Install the diskette labeled "DISK B" in drive 2, observing the same disk orientation as used with DISK A. Push the latch up to a horizontal position.

CAUTION

While the system is booting, do not transmit from the host to the PS 390, press any keys on the keyboard or auxiliary terminal, or use any interactive device.

2. Turn on the PS 390 control unit, the display, and the peripheral multiplexer to initiate the booting sequence.

3. The PS 390 begins a series of power-up confidence tests as soon as the control unit is turned on. Refer to section 2 for monitoring these tests. These tests output alphanumeric sequences beginning with

```
A B C D E F G H I J K L M N O P 0 1 2 3 4 5 Q R 0 1 2 3 S T
```

The characters A through O will appear only on a debug terminal set to 300-baud. A beep sounds after the successful completion of several of the tests (before "J" and at "O"). The remaining characters of the sequence can be seen on either the keyboard LEDs or the auxiliary terminal set to 9600-baud. When the confidence tests are complete and the files on the firmware (or other) diskettes have been loaded, the keyboard LEDs or the auxiliary terminal will display the following:

```
PS 390:  System VXXX A: (date) I: (date)
```

The version number (XXX) and dates depend on the version of the software that is booted.

1.3 Trouble-shooting Tips

If the booting process is not complete after three to five minutes, something is wrong. There are several things that should be checked before trying to boot the system again.

1. If, after installing the diskettes and turning on the control unit, display, and peripheral multiplexer, the system goes longer than 50 seconds without beeping, a confidence test has failed and the booting process should be repeated. At this point it is best to perform a "warm" reset by pressing the RESET button located on the front of the control unit next to the ON/OFF switch. If the confidence tests still fail to complete, start over by turning off the control unit and taking out the diskettes. Then reinsert the diskettes, close the drives, and turn the control unit back on.
2. Make sure the display and peripheral multiplexer are turned on.
3. Check to make sure that the diskettes are properly inserted and that the drive latches are closed.
4. Check all power and cabling connections to make sure that host line(s) and debug terminal line(s) are in correct ports for the firmware being booted.
5. Check to make sure that you have the proper diskettes in the drives.
6. If the confidence tests run successfully through "O", but the diskette names and the version number do not appear on the display, the problem is related to trouble in reading the diskette files. If you have other copies of the diskettes, attempt to reboot using them.

2. Confidence Tests

Before the firmware is loaded into the system, a series of hardware confidence tests is performed. The status of this testing is indicated as a series of alphabetic characters, with each character reporting the successful completion of a different portion of the test. The start-up confidence tests are located in ROM in the Joint Control Processor (JCP). They are not booted with the PS 390 Graphics Firmware.

The debug terminal should be set to 300-baud to monitor these confidence tests. It is not necessary to monitor the initial confidence tests unless the system fails to boot properly.

If the hardware confidence tests are monitored, all letters of the alphabet A through O should appear on the 300-baud debug terminal as the confidence test steps are completed and the firmware is loaded. The letters appear at varying time intervals because some tests take longer than others. The audible alarm sounds upon successful completion of the confidence tests. Further characters that appear after the letter O are not related to the confidence test phase of power-up, and are under control of the program loaded from the diskette. These tests can be monitored on the keyboard LED displays or the auxiliary terminal.

If the system goes longer than 50 seconds without issuing the audible alarm, a confidence test step has failed, and the power-on sequence must be repeated. The last letter appearing on the terminal should give a clue as to what went wrong. For instance, if the last character appearing is L, the diskette is probably not mounted in the drive.

Failures after the confidence tests complete (after O is displayed) are almost always related to problems reading the diskette files.

The confidence tests are initiated when the JCP receives a Reset signal while in RUN mode. This reset may be generated either by a power-up condition or by pressing the RESET button. The confidence tests evaluate a minimal hardware configuration and then read in an executable program from the PS 390 floppy diskette. Because some components must work for the confidence tests to run, several components are assumed operational at this point: the MC68000 microprocessor, the two EPROMs, a serial port, and the logic between these components.

The confidence tests perform the following operations:

- Initialize serial output ports.
- Perform a quick check of the MC68000 microprocessor.
- Perform a checksum operation to verify EPROM.
- Verify memory refresh.
- Verify error detection and correction.
- Test local memory.
- Test the floppy diskettes.
- Load the boot file from the diskettes and begin executing.

When the entire test is completed, a BEL character sounded from the keyboard provides an audio indication that the system is functioning correctly. Control is then passed to a program loaded in from the floppy diskette.

The non-resident code that is read in may continue with the self-testing, as the Graphics Control Program does, or the non-resident code may be an operating system for diagnostics. Whenever a confidence test fails, Debug mode is entered automatically. Whenever the program designated to be read in cannot be located on the diskette, the Debug mode is entered automatically on the debug terminal connected to Port 3. If there is no debug terminal, the user will not see that the debug mode has been entered.

2.1 Graphics Firmware Self Tests

After the confidence tests have been completed successfully, self-testing continues under control of the program code that was loaded in from the diskette. This code tests and/or initializes all PS 390 components not tested or initialized by the confidence tests (i.e., all serial ports, the clock timer, mass memory). If the system fails to initialize these properly, the "runtime debug" mode is entered and the system prompts the user to see if a restart is desired. Once the entire system has been tested and initialized, the Graphics Control Program is loaded and the PS 390 enters the normal mode of operation.

3. The PS 390 Graphics Firmware

The PS 390 Graphics System Firmware is distributed on 5 1/4-inch floppy diskettes. All versions of the system firmware contain files that perform the described functions.

Microcode defines the functions of the Display Processor.

Runtime code defines the PS 390 command language and functions. This is made up of various files and includes:

- THULE.DAT file contains code that is loaded into mass memory.
- CONFIG.DAT file links the microcode and the runtime code into a coherent system function network.
- SITE.DAT file allows users to enter PS 390 commands into a readable file to change system default parameters for the specific site. This is not a required file.
- INTFCFG.DAT file allows the user to boot the PS 390 with different interface options.

