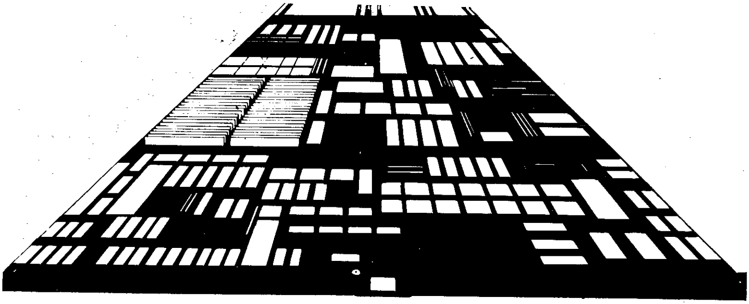
MODEL 3205

SYSTEM OVERVIEW



PERKIN-ELMER

PERKIN-ELMER

MODEL 3205 SYSTEM

Overview Manual

50-020 R00

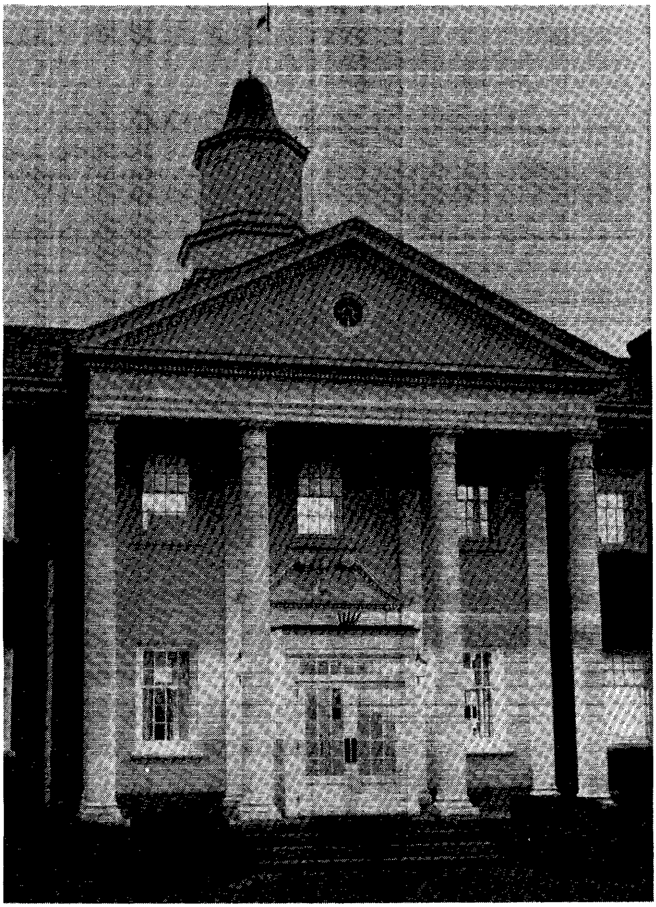
The information in this document is subject to change without notice and should not be construed as a commitment by the Perkin-Elmer Corporation. The Perkin-Elmer Corporation assumes no responsibility for any errors that may appear in this document.

The system described in this document is furnished under a license, and can be used or copied only in a manner permitted by that license. Any copy of the described software must include the Perkin-Elmer copyright notice. Title to and ownership of the described software and any copies thereof shall remain in the Perkin-Elmer Corporation.

The Perkin-Elmer Corporation assumes no responsibility for the use or reliability of its software on equipment that is not supplied by Perkin-Elmer.

The Perkin-Elmer Corporation
Data Systems Group
2 Crescent Place
Oceanport, New Jersey 07757

Printed in the United States of America



HISTORICAL PROFILE

A program of increasingly advanced products has culminated in the introduction of the first generally available two-board computer system based on a superminicomputer, offering performance advantages comparable, and in many ways superior, to larger systems - at a considerable cost savings.

The Perkin-Elmer Model 3205 System brings a new dimension in minicomputer capabilities to the successful Perkin-Elmer Series 3200 minicomputer family. The Model 3205 System is based on more than a decade of Perkin-Elmer's experience in high-performance 32-bit minicomputer system development and production. The architecture of the Model 3205 System is designed to fully exploit the field-proven Series 3200 software.

TABLE OF CONTENTS

CHAPTERS

CHAPTERS (Continued)

CHAPTERS (Continued)

CHAPTERS (Continued)

CHAPTERS (Continued)

CHAPTERS (Continued)

FIGURES

FIGURES (Continued)

TABLES

PREFACE

This manual is an overview of the Perkin-Elmer Model 3205 System. It is designed to function as an entry-level document for new users of the system and as an introductory document for prospective users of the system.

Chapter 1 of this manual describes a multiboard processor system. This chapter is the foundation for the subsequent discussion of the features and advantages of the Model 3205 two-board system.

Chapter 2 describes the two-board system developed by Perkin-Elmer using the current state-of-the-art technology.

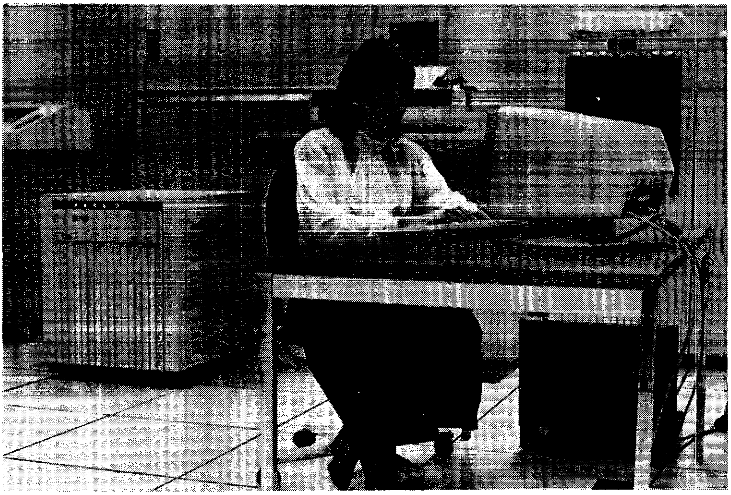
Chapter 3 describes the features, functions and benefits of the Model 3205 System.

Chapter 4 provides an overview of the application environments that the Model 3205 System is capable of supporting.

Chapter 5 is an overview of the extensive reference publications support provided with the Model 3205 System. The organization of this section will assist the reader in obtaining further information related to a specific item of interest regarding the Model 3205 System hardware and software.

Additional information regarding the documentation available to users of Perkin-Elmer 32-bit minicomputers can be found in the 32-Bit Systems User Documentation Summary, Publication Number 50-003.

This is a blank page inserted to retain pagination sequence.



CHAPTER OVERVIEW

Until now, minicomputers generally consisted of a multiboard central processing unit (CPU), system control boards and input/output (I/O) boards. This resulted in higher cost and greater size requirements. With the introduction of the Model 3205, a two-board system, these restrictions were removed.

The concept of a processor system contained on two printed circuit (PC) boards, consisting of a processor/memory board and a multiperipheral controller (MPC) board, is best explained when compared to a multiboard processor system. With this in mind, the following chapter provides a description of a multiboard processor system.

CHAPTER 1
MULTIBOARD PROCESSOR SYSTEM

1.1 INTRODUCTION

Generally, multiboard processor systems include a computer, system control modules, input/output (I/O) controllers and memory modules. This could require as many as 40 individual boards to respond to programmed instructions and determine the sequence in which the instructions are to be executed.

1.2 MULTIBOARD PROCESSOR SYSTEMS

In traditional minicomputer multiboard processor systems, each board or group of boards perform specific functions as determined by the configuration requirements. A processor system generally contains at least the following:

- a computer consisting of:
  - a central processing unit (CPU), the brain of the computer
  - a memory system, which contains the storage elements

  - an I/O system, which provides the processor system signals to the I/O controllers for user access to the system

- ancillary system control modules, which provide system control functions, such as system clocks and system loading program storage, etc.

- I/O controllers, which enable user devices to interface to the processor system

This multiboard design required various cabinet types and sizes, interconnecting cables and straps between cabinets, chassis and boards to complete the system. The following sections briefly describe the quantity and functions of the individual circuit boards.

Figure 1-1 represents these components graphically.