3.1 The Runtime Code

The runtime code contains all definitions for system level commands and functions, as well as the definitions for user-accessible commands and functions. These definitions are loaded into the JCP local memory.

During the loading of the PS 390 firmware, all intrinsic system and user functions are "instanced" by the system. "Instancing" is the process of creating a unique case of the function. This case includes all the information necessary to identify the specific case, including an index pointing to the Pascal-callable routine that performs the required function, the input source(s), the output destination(s), the priority value assigned for scheduling purposes, and the current status of the function.

User-accessible functions and commands are documented in the Section *RM1 Command Summary* and Section *RM2 Intrinsic Functions.*

3.2 The CONFIG.DAT File

The CONFIG.DAT file contains the instances and connections to link the system functions into a system network. This file is written in the PS 390 command language and is similar to any function network built to manipulate graphical data. The design of the configuration file allows for flexibility in changing the initial environment; the change to the file is loaded and read from the disk, rather than being a change to a program. This means that changing the design features of a system does not require recompiling and relinking code and it reduces the size of executable code in the system.

The CONFIG.DAT file is interpreted by an instance of the command interpreter. While reading this file, the command interpreter is in a privileged mode of operation called the "configure mode." The command interpreter must be in the configure mode to configure any system functions.

One of the final commands in the CONFIG.DAT file is a command that will attempt to read the SITE.DAT file, which may or may not reside on the firmware diskette, depending on whether it has been written to the diskette. This file normally contains commands to tailor PS 390 system default parameters to the requirements of that particular site.

The command interpreter will read the SITE.DAT file and exit the CONFIGURE mode. If no SITE.DAT exists on the diskette, the command interpreter will execute the final commands in the CONFIG.DAT and then exit the configure mode.

3.3 The SITE.DAT File

The final file on the PS 390 Graphics Firmware is the SITE.DAT file. This file is not required, but when used enables the user to change default features for the PS 390 system in a bootable file. The file is assumed to contain a stream of ASCII commands. Because of limited disk space, the file can only contain a small number of commands.

The SITE.DAT file allows users to store information across power-up sequences. The actual values and parameters that may be set or changed using the SITE.DAT are described in the pertinent sections.

Section *RM5 Host Communications*, describes the port values that may be changed.

Section *RM7 Host Input Data Flow*, describes certain data packet and routing characters that can be changed.

Section *RM10 Terminal Emulator*, describes the keyboard and display features that can be changed or set.

There are three methods for creating and installing the SITE.DAT file:

- Creating the SITE.DAT file (an ASCII file) on the host and downloading to the PS 390 over an asynchronous line.
- Entering the PS 390 command mode and typing the SITE.DAT commands directly to the diskette from the PS 390 keyboard. This method is not recommended since it involves writing directly to diskette and does not provide for error correction.
- Using the GSRs.

Refer to Section *TT2 Helpful Hints* for instructions on creating the SITE.DAT file.

4. Interactive Devices

This section contains descriptions and typical uses of interactive devices that can be configured with the PS 390. Two sets of interactive devices are available with the PS 390: the PS 300-style devices and the PS 390-style devices. Interactive devices from the two styles cannot be mixed with the exception of the data tablets and the optical mouse. These devices are common to both styles. With each device description there are operational instructions.

Setup procedures, brief maintenance information, and a description of the control codes generated by the interactive devices are provided in Section *RM13 Interactive Devices*.

The interactive devices that can be configured with the PS 390 include the following:

PS 300 Style Devices

- Peripheral multiplexer
- Keyboard with LEDs
- Control dials unit with LEDs
- Data tablet (6X6) with 4-button cursor
- Data tablet (12X12) with 4-button cursor
- 32 Function buttons unit
- Optical mouse

PS 390 Style Devices

- Peripheral multiplexer
- Keyboard without LEDs
- Control dials unit without LEDs
- Data tablet (6X6) with 4-button cursor
- Data tablet (12X12) with 4-button cursor
- 32 Function buttons unit
- Optical mouse

The acceptable inputs and programmable outputs from these devices are found in Section *RM2 Intrinsic Functions*, and Section *RM3 Initial Function Instances*. The interactive devices are sensitive and should be handled and stored in an appropriate manner. Normal considerations should be taken in use; liquids or foreign objects spilled or dropped on any of the devices may damage them or cause them to malfunction.

4.1 Keyboard and Dial LED/Softlabel Display Operation

The PS 300-style keyboard and the PS 300-style control dials unit have LED displays. These are used to display character strings that serve the following purposes:

- Fatal error messages are displayed on the keyboard LEDs.
- The LEDs can be programmed by the user to name or display information about the use of the function key or control dial located directly below each LED display segment.

Most users will use PS 390 function networks to set up and send character strings to the LED display. Information for sending characters and text to the keyboard LEDs is found in Section *RM2 Intrinsic Functions* and Section *RM3 Initial Function Instances.*

On the PS 390 style keyboard and the PS 390 style control dials unit a "softlabels" network can be used to display function key labels and the control dial labels. The softlabels network redefines the dynamic viewport on the PS 390 to allow the left edge of the display to be used as a display area for the labels. Refer to Section *TT2 Helpful Hints* for instructions on using softlabels.

4.2 The Keyboard Function Keys

The keyboard function keys are the top row of 12 keys on the two keyboard styles. These keys typically act as graphics input devices to user function networks. They can be used for internal control of or communication back to an application program on the host. The mode of the keyboard also determines the function of the keys. In local mode, the functions keys supply values to user designed function networks; in terminal emulator mode the function keys are configured to act as input keys that send data to the host system. The keys can be set to act as graphics input devices (as in local mode) in all modes by using the terminal emulator SETUP feature.

These keys can be used to do such things as:

- Switch the input of the control dials between function networks
- Send specific information back to an application program
- Determine the function of other interactive devices (e.g., control dials, data tablet).

4.3 Control Dials Unit

The control dials unit consists of eight dials with LED displays on the PS 300 style and no LEDs on the PS 390 style. The dials are used to communicate dynamic, incrementing, and decrementing data to the PS 390. The control dials unit connects to the port beneath the DIALS icon on the peripheral multiplexer.

In typical applications, the control dials can be used to perform the following operations.

- Rotate objects about the X, Y, or Z axis, each type of rotation typically using a different dial.
- Zoom in or out.
- Translate objects in X, Y, and Z, each translation typically using a different dial.

Refer to Section *GT6 Function Networks I* for information on using the control dials as an input device to user-created function networks.

The information that describes system-level data formats and codes exchanged in dial and LED display operation is found in Section *RM13 Interactive Devices.*

4.4 Data Tablet

The data tablet consists of a tablet and a four-button cursor called a "puck." Puck position information is sent to the PS 390 in digital form that expresses a two-dimensional coordinate value (X,Y). The data tablet connects to the port below the TABLET icon on the peripheral multiplexer.

Maps, diagrams, and drawings may be used as overlays on the surface of the tablet.

The operating mode and sampling rate of the PS 390 data tablet are both controlled by the JCP. The following modes are available.

- Point Mode - Pressing the puck button at a given tablet location causes one X,Y coordinate pair (sample) to be transmitted.
- Stream Mode - X,Y coordinate pairs are generated continuously at the selected sampling rate when the puck is near the tablet surface.
- Switched Stream Mode - Pressing the button on the puck causes X,Y coordinate pairs to be output continuously at the selected sampling rate until the puck button is released.

4.5 Function Buttons Unit

The function buttons unit is a set of 32 programmable buttons. Each button is backed by a light that can be used to indicate the button setting. The function buttons unit is connected to the port below the BUTTONS icon on the peripheral multiplexer.

The only controls on the function buttons unit are the 32 buttons. They are arranged with one row of four buttons, four rows of six buttons, and a final row of four buttons. The buttons are not numbered but are counted from left to right, beginning at the top row of four buttons. A button is triggered by pressing it down and releasing it. The lights backing the buttons are programmable and can be set to on or off.

As with the function keys on the keyboard, the function buttons are used to send inputs to user-created function networks. Typical applications are selecting menus and selecting data structures for display.

4.6 Optical Mouse

The Optical Mouse consists of a three-button mouse unit with a reflective pad. The optical mouse transforms X and Y-axis position information to a digital form acceptable to the PS 390. The optical mouse unit is connected to the port below the MOUSE icon on the peripheral multiplexer.

The control logic in the mouse translates directional information into relative X and Y movement information. The data is transmitted serially to the PS 390 through the peripheral multiplexer.

Interfacing the mouse to an application can be accomplished with the MOUSEIN initial function instance.

5. Operation

The operational instructions for the PS 390 begin with the very basic features of the PS 390 display and keyboard. It is important to read the information to get an idea of the design and implementation of the system.

5.1 The PS 390 Display

The PS 390 display has no particular operating instructions other than turning it on and off, and adjusting the brightness and contrast of the screen.

The display should be turned on before powering up the PS 390 control unit. The display is turned on using the ON/OFF switch located on the lower right-hand side of the display. The ON/OFF switch is the bottom of the two switches. The switch above the ON/OFF switch is the DeGauss switch. The two thumbwheel knobs above the switches are used to control contrast and brightness to reach a desired intensity. The top thumbwheel regulates the contrast; the bottom regulates brightness. Intensity is controlled by the use of both these controls.

The main area of the screen is used for viewing text or graphics. The terminal emulator viewing area is 80 columns wide and 24 lines high. The screen space used for viewing is dependent on the application. (Refer to section 5.3.5 below for information on adjusting the viewing area of the screen.) The graphics viewing area is a 10.5 inch square in the center of the screen.

The bottom line on the PS 390 screen is the message display area. This line will display a memory-alert warning message when the user is running out of mass memory space and can be used to display messages generated by a user application program. The line can be "blanked" by the user. When memory usage drops below the alert status, the message display area will automatically blank.

5.2 The PS 390 Keyboards

The PS 390 has two available keyboards: the PS 300 style keyboard and the PS 390 style keyboard. Both keyboards operate as a standard ASCII keyboard. The PS 300 style keyboard has an optional LED array located on the top of the keyboard panel that can display a full line of text or text segments. The LEDs can be used to label the action performed when a function key is selected. They also display error messages generated by a fatal error and provide visual feedback during the runtime self-tests.

The keyboard is activated when the system is booted. The keyboard connector cable should be plugged into the appropriate port on the peripheral multiplexer. The keyboard's grounding wire must be connected to the grounding screw on the front of the peripheral multiplexer box.

The keyboard keys fall into eight categories.

- **Keyboard Function Control Keys** — These keys are the CTRL (control), SHIFT, CAPS LOCK, and REPT keys. They are local control keys that modify the signal generated by other keys when struck in combination with them.

- **Alphabetic Keys, Standard Numeric, and Special Character Keys** — These keys all generate standard ASCII character codes and are used to display uppercase and lowercase characters. The keys may be struck alone, or in combination with the keyboard function control keys.

- **Terminal Function Keys** — These keys produce codes used by a standard terminal. They are ESC, TAB, BACKSPACE, DEL, RETURN, LINE FEED, and the space bar.

- **Numeric Keypad Keys** — The function of these keys is determined by the host system and the mode in which they are used.

- **Device Control Keys** — These are system keys and are used to activate certain applications of the PS 390 in the various modes of the terminal emulator.

- **Function Keys** — These keys are interactive device keys that are set up by user network functions. They can be used for internal control or for communication back to the application program in the host.

5.2.1 Keyboard Modes Of Operation

The keyboard initiates actions that are defined by the mode of operation. These modes are entered by pressing one or more keys on the keyboard. The availability of various keyboard modes is in part determined by the host system. Table 3-1 describes the modes of operation for systems with a PS 300-style keyboard, and the key(s) used to access them. Table 3-2 describes the modes of operation for systems with a PS 390-style keyboard, and the keys used to access them.