020-3

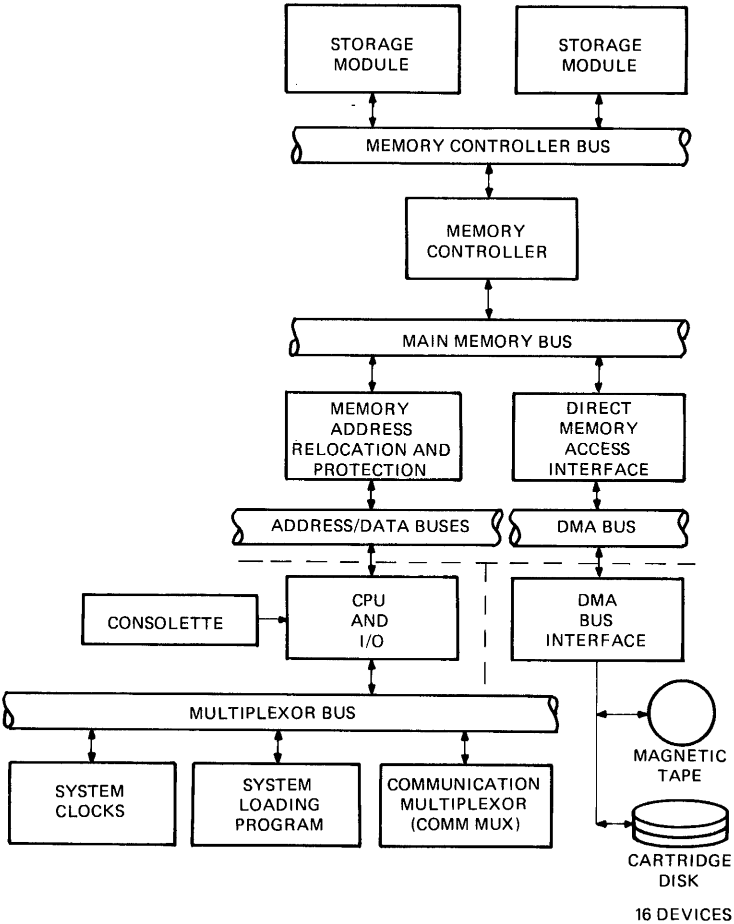


Figure 1-1 Multiboard Processor System

1.3 CENTRAL PROCESSING UNIT (CPU) BOARDS

The brain of most minicomputers is the CPU, usually contained on three or four PC boards. The CPU reads and interprets coded instructions and data stored in the computer's memory, and performs operations as required by a program. The CPU contains two sections: the control section and the arithmetic and logic section.

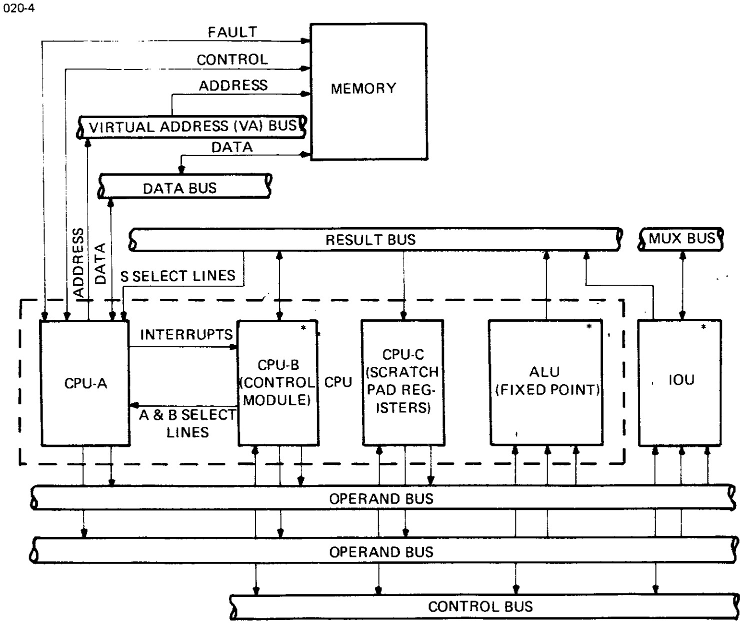
The hardware components for these two sections are divided among the multiple CPU boards by function and are located on one or two boards as required. This is represented in Figure 1-2.

1.3.1 Control Section

The control section of a CPU is the nerve center of the processor system and controls most of the components of the computer. It receives signals from various components of the computer, interprets the signals and acts upon them. This section provides sequencing and timing for the processing of instructions and commands. It also controls the data paths within and among the other components of the system.

1.3.2 Arithmetic and Logic Section

The arithmetic and logic section contains the arithmetic logic unit (ALU), processor registers, clock control and memory interface.



*MICROCODE ADDRESSABLE MODULES

Figure 1-2 CPU and I/O

1.3.2.1 Arithmetic Logic Unit (ALU)

The ALU is the unit in which arithmetic and logical operations are performed. Arithmetic operations consist of addition, division, subtraction, multiplication, rounding, square root operations and sign manipulation. Logic operations consist of examining and comparing operations. These functions are performed by devices known as logic circuits that operate according to certain preset rules.

1.3.2.2 Processor Clock Control and Memory Interface

Each processing system requires timing for operational control and synchronization. Due to the large number of components required for the control section timing, these components are located on additional CPU boards. The memory/CPU interface is part of the backpanel wiring, which is common to all PC boards.

1.4 MEMORY SYSTEM BOARDS

The memory is the part of the computer that stores data and instructions. All data and instructions are assigned an address that is used by the CPU when reading from or writing to memory. The memory system of a minicomputer contains the various memory access paths, internal buses, memory controllers and storage elements.

The system memory boards and interconnecting buses generally consist of the following:

- Memory address relocation and protection board
- Direct memory access (DMA) interface board
- Main memory bus
- Memory controller board
- Memory controller bus
- Storage module (STM) board

Figure 1-3 illustrates these components.

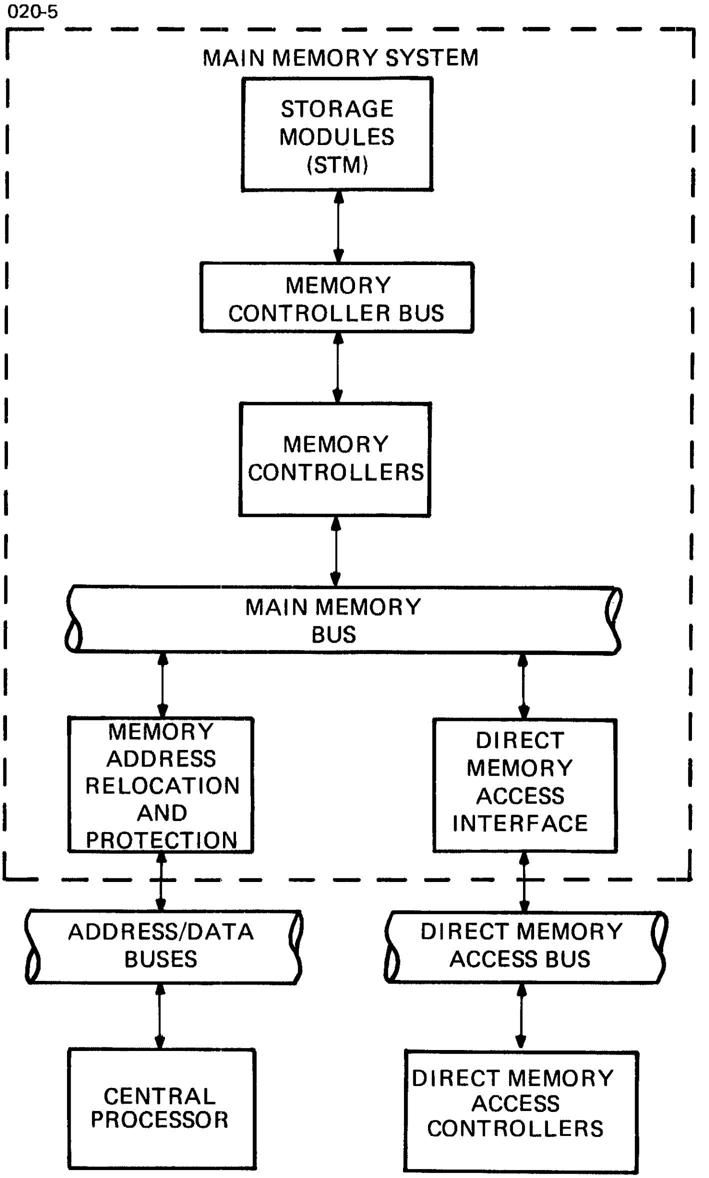


Figure 1-3 Memory System

1.4.1 Memory Address Relocation and Protection Board

A memory address relocation and protection board is used in many minicomputers and is populated on one or two PC boards as required.

1.4.1.1 Memory Address Relocation and Protection

Memory addresses in minicomputers and their physical locations in memory are identical. To increase the memory size in a processor system, a function referred to as virtual addressing is used. That is, the memory addresses are not referenced to physical address locations in memory and must be translated to additional physical address locations by a device such as a memory address relocation and protection board.

When a memory address is referenced to a physical address location, the memory address relocation and protection board is disabled and becomes transparent to the memory address. When the address to memory is a virtual address, the memory address relocation and protection board is enabled and translates the virtual address into a physical address location.

The memory address relocation and protection board usually decodes addressing faults and sends a fault code to the processor.

1.4.2 Direct Memory Access (DMA) Interface Board

In order to obtain faster response time and higher data transfer rates, many minicomputers use some form of DMA to facilitate the use of high-speed peripheral devices such as magnetic tape, cartridge disk, etc.

A DMA interface board provides high-speed access directly to memory without CPU intervention. It is connected to a main memory bus (common to all memory interface components) by a DMA bus with two qualifications:

- The DMA interface has higher priority on the main memory bus than the CPU interface.
- Transfers between a DMA and memory can be performed simultaneously with transfers between the CPU and memory.

1.4.3 Main Memory Bus

In most minicomputers, the main memory bus is usually part of the chassis backpanel wiring, which is common to all PC boards. This bus interfaces to memory all the components of the system that require memory access (CPU and DMA).

The CPU gains access to the bus through the memory address relocation and protection board, and the DMA devices gain access to the bus through the DMA interface.

1.4.4 Memory Controller Board

A memory system requires a device that controls the address/data transfer between the main memory bus and memory. In many minicomputers, a memory controller is located between the main memory bus that interfaces the CPU and DMA to the memory system, and the memory controller bus that interfaces to the memory storage modules (STMs). In MOS memory systems, the memory controller is the computer's error detecting and correcting device, capable of detecting and correcting all single-bit errors and of detecting all double-bit errors and some multiple-bit errors in each data word transmitted from memory.

1.4.5 Memory Controller Bus

In many minicomputers, the memory controller board is connected to the STM by a second internal bus, a memory controller bus. The memory controller bus is usually part of the chassis backpanel wiring, which is common to all PC boards.

1.4.6 Memory Storage Module (STM) Board

All minicomputers must contain storage elements to retain the information gathered by the computer. A typical memory storage module is a random access memory (RAM) using either 16 kilobit or 64 kilobit dynamic MOS RAM chips as memory storage elements.

The STM memory capacity varies but can be expanded from .25Mb to as much as 2Mb per board as required by the system design. A system's memory can be expanded from .25Mb (one STM board) to 32Mb by adding more STMs as directed by the system designer.

1.5 INPUT/OUTPUT (I/O) BOARD

All the processor signal, address and data lines are available on the I/O board.

The main function of the I/O board in a minicomputer is to provide a means for communicating with peripheral device controllers and interfaces.

The I/O board provides a versatile means for the exchange of information between the CPU and the external devices on a priority basis.

1.6 ANCILLARY SYSTEM CONTROL MODULE BOARDS

Due to board size and power constraints, which limit the chip count on a PC board, many minicomputer processor boards cannot accommodate the chips required for their system clocks, system loading program storage element, etc. Therefore, the chips required for these functions must be installed on ancillary system control boards.

1.6.1 System Clocks

System clocks are designed to provide time of day and programmable timer interrupts.

The line frequency clock (LFC) and the precision interval clock (PIC) are external clocking devices commonly used in many minicomputers.

Both clocks provide timer-controlled processor interrupts, but have different timing mechanisms. Usually, one clock is controlled by the system and one clock is program controlled by the user.

The LFC is derived directly from the AC power line and has a fixed clock rate. The user has no control over the LFC other than to disable or enable its function.

The PIC is derived from a crystal oscillator and is dynamically variable through program control. The user can select an interval of time and cause an interrupt in the hardware at the completion of the interval.

1.6.2 System Loading Program

The system loading program automatically initializes/restarts the computer system. The initial start of a processor system usually involves loading the operating system into memory before a user program can be loaded and executed. The system loading program is usually contained in a loader storage board.

1.7 SYSTEM INPUT/OUTPUT (I/O) CONTROLLER BOARDS

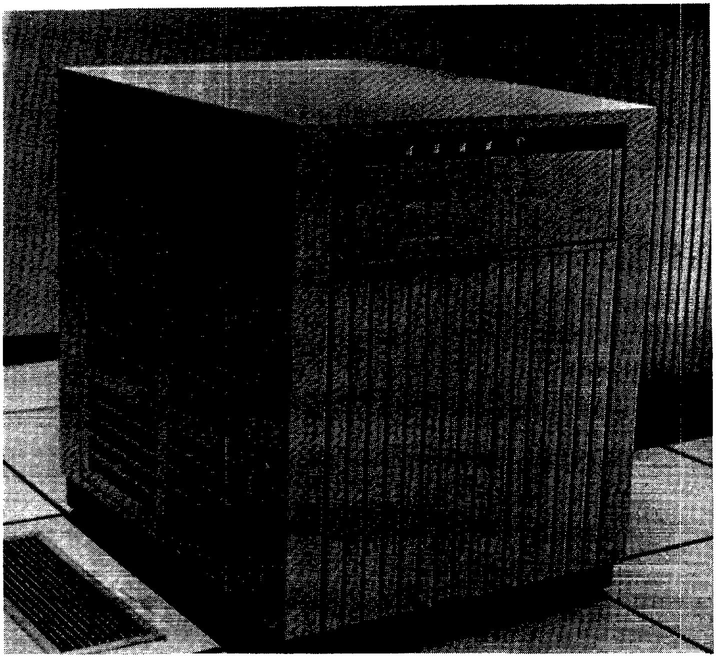
The minicomputer provides space for compatible I/O controller boards that enable the users to interface to a processor system.

1.7.1 Communication Multiplexor (COMM MUX)

Many minicomputers have a requirement to interface multiple terminals to the system. Due to the large number of chips required on the processor boards, this function is usually located on a separate board.

The COMM MUX provides an interface between a low-speed MUX bus and a variety of asynchronous data sets or local terminals in 2-wire half-duplex (HDX) or 4-wire full-duplex (FDX) mode.

This is a blank page inserted to retain pagination sequence.



CHAPTER OVERVIEW

Perkin-Elmer used state-of-the-art technology, including very large scale integrated circuits (VLSI) and large scale integrated circuits (LSI) to reduce the chip count and develop a two-board data processing system containing a single processor/memory board and a single system control and I/O controller board (multiperipheral controller). This reduction in the number of boards is made possible by the consolidation of multiple chip functions into one chip.

The following chapter provides a description of the two-board data processing system architecture of the Model 3205 System.

CHAPTER 2
THE MODEL 3205 TWO-BOARD PROCESSOR SYSTEM

2.1 INTRODUCTION

By using state-of-the-art technology that enables a single chip to perform the same functions as multiple chips, the Model 3205 System requirements are satisfied by two printed circuit (PC) boards: a processor/memory board, capable of performing all data processing and memory operations, and a multiperipheral controller (MPC) board, capable of performing all the system control and input/output (I/O) functions. Therefore, users of the Model 3205 System have all of the capabilities of the Perkin-Elmer Series 3200 line of processors at a lower cost and in a dramatically smaller package.

2.2 THE TWO-BOARD DATA PROCESSING SYSTEM

The Model 3205 System contains two PC boards that have all of the integrated circuits (ICs) and logic circuits required to comprise a complete system. They are:

- the processor/memory board, which performs all the arithmetic, computer control and memory operations, and

- the MPC board, which provides the clocks, operating system loading capability, hardware communication assist, data communication lines (synchronous and asynchronous) and a dedicated line printer port.

This chapter describes the Model 3205 two-board data processing system concept. Figure 2-1 illustrates this system.

The following sections identify the modules contained on the processor/memory board and the multiperipheral controller (MPC) board of the Model 3205 System.

2.3 THE PROCESSOR/MEMORY BOARD

The processor/memory board eliminates the interconnecting bus and front edge connector strapping required of multiboard systems. The following six sections are contained on this board and replace the processor and memory boards required in a multiboard system.

- Control
- Processor
- Instruction register
- Memory
- I/O interface
- Consolette interface

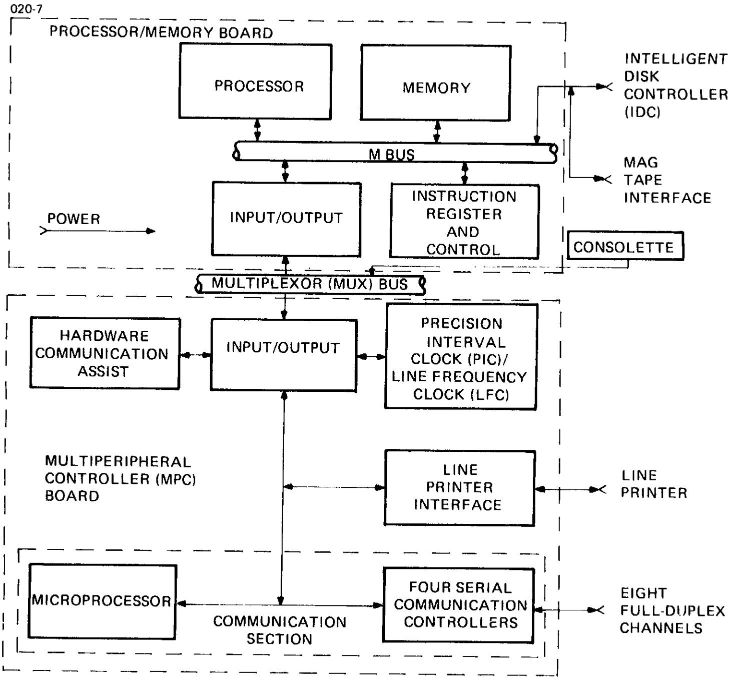

020-7
PROCESSOR/MEMORY BOARD
PROCESSOR
MEMORY
M BUS
INPUT/OUTPUT
INSTRUCTION REGISTER AND CONTROL
POWER
INTELLIGENT DISK CONTROLLER (IDC)
MAG TAPE INTERFACE
CONSOLETTE
MULTIPLEXOR (MUX) BUS
HARDWARE COMMUNICATION ASSIST
INPUT/OUTPUT
PRECISION INTERVAL CLOCK (PIC)/ LINE FREQUENCY CLOCK (LFC)
MULTIPERIPHERAL CONTROLLER (MPC) BOARD
LINE PRINTER INTERFACE
LINE PRINTER
MICROPROCESSOR
COMMUNICATION SECTION
FOUR SERIAL COMMUNICATION CONTROLLERS
EIGHT FULL-DUPLEX CHANNELS

Figure 2-1 Model 3205 System Boards

2.4 THE MULTIPERIPHERAL CONTROLLER (MPC) BOARD

The MPC board eliminates the multiboard requirements of backpanel interconnecting buses and front edge connector strapping required in multiboard systems. The following six sections are contained on this board and replace the system controllers and I/O controllers required in a multiboard system.