Table 3-1. Keyboard Modes with PS 300-Style Keyboard

MODE	USE	KEY(S)TO ACCESS	KEYBOARD APPLICATION
Terminal Emulator	Access line to host	LINE_LOCAL	Keyboard input sent to host and characters displayed on PS 390 screen.
Command	Enter direct PS 390 commands	CTRL/LINE_ LOCAL	Text commands are displayed on the screen and executed by the command interpreter.
Local	Keyboard as graphics device	SHIFT/LINE_ LOCAL	Fkeys and Keyboard are activated as input devices for function networks. Text does not appear on screen.

Table 3-2. Keyboard Modes with PS 390-Style Keyboard

MODE	USE	KEY(S)TO ACCESS	KEYBOARD APPLICATION
TE	Access line to host	CTRL/HOST or ALT/HOST	Keyboard input sent to host and characters are displayed on screen
5080	Run 5080 applications	CTRL/5080 or ALT/5080	Applies to IBM 5080 Interface only
Command	Enter direct PS 390 commands	CTRL/CMND or ALT/CMND	Text commands are displayed on the screen and executed by the CI command interpreter
Local	Keyboard as graphics device	CTRL/LOCAL or ALT/LOCAL	Fkeys and keyboard are activated as devices for function networks

5.2.2 Key Sequences and Functions Of Interest

The following key sequences and their functions are active in all keyboard modes on all host systems.

- GRAPH and TERM - The text associated with either the terminal emulator or the command mode can be blanked by pressing the TERM key (PS 300 style) or CTRL/TERM (PS 390 style). Pressing the key sequence a second time will restore the text. Pressing the GRAPH key (PS 300 style) or CTRL/GRAPH (PS 390 style) blanks any graphics being displayed. Pressing the key sequence a second time will restore the graphics display. These keys may be used at any time, and allow the user to view both the text and the graphics display simultaneously, or to clear one or both from the screen.

- CLEAR HOME (PS 300-style keyboard only) - When used alone, this key erases the terminal emulator text from the screen and places the cursor in the top left corner of the screen. When used with the SHIFT key, the cursor is moved to home, but the screen is not cleared. When used with the CTRL key, the screen is cleared, but the cursor is not moved to home. This functionality is not supported with the PS 390-style keyboard.

5.3 Keyboard Modes for DEC Systems

The following section is directed to users with a DEC/VAX or PDP-11 host system or systems that are functionally similar to the DEC systems.

The PS 390 terminal emulator (TE) package provides the user with the ability to use the PS 390 display and keyboard as interactive graphics devices and allows for DEC VT100 emulation on the PS 390. Both the command mode and the local mode of the PS 390 require no host system interaction.

The following sections describe the three keyboard modes available for VT100 emulation on the PS 390 and how to access them.

5.3.1 VT100 Terminal Emulator Mode

In the terminal emulator (TE) mode, the PS 390 terminal functions as a standard terminal on the host. The user can log on to the host, access and edit host-resident files, and use the available host system utility commands.

Upon entering the TE mode, the RETURN key should be pressed at least once to generate a host prompt character on the screen. The cursor, a blinking square character, will appear on the active line in the active column. The PS 390 powers up in the TE mode.

5.3.2 VT100 Command Mode

In command mode, commands entered on the PS 390 keyboard are displayed as text on the screen. While in command mode, valid commands with the proper ";" terminator are immediately interpreted by the PS 390 and are not stored by the system or host in a file. Error messages received from the parser and the command interpreter are displayed on the screen.

The RETURN key should be pressed after entering the command mode to generate the PS 390 command prompt '@@' and the cursor.

5.3.3 VT100 Local Mode

In local mode, the function keys on the keyboard (and any other programmed keys) act as local input or selection devices for user-constructed PS 390 function networks. There is no cursor or screen prompt in the local mode, and the keyboard does not send any information to the host.

5.3.4 General Information

Keyboard activities are dependent on the mode used. The following list is an easy reference for keyboard use in various modes.

1. In the terminal emulator mode, movement of the text on the screen is host dependent.
2. In command mode, text appears on the first available line and moves down the screen. Text scrolls up when the last line is filled. By default, there is no automatic line wrap-around; text terminates at 80 columns. This can be changed, and is dependent on host system configuration.
3. The function keys generate certain escape sequences when used in modes other than local mode.

5.3.5 Terminal Emulator SETUP

The terminal emulator SETUP facility allows users to change some features of the display and keyboard. SETUP also gives the user access to a menu display of the current keyboard configuration. The SETUP facility is accessed by pressing the SETUP key (PS 300-style keyboard), or CTRL/SETUP (PS 390-style keyboard). While in SETUP, the SETUP menu is displayed on the PS 390 screen. All keyboard input is sent to the SETUP facility, and the only active keys are the PS 390 device keys and the keys that the SETUP facility uses. SETUP may be entered from any of the three keyboard modes.

The status of features that have been changed using SETUP are in effect until changed by the user or until the system is rebooted. When the system is rebooted, the default values of SETUP are reinstated. The default values of the features in SETUP may be changed by entering the new values into the SITE.DAT file on the PS 390 graphics firmware. Refer to *Section TT2 Helpful Hints* for information on creating and using the SITE.DAT file.

5.3.6 SETUP Menu Display

The following menu is displayed when the SETUP key is pressed.

```
SETUP

F2=SRM:T  F3=AWRP:F  F4=ANSI:T  F5=VT52:F
F6=KPM:F  F7=CKM :F  F8:Cnum:T  F9=Knum:T

F10= Define break key :↑V
F11: Move TE viewport, lower left corner
F12= Move TE viewport, upper right corner
Mode: KB  Term: On   Graph : On
```

The status of the features (T or F) is displayed on the screen when the SETUP menu is displayed. With the exception of defining the BREAK key and setting the screen viewport for text, all the features are set or reset by pressing the indicated function key. The status of the feature changes (from T to F, or F to T) on the screen menu when the key is pressed. When additional keystrokes are required, as in setting the viewport location and size, additional prompts are displayed after the function key that accesses that feature has been pressed. Table 3-3gives a breakdown of the function keys, their purpose, and the default values assigned to the features by the system.