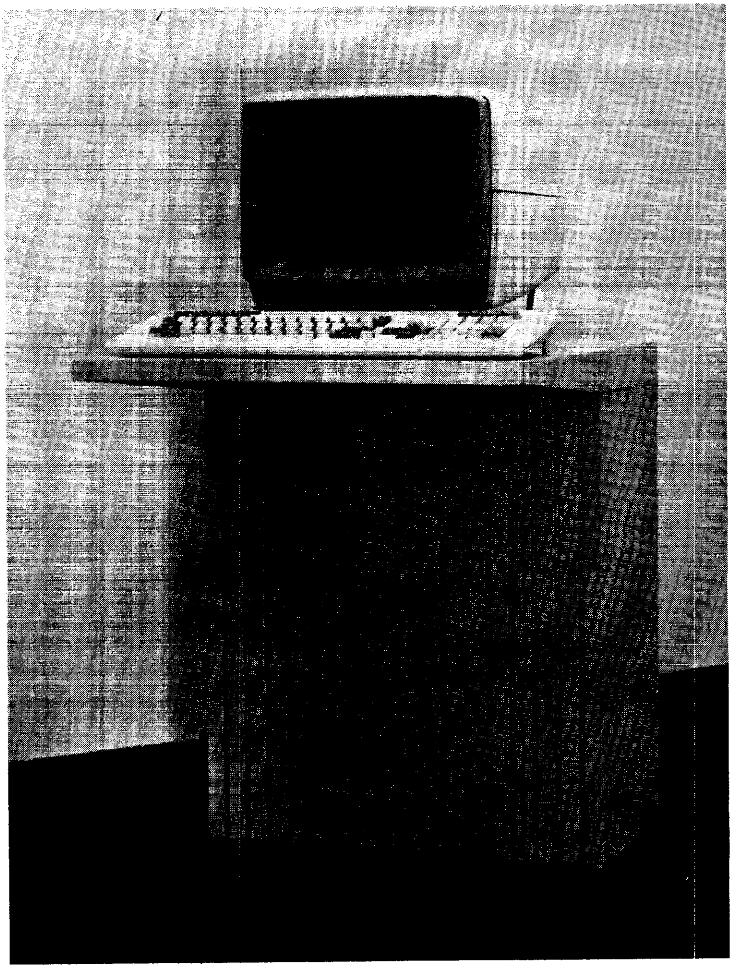
- Multiplexor (MUX) bus interface
- Communication section data communication lines (synchronous/asynchronous)
- Parallel line printer interface
- Precision interval clock/line frequency clock (PIC/LFC)
- Hardware communication assist
- Loader storage unit (LSU)

2.5 SUMMARY

The use of high-technology components and space-saving memory packaging enabled Perkin-Elmer to produce the Model 3205 32-bit minicomputer consisting of a processor/memory board and an MPC board.

This is a blank page inserted to retain pagination sequence.

020-8



CHAPTER OVERVIEW

This chapter deals specifically with the complete Model 3205 System and, along with the description in Chapter 2, describes the features, functions and benefits of this system.

CHAPTER 3
MODEL 3205 SYSTEM FEATURES, FUNCTIONS AND BENEFITS

3.1 INTRODUCTION

In the previous chapters, the evolution of a multiboard system to a single-board data processing and a single-board input/output (I/O) controller system was described. This chapter details the complete Model 3205 System and all of the features and capabilities of this system.

3.2 COMPONENTS OF THE SYSTEM

The Model 3205 System features a processor under the control of Perkin-Elmer's proprietary operating system, OS/32, or the Perkin-Elmer Edition VII Programmer's Workbench operating system. This software is described in Chapter 4. The I/O section is controlled by the processor providing a focal point for system control. Figure 3-1 shows the components of this system.

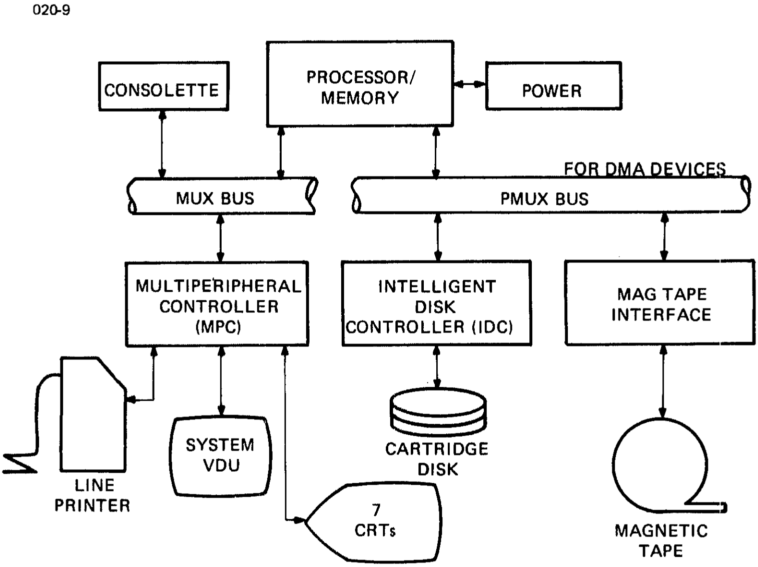


Figure 3-1 Model 3205 System

3.3 SYSTEM HARDWARE

The primary components of the Model 3205 System are:

- Processor/memory
- Multiperipheral controller (MPC)
- Intelligent disk controller (IDC)
- 50Mb cartridge disk drive (CDD50)

- Consolette

- Model 6100 video display unit (VDU)

3.4 PROCESSOR/MEMORY

The processor of the Model 3205 System is the controller of system resources. This control includes task scheduling, I/O management, handling fault conditions and executing application tasks.

The processor, together with the memory system, comprises a fully functional stand-alone computer capable of performing all of the duties of a typical processing system.

Figure 3-2 illustrates the internal structure of the Model 3205 processor/memory board. All the functions of a multiboard processing system have been retained, but reduced in size.

3.4.1 Control

The control section contains a microprogram control store that holds a sequence of commands. The processor is directed by commands fetched from the control store. The sequence of user instructions are controlled by the control unit, which determines where the next user instruction is coming from.

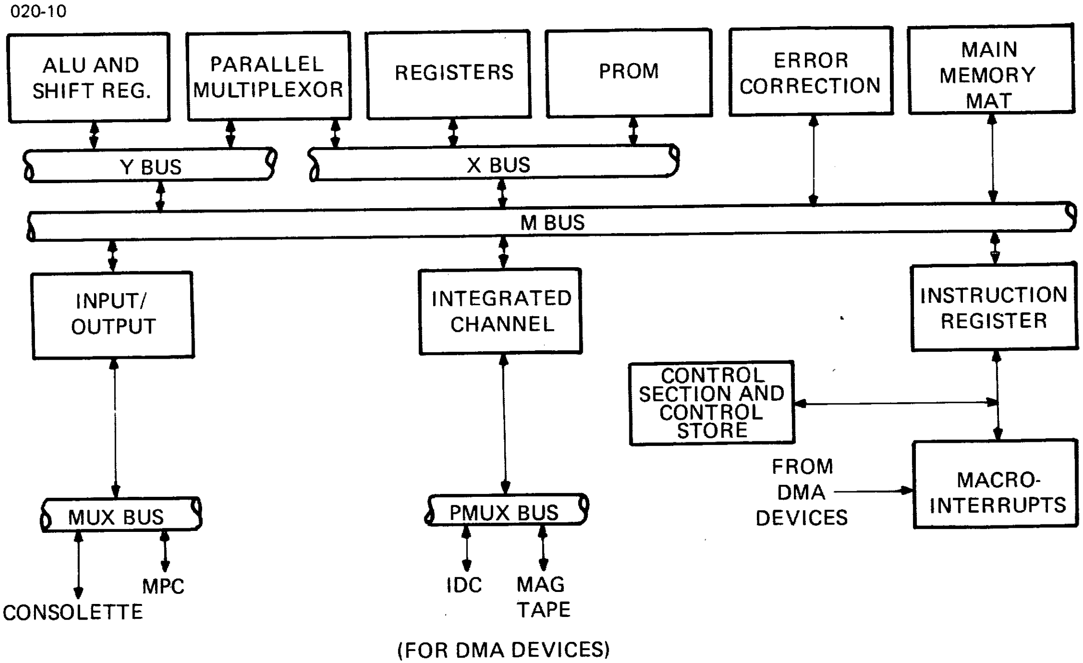


Figure 3-2 Processor/Memory Board

3.4.2 Processor

The processor contains the arithmetic logic unit (ALU), the direct memory access (DMA) counter, macrointerrupts and logic capabilities for the system. The following registers are also included:

- General-purpose registers
- Single and double precision floating point registers
- Microinstruction register
- Program status word (PSW) register
- Shift register
- Instruction register

3.4.3 Memory

The Model 3205 memory system has error correcting, logging, diagnostic capabilities and expansion and battery back-up options.

It supports up to 1Mb of memory on the single processor/memory board. The memory address registers access memory for read/write operations during multiplexor (MUX) bus or private multiplexor (PMUX) bus DMA transfers. The memory address translator (MAT) is enabled when a translation is required from a virtual address to a physical address.

NOTE

> The MAT functions are similar to the memory address relocation and protection circuit described in Chapter 1.