Table 3-3. Terminal Emulator SETUP Features

FUNCTION KEY	FEATURE	DEFAULT VALUE
F2	SRM mode	TRUE
F3	Autowrap	FALSE
F4	ANSI mode	TRUE
F5	VT52 mode	FALSE
F6	Keypad mode	FALSE
F7	Cursor keys	FALSE
F8	Numeric keys in command mode	TRUE
F9	Numeric keys in keyboard mode	TRUE
F10	Define BREAK key	User defined
F11	Viewport lower left corner	User defined
F12	Viewport upper right corner	User defined

5.3.7 SETUP Feature Definitions

The following features may be set to ON or OFF by toggling the appropriate function key in the SETUP Mode. To exit SETUP, use the SETUP key (PS 300 style) or the CTRL/SETUP sequence (PS 390 style).

Function key F2 (SRM) sets or resets the send-receive mode of the PS 390 terminal. This mode determines whether the input to the screen from the keyboard is sent via the host or a PS 390 system function. (Refer to Section *RM10 Terminal Emulator* for more information on this feature.) The send-receive mode is set to TRUE by the system and may be changed by the user at any time.

Function key F3 (AWRP) lets you set or reset (on or off) automatic line wrap-around at 80 characters.

Function key F4 (ANSI) sets the ANSI (VT100) mode of the PS 390 terminal. When set (or TRUE) the PS 390 will generate and respond to VT100

(ANSI) control sequences. This mode defaults to TRUE and should only be changed when the user wants the PS 390 to respond like a teletype-style terminal and not respond to VT100 control sequences.

Function key F5 (VT52) allows the PS 390 to respond VT52 coded sequences. This defaults to FALSE.

Function key F6 (KPM) determines the function of the numeric keypad on the PS 390 keyboard. If set to TRUE, the numeric keypad will generate the control sequences used in the keypad application mode (i.e., host editing utilities such as DEC's EDT and KED). When set to FALSE, the numeric keypad generates its numeric keycap values.

Function key F7 (CKM) sets the cursor keys mode. When this mode is TRUE, the cursor keys are operational in all keyboard modes. The ANSI/VT52 feature must be set to TRUE for this to be effective.

Function key F8 (Cnum) determines whether the numeric keyboard will generate keycap values or escape sequences in command mode. When set to TRUE, the keypad will generate the numeric keycap values in command mode.

Function key F9 (Knum) determines whether the numeric keyboard will generate keycap values or escape sequences in local mode. When set to TRUE, the keypad will generate the numeric keycap values in local mode.

Function key F10 (BREAK) is used to designate a key to send a break sequence to the host system. It is up to the user to decide which PS 390 key will be interpreted as a BREAK key by the host. To designate the BREAK key, first press function key F10, and then press the key that is to be used as the BREAK key. After pressing the designated BREAK key, press function key F1 to indicate the BREAK key has been selected and to return to SETUP Mode.

Any key may be used as a BREAK key, with the exception of those listed below.

- SETUP
- Function key F1
- GRAPH key
- TERM key

- LINE LOCAL key (PS 300 style)
- HOST (PS 390 style)
- CMND (PS 390 style)
- LOCAL (PS 390 style)

Any other function key, and any unspecified keys on the numeric keypad can be designated as the BREAK key. The BREAK key can be designated as a single key sequence, the shifted value of a key, or the control value of a key.

A break key time (duration) must also be defined in order to send a break sequence to the host. Defining this time is done by using a SETUP INTERFACE option. Normally, the following command would be used:

```
Setup Interface Port 10/Break_Time = 127;
```

Instead of a key, users may designate a CTRL/V (single character) escape sequence that will cause the terminal emulator to send a break to the host. A restriction on this sequence is that it must not be one that is generated by the E&S keyboard. Section *RM13 Interactive Devices* contains the escape sequences generated by the keyboard.

The BREAK key is only functional in the terminal emulator mode of operation.

Function keys F11 & F12 (Viewport). When the PS 390 System is booted, the display screen is initialized with a terminal emulator screen area that is 24 lines high and 80 columns wide. When the size and location of the viewport is changed, the viewport still contains the 24 horizontal lines and the 80 vertical columns, but the size of the rows and columns can be changed. For example, if the upper-right corner is moved from its original position to the top right-hand corner of the PS 390 screen, this does not add more available lines for text or display, but rather transforms the text appearing on the 24 lines. This text will appear to have twice the height.

The viewport can be made any size and any rectangular shape and can be placed anywhere on the PS 390 screen. To remove any previous text from the screen, use the TERM key (PS 300 style) or CTRL/TERM key sequence (PS 390 style). The SETUP menu is then visible.

There are two steps needed to change the viewport size and/or location:

1. To move the upper-right corner, first press function key F12, then the cursor key or keys that move the corner in the desired direction. As there is no cursor in SETUP, use the TERM key or CTRL/TERM key sequence again to get some text on the screen. As the cursor key is pressed, the text on the screen conforms to the corner position. When the appropriate position is reached, press function key F1.
2. To move the lower-left corner, first press function key F11, then the appropriate cursor key or keys. Again, have some text on the screen so that the movement of the corner is visible. When the lower-left corner reaches the desired position, press function key F1.

When the right corner of the text display area meets the left corner, or the bottom edge meets the top edge, the sizing action stops until cursor keys are pressed that move the corners in an appropriate direction. The movement of the corners also stops when they reach the edge of the display area of the screen.

5.4 Keyboard Modes For IBM 3278

The following section is directed to users with an IBM host system with the IBM 3278 emulator. The 3278 Interface option uses the PS 390 style keyboard only, and allows the keyboard to function in three modes: terminal emulator, command, and local mode.

5.4.1 Terminal Emulator Mode

In terminal emulator mode, the keyboard closely emulates the IBM 3278 Model 2 terminal.

In terminal emulator mode, the user can log on to the host, access and edit host resident files, and use the available host system utility commands. Files containing PS 390 commands can be built on the host and sent to the PS 390 using the GSRs.

Once exited, the terminal emulator mode is re-entered by pressing CTRL/HOST or ALT/HOST.

5.4.2 Command Mode

Command mode is used to enter PS 390 commands locally. To enter command mode, press the CTRL or ALT key, and while holding it down press the HOST key. The RETURN key is pressed to generate the PS 390 command prompt '@@' and the cursor. In command mode, PS 390 commands are displayed as text on the screen. Valid commands with the proper terminator are immediately processed and are not stored by the system. Error messages received from the parser and the command interpreter will be displayed on the screen.

In the command mode, the delete key functions as a CRT type delete.

Pressing the delete key backspaces and deletes the previously entered text on the active or bottom line until the key is released. The backspace key is not operational in command mode.

5.4.3 Local Mode

Local mode allows the function keys and standard keyboard keys to act as inputs to any user-created function networks that are connected to them. There is no cursor or screen prompt in the local mode and the keyboard will only act as an input device to function networks. To enter the local mode, press the CTRL/LOCAL or ALT/LOCAL keys.

5.4.4 Terminal Emulator Displays

Additional indicator characters which give the status of the PS 390 are included in the IBM terminal emulator displays. They appear on the right side of the indicator line. They are defined as follows:

- H — Terminal emulator mode
- C — Command mode
- L — Local mode
- S — Setup mode
- G — Indicates that graphics display is enabled

5.4.5 SETUP For Terminal Emulator

The SETUP mode for the 3278 terminal emulator is accessed by pressing the <CRTL> SETUP. SETUP can be entered from any mode and is used to make the following adjustments.

Intensity – Function key #1 brightens the screen, Function key #2 dims the screen.

Contrast – Function key #3 raises the contrast, Function key #4 lowers the contrast.

The SETUP key sequence must be executed again after the appropriate adjustments are made to exit the SETUP mode.

5.4.6 General Information

The display and 3278 keyboard activities are dependent on the mode used. The following list of differences in mode operation for the keys and the display will provide an easy reference for keyboard and display use.

1. In the terminal emulator mode, movement of the text on the screen is host-dependent. Refer to the IBM 3278 operator's manuals for more information.

2. In command mode, text appears on the first available line and moves down the screen. Text scrolls up when the last line is filled. There is no automatic line wrap-around; text terminates at 80 columns and is typed over until a carriage return is performed. The Delete key is active for text editing. The ENTER key is not functional. The RETURN key should be used as the PS 390 line delimiter key, but a line is not sent to the command interpreter until the ";" command delimiter is entered.

5.5 Keyboard Modes for IBM 5080 Interface Option

The PS 390/IBM 5080 Capability Option allows the keyboard to operate in four modes. When a particular mode is in effect the system displays an indicator character to signal the operator.

5.5.1 5080 Mode

The 5080 mode is used to run 5080 application programs downloaded from the host. When this mode is entered, a 5 appears at the bottom right-hand corner of the screen.

5.5.2 Command Mode

Command mode is a PS 390 mode and is entered by pressing the CTRL or ALT key in conjunction with the CMND key. When this mode is entered, a C appears at the bottom right-hand corner of the screen.

5.5.3 Local Mode

Local mode is entered by pressing the CTRL/LOCAL or ALT/LOCAL keys. When this mode is entered, an L appears at the bottom right-hand corner of the screen.

5.5.4 Terminal Emulator Mode

Terminal emulator mode is used to communicate with the host system. Terminal emulator mode is entered by pressing the CTRL/HOST or ALT/HOST keys. The letter H appears at the bottom right-hand corner of the screen when terminal emulator mode is active.

6. Communication

The PS 390 does not rely on the host to perform any graphical manipulations, nor the mathematical calculations behind the manipulations. Also, the PS 390 is host independent in the sense that it can be interfaced to a variety of host systems. These interfaces use different communication protocols, and require varied means of access to the PS 390 system.

Even as a host-independent system, communication between the host and the PS 390 is very important. This section discusses what different options are available for communicating with the host, and when that communication is important.

These communication options include:

- Standard host/PS 390 data transfer. In this method of communication ASCII files are downloaded to the PS 390 just as they would be sent to any physical device. In this scheme, the PS 390 system functions perform the parsing of data, and then "package" that data in a format that is acceptable by the PS 390 command interpreter. "Standard" communication implies that the host can transmit data that is immediately acceptable by the PS 390 Chopper/parser, including routing bytes to multiplex graphics commands and terminal emulator information (ASCII escape and control sequences).

- Communication using the GSRs. In this method of communication the GSRs perform all prepackaging on the host. PS 390 commands are invoked by Pascal, FORTRAN or C routines and are sent directly to the PS 390 command interpreter. The GSRs utilize the host CPU to prepackage data before downloading to the PS 390. This allows for substantially faster throughput, as the PS 390 command interpreter can immediately accept the data from the host and initiate the appropriate actions without having to parse the incoming commands. The GSRs also contain the routing bytes within the routines. This is the recommended communication scheme for finished user applications.

6.1 Using the Host System

The cable between the PS 390 and the host provides a way to utilize different components and facilities of each system. The host is used as a storage device by the PS 390 and, optionally, is used as a data processing device by the GSRs.

6.1.1 Using the Host as a Storage Device

The PS 390 has no local storage capability beyond what is loaded into memory during a graphics session. Once the system is powered down, or an initialization is performed, memory is cleared and the loaded data are lost. To retain information, the host system must be used as a storage device.

Initially, the best way to visualize this is to think of the PS 390 as a physical device attached to the host, like a line printer or a terminal. They both have local memory, but that memory is temporary. Therefore, the host system is used to store data that will be used repeatedly.

Conventionally, the storage is accomplished by building files, or loading files that will be used by the PS 390 into the host system. The host names the files and stores them in an appropriate manner. Files can be built on the host directly (using the PS 390 terminal emulator facility) and then downloaded to the PS 390, using the host system facilities. The content of these files depends on the method of communication that is going to be used. The first communication method that will be covered here is standard ASCII communication with the host, where the host recognizes the PS 390 to be a physical device attached to the host, and the PS 390 can be assigned a logical name.