The memory section also contains an error detection and correction circuit and an error logger that records all errors for future analysis.

3.4.3.1 Error Detection and Correction

Error detection and correction is performed via a modified Hamming code consisting of 39 bits (32 data bits and seven detection bits). The additional seven detection bits of each word in memory allow detection and correction of all single-bit errors, detection of all double-bit errors and detection of some multiple-bit errors.

3.4.3.2 Error Logger

The error logger records data that identifies error trends. This data is collected and stored in a random access memory (RAM) and is used to isolate faulty memory chips before they affect memory system reliability. This information can be accessed and reports generated through an Error Reporting Utility, executed under the operating system.

3.4.4 Input/Output (I/O)

The processor/memory board contains two communication buses: the low-speed MUX bus and the high-speed PMUX bus that provides DMA.

3.4.4.1 Multiplexor (MUX) Bus

The MUX bus can address up to 1,023 medium-speed devices such as printers or VDUs. Data transfers over the MUX bus are accomplished in byte or halfword mode by the auto driver channel. The auto driver channel replaces conventional interrupt service routines with firmware sequences, thus providing a significant performance improvement in interrupt responsiveness and handling.

3.4.4.2 Private Multiplexor (PMUX) Bus

The PMUX bus is a DMA bus that can support up to five user device controllers. Once the PMUX has been activated, it functions in a completely autonomous fashion with one user device controller actively transferring data at any one time.

3.4.5 Consolette Interface

The consolette is interfaced to the Model 3205 System through the processor/memory board. The interface circuit generates three macrointerrupts and a single instruction mode signal to the microsequencer.

3.4.6 Power

The power fail circuit monitors both the P5 logic power and the P5U memory power and generates a power fail interrupt if a fault occurs.

3.5 THE MULTIPERIPHERAL CONTROLLER (MPC)

The MPC uses state-of-the-art microprocessor technology to implement the most commonly used I/O functions on a single printed circuit (PC) board. The communications section of the MPC is based on a 12.5MHz MPC microprocessor. The communications integrated circuits (ICs) are serial communications controllers. The remainder of the MPC contains field programmable logic arrays (FPLAs), programmable logic arrays (PLAs) and medium scale integration (MSI). These functions are used to map Series 3200 commands to the serial communications controller control sequences for compatibility. An MPC can serve up to eight device controllers; the Model 3205 System can be configured with one MPC and one 8-line COMM MUX to service 16 device controllers. Figure 3-3 shows the MPC board layout.

3.5.1 Multiplexor (MUX) Bus Interface

The MUX bus interface provides all the command and control request/response signals and address and data lines required to provide communication between the circuits on the MPC board and the processor.

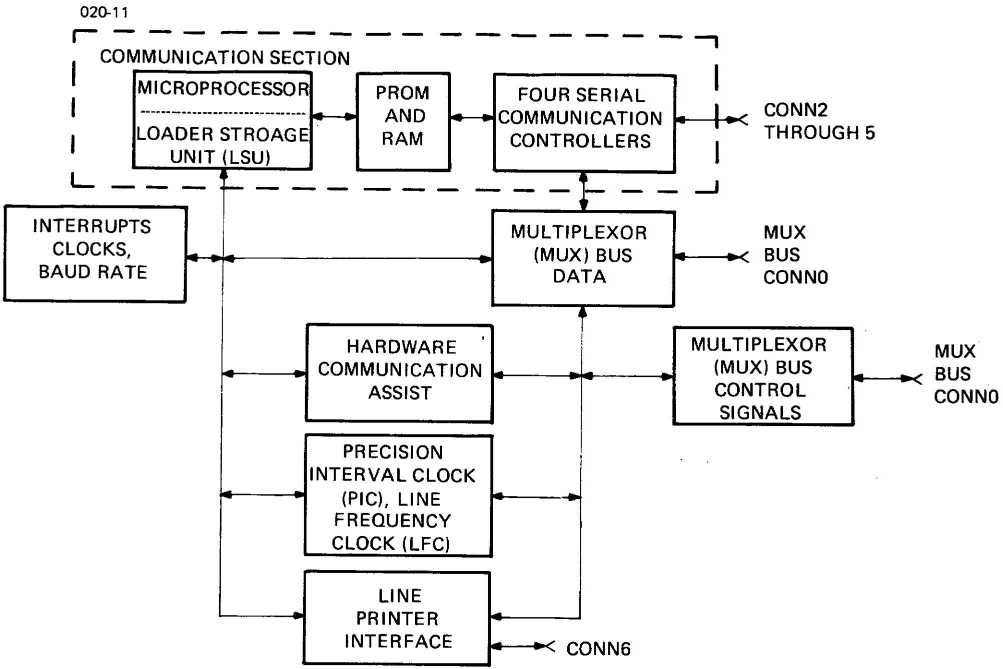

020-11
COMMUNICATION SECTION
MICROPROCESSOR
LOADER STROAGE UNIT (LSU)
PROM AND RAM
FOUR SERIAL COMMUNICATION CONTROLLERS
CONN2 THROUGH 5
INTERRUPTS CLOCKS, BAUD RATE
MULTIPLEXOR (MUX) BUS DATA
MUX BUS CONN0
HARDWARE COMMUNICATION ASSIST
MULTIPLEXOR (MUX) BUS CONTROL SIGNALS
MUX BUS CONN0
PRECISION INTERVAL CLOCK (PIC), LINE FREQUENCY CLOCK (LFC)
LINE PRINTER INTERFACE
CONN6

Figure 3-3 MPC Board

3.5.2 Communication Section

The communication section contains the microprocessor, the programmable read-only memory (PROM) and RAM, the loader storage unit (LSU), and the eight-channel data communications MUX.

3.5.2.1 Microprocessor

The microprocessor controls the request/response signals and the address and data lines between the MUX bus and the eight-channel data communications MUX.

3.5.2.2 Programmable Read-Only Memory (PROM) and Random Access Memory (RAM)

The microprocessor has access to 16kb of 200ns PROM and 8kb of 70ns static RAM.

3.5.2.3 Loader Storage Unit (LSU)

The LSU is located in the microprocessor control store. On initialization, the LSU intercepts LSU references by the processor microcode and reads the microprocessor control store. If the system is configured with two MPC boards, only one LSU is enabled.

3.5.2.4 Eight-Channel Data Communications Multiplexor (MUX)

The eight-channel data communications MUX consists of four serial communication controllers.

Each controller contains two independent, full-duplex channels and provides the following features:

- Up to 1Mb per second data transfer rate
- Crystal oscillator
- Baud rate generator
- Digital phase locked loop (PLL) for clock recovery
- Nonreturn to zero (NRZ), nonreturn to zero inverted (NRZI) and frequency modulation (FM) encoding, used on the MPC board to establish the eight-line protocol
- Local loopback and auto-echo

Any one of the eight channels can be asynchronous, synchronous data link control (SDLC), bisynchronous (BISYNC), or synchronous. Any mixture of protocols is allowed to exist on the MPC board.

3.5.3 Parallel Line Printer Interface

The line printer logic interfaces the line printer to the MUX bus. The interface communicates with the printer over eight data lines, one strobe line, one acknowledgement line, one busy line and three status lines (paper empty, fault and selected). The printer accepts the standard 7-bit ASCII code with the capability of converting lower-case characters to upper-case if required.

3.5.4 Precision Interval Clock/Line Frequency Clock (PIC/LFC)

The MPC contains two independent clock devices; both clocks provide timer controlled processor interrupts, but have different timing mechanisms.

3.5.4.1 Precision Interval Clock (PIC)

The PIC produces or queues a processor interrupt. A specified time interval determines the point at which the interrupt occurs.

Interrupt intervals are requested by the user and managed by the operating systems.

3.5.4.2 AC Line Frequency Clock (LFC)

The LFC generates or queues a processor interrupt. The point at which the interrupt occurs is determined by a fixed clock rate that is derived from the frequency of the AC power line. The user has no control over the LFC other than to enable or disable time-keeping.

3.5.5 Hardware Communication Assist

The hardware communication assist upgrades the auto driver channel of the processor and adds three instructions to the instruction repertoire. The auto driver channel time for the cyclic redundancy check (CRC) BISYNC is greatly reduced, and the error checking capability is increased to include the CRC

SDLC format. The three additional instructions are capable of performing error checking of characters one byte at a time or as a string of data bytes. An error check can be calculated in any one of the three formats: CRC-SDLC, CRC-BISYNC (also called CRC-16), or longitudinal redundancy check (LRC).

3.6 THE INTELLIGENT DISK CONTROLLER (IDC)

The IDC provides the processor with the capability of writing to and reading from disk media. The disk system consists of the following hardware components:

- one or more disks,
- a controller (IDC) for interfacing the disks to the processor, and
- the necessary cables for interconnecting the IDC hardware and the disks.

Figure 3-4 shows the IDC divided into three logic sections:

- PMUX bus interface
- Disk interface
- Microprocessor controller logic

3.6.1 Private Multiplexor (PMUX) Bus Interface

The PMUX bus section of the IDC board interfaces the PMUX bus of the processor with the IDC logic. The interface functions with the microprocessor controller logic in generating the commands, data, and control sequences necessary for disk I/O. I/O through the PMUX bus is either programmed or DMA.