6.1.2 Using the Host as a Processor

The GSRs use the host to package all data before it is sent to the PS 390. The GSRs provide an alternative method of invoking almost all the PS 390 standard commands as FORTRAN, Pascal or C routines. The prepackaging that is performed on the host gives a much faster throughput for any data being transferred.

The GSRs incorporate the appropriate routing bytes into each routine or, in the case with some of the utility routines, allow the user to select the routing byte in a parameter of the call.

6.2 PS 390 Local Communication

When entering the PS 390 command mode all commands sent after the "@@" prompt and terminated with the ";" command delimiter character are routed directly through the keyboard system function to the PS 390 parser, and then passed to the command interpreter after being processed. Local communication implies that nothing is sent down from the host, nor sent back to the host.

6.3 Direct Communication Using Host Facilities

Under DEC VAX/VMS and DEC PDP/RSX-11, files built on the host using the standard ASCII PS 390 command language can be routed through the command interpreter to the appropriate PS 390 system function network using standard host utilities such as the VMS WRITE, TYPE, and COPY commands and the RSX-11 PIP commands. These same methods are also available to other operating systems that have ways of building a file on the host, including the proper routing bytes, and then sending that file to the PS 390 using the logical or physical name given to the PS 390 by the host system.

6.3.1 Routing to the Appropriate Function

In building the host file, one of the first considerations is to determine what system network in the PS 390 the file should be sent to. For the ASCII commands or vector lists in a file to be sent to the PS 390 command interpreter to create displayable data structures or create function networks, the file must contain a field separator and a routing character as the first characters in a file.

The field separator character is the single <CTRL \> character (made by pressing the CTRL key and the \ (backslash) key simultaneously). The routing character is 0 (zero). The characters at the top of the file, then, would be:

```
↑\0
```

The file must terminate with an field separator character and another routing character that will send any following data to the terminal emulator, so that messages from the host will appear as text on the screen. The field separator character and the routing character for the terminal emulator are:

```
↑\>
```

A good illustration of how these bytes work is to send a file containing a small vector list and PS 390 commands from the host to the PS 390. With the ↑\> characters at the top and bottom of the file, the file will appear as text on the screen. By editing the file and replacing the top ↑\> characters with the ↑\0 characters, the file will be processed by the command interpreter and the graphics will be displayed on the screen.

6.4 Using Operating System Utilities for Downloading

The following example shows how DEC operating utilities can be used to download a file containing a vector list and/or other PS 390 commands to the PS 390.

This example uses the EDT editor on VAX/VMS systems or the EDT or KED editor on RSX-11M systems. Perform the following steps:

1. Log on to the host.
2. Go to the directory in which you wish to create and store your file.
3. Access the keypad editor. For VAX/VMS this is done as one of the following lines:

   ```
   $ EDT filename.extension
   $ EDIT/EDT filename.extension
   ```

4. The first characters in the file must be ↑\0 to route the file to the command interpreter of the PS 390.
5. Enter in the appropriate vector list and/or other PS 390 commands.
6. The final characters in the file must be ↑\> to route the next data received to the terminal emulator.

Under VMS, while in the editor, the file can be written to the physical or logical device name of the PS 390 terminal by using the appropriate system commands. The file must contain the routing characters described above to do this.

For example, where the PS 390 is **tta0**, do the following:

- Press function key F9 (emulating the "gold" key of the VT100 numeric keypad) and then the "7" key on the numeric keypad (emulating the "Command" key on a VT100).

- The editor will prompt with "Command:" at the bottom of the screen.

- To send a copy of the file you are editing to the PS 390 enter in:

```
WR tta0
```

- Or, the file can be "exited", leaving the editor utility and then copied to the physical or logical device name of the PS 390 by using the COPY command in VMS and the PIP command in RSX-11M.

For example where the filename is ROBOT.DAT and the physical device name of the PS 390 is tta0: for VMS and tt16: for RSX-11M, one of the following commands can be used:

```
$ COPY ROBOT.DAT tta0:

> PIP tt16:=ROBOT.DAT
```

The file can also be written directly to the PS 390 using the VMS TYPE command, if entered from the PS 390.

These commands will copy or write the file to the PS 390. The routing characters at the top of the file will send the data to the command interpreter of the PS 390, where the ASCII characters are processed into graphics data to be sent as a graphics display to the screen. If the routing characters are not compatible with the host system, or need to be changed, refer to Section *RM5 Host Communications*.

6.5 The Graphics Support Routines

The graphics support routines (GSRs) were developed to provide an alternative method of invoking the standard PS 390 command language to provide for faster through-put rates between the host system and the PS 390. The GSRs are distributed on magnetic tape as source code in FORTRAN, Pascal and C programming languages.

It is the responsibility of the user to load, compile, and link the routines with their application program. Installation instructions for the GSRs are provided in the Customer Installation and User Manuals for specific interfaces and operating system. Further documentation for the FORTRAN, Pascal, and C GSRs is provided in Section *RM4*.

The GSRs provide the following capabilities:

- Reduce local PS 390 processing time by bypassing the PS 390 parsing function. The GSRs allow the host to "prepackage" PS 390 graphics commands and send the packaged data directly to the PS 390 command interpreter.

- Allow PS 390 commands to be called by an application program. The PS 390 programmer is supplied with an alternative method of invoking the PS 390 command language. With the GSRs the programmer invokes the appropriate PS 390 command and defines the command parameters in FORTRAN, Pascal or C.
- Support most standard PS 390 commands. There is almost a one-to-one correlation between each PS 390 command and each GSR routine. Most of the commands that are not supported by the GSRs are commands that have to do with system configuration rather than graphics applications.

- Maintain PS 390 command language syntax and naming conventions. The GSRs maintain the naming conventions and command syntax established by the PS 390 command language as much as possible. Most commands have simply been prefixed with a 'P' and shortened to conform to FORTRAN, Pascal or C name limitations. Syntax follows command language syntax, when appropriate.