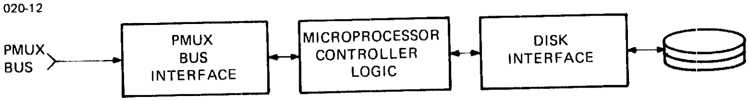


Figure 3-4 Intelligent Disk Controller (IDC)

3.6.2 Microprocessor Controller

The microprocessor controller section of the IDC board interprets and acts on the processor commands (e.g., seek, read, write and format) causing the disk drive to perform specific operations or data transfers. It also controls the IDC interface circuitry between the PMUX bus and the disk.

3.6.3 Disk Interface

The disk interface section of the IDC board interfaces the disk to the microprocessor controller logic to generate the necessary control sequences, data, and error detection and correction.

See Figure 3-5 for daisy-chain interconnection of four disks.

3.7 50MB CARTRIDGE DISK DRIVE (CDD50)

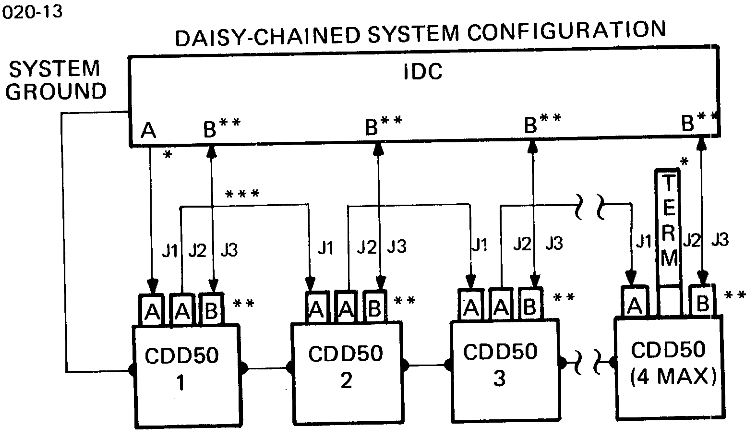
The CDD50 is a small, economical, medium-performance disk that features random access and a rotating disk mass memory device, and both removable and fixed storage. The CDD50 uses the latest eight-inch rigid disk technology using low mass flying read/write heads attached to a precisely controlled linear head positioner. The unformatted storage capacity of the CDD50 is 50Mb. 25Mb of storage is provided by the removable disk (cartridge) and 25Mb of storage is provided by the nonremovable disk.

3.8 CONSOLETTE

The consolette provides the user with information on the system's status and control of the following system functions.

- System primary power
- System initialization sequence
- Selection of free-running or single-instruction mode
- System current operational status
- Information from the consolette LEDs about the current state of the system

Figure 3-6 illustrates the consolette.



NOTES:

* TERMINATION OF "A" CABLE LINES ARE REQUIRED AT CONTROLLER AND THE LAST UNIT OF THE DAISY-CHAIN OR EACH UNIT IN A RADICAL CONFIGURATION.

** TERMINATION OF "B" CABLE RECEIVER LINES ARE REQUIRED AT THE CONTROLLER AND ON THE UNIT'S RECEIVER CARDS.

*** MAXIMUM CUMULATIVE "A" CABLE LENGTH PER CONTROLLER = 100 FEET (30.48 METERS). MAXIMUM INDIVIDUAL "B" CABLE LENGTH = 50 FEET (15.24 METERS).

Figure 3-5 CDD50 Daisy-Chain Cabling

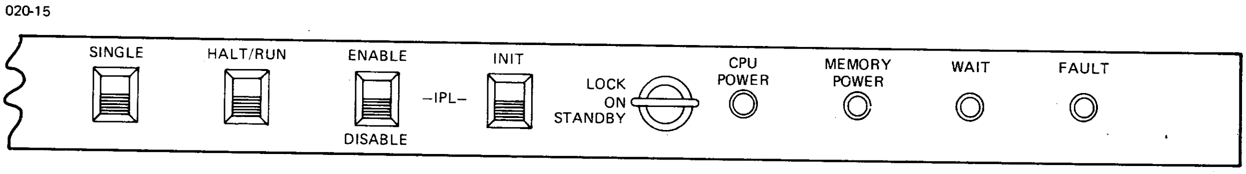


Figure 3-6 Consolette

3.9 MODEL 6100 VIDEO DISPLAY UNIT (VDU)

The Model 6100 VDU is a two-unit, desk-top terminal. It is a self-contained unit consisting of a display, detached keyboard, power supply, printer port and associated electronics. The Model 6100 VDU can be connected to a remote computer via telephone data lines.

The Model 6100 VDU has a full 128 ASCII character set, four programmable function keys, an adjustable low profile keyboard, 34 stepped and sculptured keycaps, tilt and swivel monitor, absolute cursor positioning, RS-232 interface, printer port and 24 lines with 30 characters per line. There is also a 25th line, which is used for terminal status, setup mode or host messages.

All displayable characters are written into memory and displayed on the screen. The cursor moves one position at a time to the right and remains at column 80 unless the new line enable feature is active. If the new line enable feature is active, the cursor moves to column 1 of the next line.

The Model 6100 VDU is shown in Figure 3-7, and the keyboard is shown in Figure 3-8.

3.9.1 Options

The Model 6100 VDU can be equipped with a 20mA current loop interface (CLI), RS-422 interface, large scale integration (LSI) graphics board and a modem board.

020-14

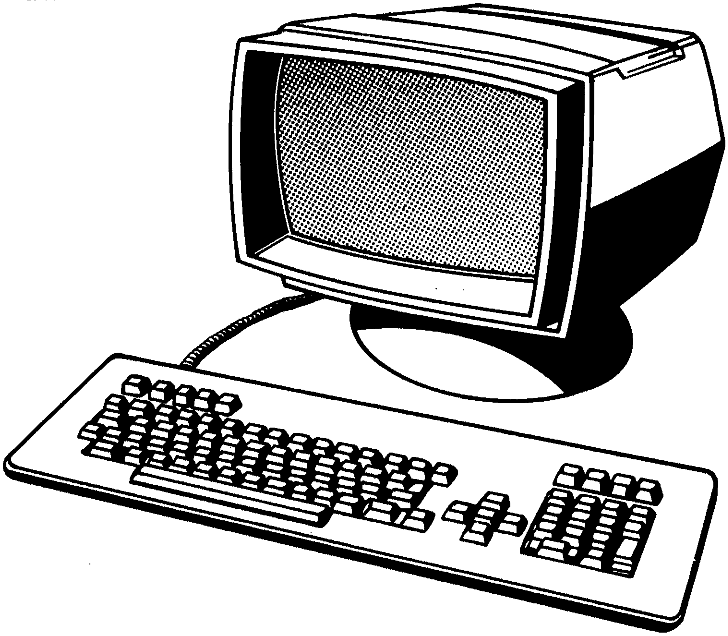


Figure 3-7 Model 6100 VDU

020-16

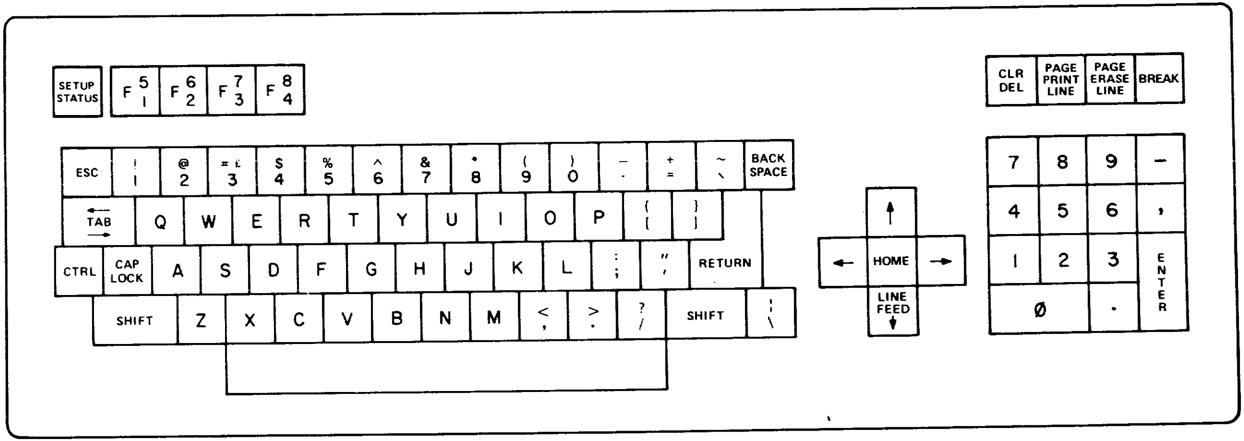


Figure 3-8 Model 6100 VDU Keyboard

3.10 SYSTEM SOFTWARE

All of the advanced features of the Model 3205 System outlined in the previous sections are fully exploited by the system software in order to maximize the benefits of the system. There are three areas of software support designed for users of the Model 3205 System. They are:

- Operating system support (via OS/32 or Edition VII Workbench)
- Application level support (via the range of language support provided by the chosen operating system)
- Program development support (via the Ease of Use (EOU) feature under OS/32 and the Source Code Control System (SCCS) under Edition VII Workbench. Program development is also aided by the symbolic debuggers available with either operating system.

3.10.1 Operating System Support Under OS/32

OS/32 is a versatile, multienvironment operating system. It provides a responsive environment for a broad spectrum of applications and comprehensive support for all phases of program and system development. A central system console provides system control, or control can be distributed to user-written monitor tasks so that subsystems can be executed under independent, local control.