- Error-handling utilities. Each GSR routine has an error handling parameter that can be used by the programmer to call an error-handling subroutine or procedure.

- Support new products and options. The graphics support routines will continue to support new PS 390 commands as they are added to the PS 390 command list.

All software distributed by E&S are the latest released versions. As the GSRs would need to be relinked with user application programs at the time of a new or updated release, a strong attempt has been made to make the GSRs as "fixed" as possible. Modifications made to the low-level routines that set up the communication line between the host and the PS 390 that might be made should not affect the routines and/or parameters that are used by the programmer.

IS4. MAINTENANCE AND SERVICES

CONTENTS

Section IS4
Maintenance And Services

Complete hardware and software support of all E&S products is provided nationwide and in Europe. Field service operations and customer sites receive back-up support from the Software and Technical Support Groups based in Salt Lake City.

E&S provides the following services to provide high-response support:

- A customer service number
- Three standard maintenance service plans
- Negotiated special service contracts
- Maintenance support for OEM customers

The following sections discuss the support services available. For more information, contact the E&S Customer Engineering Department.

1. Customer Service Number

The Customer Service number is: (800) 582–4375.
(801) 582–9412 (Utah only)

This connects you to the dispatch office which will coordinate the support service you need. The customer service number is intended for logging problems; hardware or software. Office hours are 9:00 A.M. Eastern Time to 4:30 P.M. Pacific Time (7 A.M. to 5:30 P.M. Mountain Time). During non-business hours a call recorder will be operational. If you leave your name, area code, phone number, system serial number and a short description of the problem a call will be logged for you.

1.1 Hardware Support

Hardware support is provided based on your selected maintenance service. The dispatch office relays the problem and urgency to the appropriate E&S Customer Engineer. The Customer Engineer then schedules the time with the customer to visit the PS 390 site for any necessary repairs.

1.2 Software Support

Software support is provided to customers by the Customer Engineering Software Support Group at E&S corporate headquarters in Salt Lake City, Utah. Customer software problems will be logged into a problem tracking system which ensures that all software calls receive the fastest possible resolution.

Once you have logged your problem through dispatch, a Software Support Analyst will call back within one hour to get a detailed description of your problem. Within 48 hours, you will receive by telephone either a solution for your problem or a status report. There is no charge for this service, provided the following conditions are met:

- You have a Continuous Maintenance Service Contract, or selected the software support option of the Customer Participation plan.
- You are having a problem with an E&S supported product which fails to perform as specified.
- Your system is PS 390.
- Your system is being installed or is still under warranty.

Customers who do not hold a Continuous Maintenance Service Contract, and who request special help with software not covered under a Service Contract (and are not covered under installation or warranty) will be billed for the service at prevailing Customer Engineering hourly rates. If you will be billed, your consent will be obtained before the problem is investigated.

Before you call, please check the documentation to be sure your question cannot be answered there. Also, please have the following information at hand when you call:

- PS 390 graphics firmware version number. This is the E&S part number that appears on the distribution media of the PS 390 graphics firmware package (for example, 904015-001 A1).
- The host computer model, and operating system and its version number under which you are encountering difficulties (for example, DEC VAX 11/780, VMS V3.3).
- Complete error numbers and messages which were output.
- If possible, the commands issued that caused the problem in question.

1.3 Documentation Support

Additional support is provided to customers by the Interactive Systems Publications Group at corporate headquarters in Salt Lake City, Utah. During the hours of 8:30 a.m. to 4:30 p.m. Mountain Standard Time, technical writers are available for questions or comments regarding documentation. Your corrections and/or suggestions are encouraged regarding existing documentation or needed documentation.

This service is available free of charge to anyone using E&S equipment and its related documentation. No other requirements (such as a maintenance contract) are required. Documentation must be ordered through your E&S Marketing Representative.

2. Standard Maintenance Service Plans

E&S offers three standard maintenance plans. These are described in the following sections.

2.1 Continuous Maintenance Service

Continuous Maintenance Service is an on-call contract maintenance service. It is designed to supply you with complete and continuing maintenance of all hardware, software, and firmware supplied by E&S. It features 24-hour response for corrective maintenance, and periodic preventative maintenance. Standard service hours are eight hours per day, 8:00 a.m. to 5:00 p.m. five days per week, excluding holidays. Extended service hours are available under the Continuous Maintenance Service plan. The cost of all labor, parts, travel and living are covered by the monthly service fee for the Continuous Maintenance Service.

Software maintenance under the Continuous Maintenance Service plan covers identification and correction of errors in the software, and consultation on the programming of the graphics system. On-site assistance is supplied when necessary.

2.2 Requested Maintenance Service

Requested Maintenance Service is a time–and–material service supplied only upon customer request for E&S hardware, software and firmware. Response time is supplied on a basis of the availability of service personnel and resources.

Customers will be billed for all labor, replacement parts, travel, and per diem using the published Requested Maintenance Service rates in effect when the service request is received.

2.3 Customer Participation Service

Customer Participation Service is only offered to universities and other non-profit research organizations. E&S provides the appropriate diagnostic software and written procedures for isolating problems to the module, or assembly part level. The module can them be removed from the system and sent to E&S for repair or replacement. The turn-around time for repairs/replacements is 15 days. In addition, this service provides you with telephone consultation to aid in failure identification, and two on-site visits per year by an E&S Customer Engineer.

An optional software support service can be include with this plan.

3. Special Service Contracts

In addition to the standard maintenance services, E&S designs custom maintenance service plans to meet the needs of its customers. For example, a maintenance service can be designed for a Continuous Maintenance Service customer to guarantee a 30-minute average response time, 60-minute mean time to repair, and a 99.5% system availability over a 96-hour-per-week service period.

4. Maintenance Support for OEM Customers

Various service and support offerings are available for OEM customers. These include maintenance training, card and module repair, first level on-site support, maintenance plan design and an OEM Field Service Support Plan. In this latter plan, second and third level support is provided to the OEMs first-level field service personnel by the Technical Support and Software Support Groups of the E&S Customer Engineering Department.