OS/32 supports concurrent multiprogramming, with user tasks written in any of the supported languages:

- FORTRAN VII
- Pascal
- C
- COBOL
- RPG II
- BASIC
- CORAL 66
- Assembler

The scope and power of OS/32 can be extended through the use of its companion products: Multi-Terminal Monitor (MTM) and Reliance PLUS.

MTM provides a general-purpose programming environment for up to 64 interactive terminal users. The full program development features of OS/32 are available to each interactive terminal user. In addition, the capacity and throughput of the system are maximized by the provision of background batch processing and I/O spooling.

Reliance PLUS is Perkin-Elmer's software package for the implementation and execution of high-performance, on-line data base management applications for up to 240 interactive terminal users. It supports transaction processing, data base management, data definition, query/report processing and application development. Reliance PLUS is suited to many different types of applications, including manufacturing, personnel management and financial services.

Both of these environments can coexist on the same Model 3205 System and be made available to interactive users through the Environmental Control Monitor (ECM).

3.10.2 Operating System Support Under Edition VII Workbench

Perkin-Elmer's Edition VII Workbench operating system is a general-purpose, time-sharing environment capable of supporting up to 128 interactive users. Edition VII Workbench provides a highly structured operating system with a proven record of improving the productivity of systems and applications programmers.

Edition VII Workbench is specifically designed for simplicity of operation, user productivity and versatility. Applications range from business and scientific computing to telecommunications networking. The available languages include:

- C
- FORTRAN VII
- FORTRAN 77

- RATFOR
- Pascal

3.11 SYSTEM INTEGRITY

The hardware and software capabilities of a system are always of paramount importance. However, system integrity is equally important to you, the user, since the ultimate test of the system is its ability to remain operational and reliable. The Model 3205 System reflects many years of development in this area.

3.11.1 Hardware Integrity

Hardware support is found in three areas.

- Power subsystem
- Processor
- Memory system

3.11.1.1 Power Subsystem

The power subsystem is modular in design, self-testing and constantly monitors itself and all equipment powered by it to make certain that voltage is constantly maintained within tolerance. Any sign of failure in the power subsystem causes it to alert the processor, allowing the processor to enter the power fail mode.

3.11.1.2 Processor

The processor is designed for integrity. On power up, the processor self-tests the internal data buses and registers. The processor permits the remainder of the power up sequence to complete only if these tests are satisfactorily completed.

During power fail, the task context is saved and the processor waits for recovery. After a complete system power fail, the processor automatically returns control to the operating system only after performing a nondestructive memory test of the operating system area. This ensures that the operating system is intact and capable of controlling the system.

3.11.1.3 Memory System

Memory system integrity is important to overall system integrity. The Model 3205 memory system incorporates error detection and correction so that all single-bit errors are corrected and double-bit errors are detected.

When an error is corrected, it is logged in the error logger so that at maintenance time, these integrated circuit (IC) chips can be replaced before double-bit errors become prevalent. Also, the errors are reported through OS/32 programs run at maintenance time to identify failing memory chips. Errors in the I/O section can be monitored to the extent that the peripheral hardware supports.

3.11.2 Software Integrity

Integrity is also an important consideration for system software. All software on the Model 3205 System has been designed with integrity in mind.

Data structures form the cornerstone of software integrity; OS/32 data structures are double linked (forward and backward pointers). The validity of any item in a list can be checked against the structures preceding it and following it before the item is used by OS/32 software. Hardware alignment checks are used by placing all data structures on fullword or quadword boundaries so that corrupt pointers to tables can be detected.

Further integrity is provided via memory tests, automatic retry of I/O, parity checks and system fault handling. Fault handling includes power failure, memory failure and I/O device failures. The actions taken in handling these failures (detailed in the following sections) reflect the fail-soft capability of the system.

3.11.2.1 Fail-Soft Capabilities

When a fault is detected, the processor attempts to take the affected component off-line. In the case of a memory failure, the affected area of memory is marked off and access is restricted to read-only. The operator is informed so that the appropriate corrective action can be taken. In line with Perkin-Elmer's concern for integrity, tasks have an extensive capability to respond appropriately to self-generated faults.

3.11.2.2 Application Related Failures

When an application task encounters a fault, such as a memory access fault or an illegal instruction, that task is normally paused. The processor sends an appropriate message to the console.

3.12 SYSTEM CONFIGURATION AND EXPANSION

The Model 3205 System features flexibility in system expansion, an important area in system design.

The Model 3205 System configuration provides all the capabilities for simple field expansion for a basic system containing .5Mb of MOS memory and one disk drive to a system containing 4Mb of MOS memory, four disk drives, battery back-up, and an additional I/O chassis.

3.12.1 Memory Expansion

The memory size in the basic Model 3205 System is 512kb, expandable to 1Mb on the processor/memory board. Memory can be expanded in increments of 1Mb to a maximum of 4Mb using a single additional memory board.

3.12.2 Battery Back-Up

Since MOS memory is volatile, the Model 3205 System provides a battery power retention system. It provides up to 36 minutes of power depending on the configuration. That is: 36 minutes for .5Mb of memory to 10 minutes for 4Mb of memory. When line power is lost, the battery system automatically provides the power to maintain memory contents. Upon restoration of power, the processor reloads its registers and resumes operation.

3.12.3 Input/Output (I/O) Expansion Chassis (208/230V Systems Only)

An I/O expansion chassis can be installed in the Model 3205 System when operating in the 208/230V power mode. This provides the user with eight additional slots for I/O controllers.

3.12.4 Additional Disk Drives

Up to three additional CDD50s can be connected to the system. The second disk drive can be installed directly in the system cabinet. The third and fourth disk drives must be installed in an expansion cabinet.

3.13 MODEL 3205 VERSION FOR ORIGINAL EQUIPMENT MANUFACTURERS (OEMs)

A basic variation of the Model 3205 System is available to OEMs that want to embed a minicomputer into their own system or design their own special packaging. This basic version includes a processor/memory board, multiperipheral board, chassis and a consolette. Additional options and expansions are available as required.

3.14 DIAGNOSTIC SUPPORT

In keeping with Perkin-Elmer's philosophy of integrated system support, an extensive spectrum of diagnostics is available for use with the Model 3205 System. These diagnostics are designed to test individual components of the system as well as the operation of interrelated subsystems. These diagnostics are available for stand-alone testing (without an operating system present).

Customer service engineers and on-site maintenance personnel will find these diagnostic programs invaluable for verifying the integrity of system components and in system troubleshooting situations.

Table 3-1 is a summary of the various diagnostics available.

TABLE 3-1 MODEL 3205 SYSTEM DIAGNOSTICS

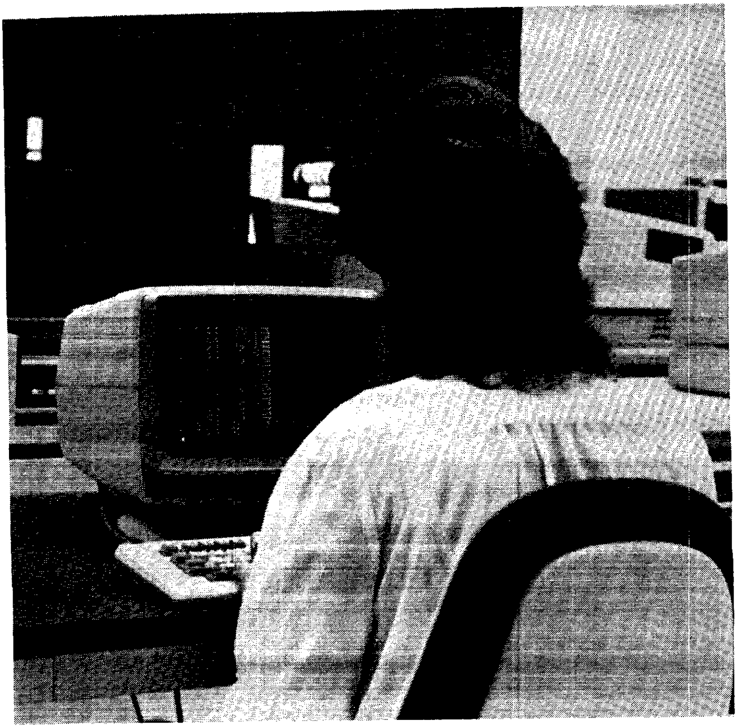
PERKIN-ELMER PART NUMBER	DIAGNOSTIC TITLE
50-025	Diagnostic Media User's Guide
06-145	Diagnostic Executive
06-228	Series 3200 Processor Test Part 1

TABLE 3-1 MODEL 3205 SYSTEM DIAGNOSTICS (Continued)

PERKIN-ELMER PART NUMBER	DIAGNOSTIC TITLE
06-229	Series 3200 Processor Test Part 2
06-231	Series 3200 Floating Point Test
06-238	Series 3200 Commerical Instruction Test
06-280	Series 3200 MAT Test
06-289	Model 3205 System MOS Memory Test
06-159	System Exercisor
06-161	32-Bit Selector Channel (SELCH) Test
06-267	IDC Test
06-268	IDC Formattor Test
06-288	Model 6100 VDU Test
06-291	MPC Test

This is a blank page inserted to retain pagination sequence.

020-17



CHAPTER OVERVIEW

The feature that is of primary importance to most system users is the manner in which the system supports the application environment. The benefits derived from any system are directly related to those system features designed to support the development, testing and execution phases of applications. The Model 3205 System provides a complete and comprehensive complement of integrated applications features, as described in Chapter 4.

CHAPTER 4
MODEL 3205 SYSTEM APPLICATION ENVIRONMENTS

4.1 INTRODUCTION

This chapter provides an overview of the application environments that the Model 3205 System is capable of supporting. Model 3205 System application environments can be broken down into those supported under Perkin-Elmer's proprietary operating system, OS/32, and those that are supported under Perkin-Elmer's Edition VII Workbench.

4.2 APPLICATION ENVIRONMENTS UNDER OS/32

OS/32 is a mature, field-proven operating system that provides flexible system control and operation. It offers comprehensive operating system features and is adaptable to a broad spectrum of user applications. OS/32 assists programmers in all phases of system and applications development, from program design and preparation through implementation and operational support for completed application systems.

Basic telecommunications, for point-to-point and multidrop configurations, are provided by the Integrated Telecommunications Access Method (ITAM) package supplied with OS/32.

ITAM also provides support for higher level Perkin-Elmer telecommunications products such as 2780/3780 RJE, HASP/32 and the zero-bit insertion/deletion (ZBID) data link control (ZDLC) Channel Terminal Manager. These products support popular line protocols such as ASYNCH, BISYNCH, HDLC, SDLC and ADCCP.

Systems networking can be achieved over both short and long distances. Local Area Networking (LAN) is provided through Perkin-Elmer's implementation of the popular Ethernet, while long distance networking is available through PENnet.

In the following sections, features of OS/32 itself, its companion products, and telecommunications and networking capabilities are described.

4.2.1 Features of OS/32

OS/32 provides an extensive range of features to aid in the development, implementation, and maintenance of user applications and the operating system. These features include:

- Management of the system environment
- Task management
- Intertask communications
- Reentrant and nonreentrant code
- Memory management
- Simple man-machine interface

4.2.1.1 Management of the System Environment

Management of the system environment is accomplished through an Executive, an Input/Output (I/O) Handler, a File Manager, a command language and Support Utilities.

The Executive manages the system and coordinates I/O requests for local devices or telecommunications facilities. It also handles task scheduling, allocation of memory, and interrupt and fault conditions. Up to 255 user tasks can execute concurrently in memory at priorities defined by the user. If necessary, tasks can be rolled to disk to free memory for higher priority tasks.

The I/O Handler offers device-independent I/O for a wide variety of devices. Peripheral devices or data communications facilities can be accessed in a file-oriented, device-independent manner. Through the facilities of ITAM, user tasks (u-tasks) can treat remote devices as though they were local devices; the details of protocols and line handling are left to the system software.

For applications that require it, full I/O queuing is provided. Access requests are scheduled according to one of several algorithms:

- first-in/first-out (FIFO),
- task prioriy, or
- shortest arm movement (for direct access devices).

Custom scheduling algorithms can be easily implemented by the user.

The File Manager coordinates the use of disk space. Several file types are available with both random and sequential access methods support for each. Files can be protected against unauthorized access through the assignment of access privileges and/or read-write protection keys.

The command language provides foreground system control over applications tasks as well as control over background processing. The command substitution system (CSS) is an extension of the command language and provides increased flexibility and decision making capabilities. Through the CSS, complex, yet flexible, sequences of commands can be stored as files and subsequently executed by issuing a single user-chosen command. In this manner, even the most complex operations can be reduced to the simplicity of entering a single command.

The OS/32 Support Utilities provide a wide range of management and maintenance facilities, including (but not limited to):

- screen and line-based editing,
- source code maintenance,
- interactive debugging to aid program development,
- task patching at the highest level,

- data protection, file archiving and restoration,

- I/O spooling,

- disk checking,

- accounting and reporting of processor and disk space usage, elapsed time, I/O transfers, and

- monitoring and reporting of processor performance and memory errors.

4.2.1.2 Task Management

The Task Management Facilities provide all the functions required to create, schedule, coordinate and supervise user applications systems. Task scheduling can be handled on a user-defined priority basis, with up to 255 separate task priorities, with or without time-slice scheduling, at the user's option.

Through time-slicing, executing tasks can be interrupted when they have used a predefined amount of central processing unit (CPU) time, thereby preventing any one task from monopolizing system resources. The time-slice is a dynamic value, and can be altered by the system operator. Priority can also be used to resolve I/O conflicts.

OS/32 recognizes three types of user-written tasks: e-task, d-task and u-tasks. An e-task (executive task) can use privileged instructions (i.e., perform program status word (PSW) manipulations) and executes with the memory address translator (MAT) disabled.

Because the MAT is disabled, e-tasks must use absolute addressing and can, therefore, access any portion of memory. A diagnostic task (d-task) can also use privileged instructions, but operates with the MAT enabled and, therefore, uses relative addressing. A u-task cannot use privileged instructions and operates with MAT enabled.

4.2.1.3 Intertask Communication

OS/32 provides facilities to allow one task to communicate with and/or control another task. The task that initiates the communication or control is referred to as the calling task; the task receiving the communication or being controlled is referred to as the directed task. The directed task can be any task in the environment, including the calling task itself. Through the intertask communications features, a task can load, start, stop, delete, exchange messages with, exchange logical unit (lu) assignments with, and manage interrupts from another task in the environment.

OS/32 provides a subtasking facility for developing subsystems within the total system of user-written tasks. Complex subsystems of cooperating tasks can coexist with one another and with independent tasks. Each subsystem has one task defined as its monitor and the other related tasks execute as members of the subsystem. The subsystem capabilities allow control of an application to be conveniently distributed to the point of maximum effectiveness. The user-written monitor is responsible for the activities of the subsystem.

Any event requiring intelligent intervention (for example, a fault or end of task) is reported to the monitor rather than to the system operator. If the condition requires human intervention, the monitor simply prompts the system operator or any other terminal user.

Another feature that is particularly valuable for intertask communication is the powerful task-level fault-handling capability of OS/32. This feature permits a task to be interrupted in its normal execution for any of a variety of hardware and/or software conditions. A trap can be caused by a number of events and task level handling of them can be enabled according to the user's needs. A task can suspend itself, waiting for the occurrence of a trap-causing event, thereby eliminating the need to continually scan for events. The standard trap procedure supports the full range of International Standards Organization (ISO) process control event facilities (CONNECT, THAW, SINT, FREEZE and UNCONNECT).

4.2.1.4 Reentrant and Nonreentrant Code

Each task operates in a logical address space of up to four megabytes. There are four types of program segments: pure (reentrant code), impure (nonreentrant code), reentrant library and task common. Each segment is a minimum of 4,096 bytes and can be any size (in increments of 4,096 bytes) up to the maximum available memory.

The segmentation of programs into pure and impure segments allows programs to be shared for efficient multiaccess or multithreading. The pure segment of such a program contains the static, unchanging code or data of the program. The impure segment contains the dynamic work space of the program, plus any impure code such as overlays. Regardless of the number of times a program is required for concurrent use, only one copy of the pure segment is loaded into memory by the operating system, whereas a separate copy of the impure segment is loaded for each invocation of the task.

Reentrant library segments, also composed of pure code, contain commonly used reentrant subroutines. The FORTRAN Run-Time Library (RTL) falls into this category.

Task common segments are data areas accessible to any number of tasks.

4.2.1.5 Memory Management

The memory management system provides optimum use of system memory for user applications. The user need not be concerned with allocating partitions for tasks; memory is allocated dynamically to tasks as they are loaded.

OS/32 regards memory as local memory. Local memory is physically contiguous from location 0. It contains the operating system, user space and system space.

User space is managed by the memory management software. Available memory is allocated on demand to task segments on a first-come, first-served basis. If user space is exhausted, the roll mechanism ensures that high priority tasks take precedence. The memory manager allocates user space on a first-fit basis, always selecting task requirements from the first free area found to be sufficiently large. Any residual memory remains free and can be used by another task. As memory becomes free, adjacent areas are merged to minimize search time.

For large tasks, OS/32 (via the Link Utility) provides the Virtual Task Manager (VTM). VTM is invoked on a task-by-task basis. This provides virtual memory support for those tasks that require it, without imposing conventional virtual memory overhead on the entire system. Through VTM, tasks as large as 16Mb can execute in as little as 128kb of user space.

System space is used by OS/32 for the transient data structures required to support operating system functions. The size of system space is dynamic and can be altered by the system operator.

The roll facility allows the total number of tasks within the system to be independent of the actual amount of memory available. Eligible tasks are rolled, on a priority basis, from memory to disk to make space for more urgent activities. Rolling is commonly used to queue low-priority tasks while high-priority tasks are active.

The roll eligibility of a task is a user-defined option enabled when a task is established. This can be dynamically changed at run-time by the operator, the task itself or by another task in the same subsystem. By using the system display facilities, the operator can request a listing of the status of all rolled tasks.

4.2.1.6 Simple Man-Machine Interface

The man-machine interface is engineered to simplify communication between the user and the system. OS/32 is controlled through an interactive system console by means of an extensive command repertoire.

All commands take the form of a command verb, followed by a list of parameters. Abbreviations, ranging from the minimum unique abbreviation to the full command spelling, are always accepted by the system. Convenient defaults are accepted for many command parameters.

4.2.2 Multi-Terminal Monitor (MTM)

MTM provides a general-purpose programming environment for up to 64 interactive terminal users. The full program development facilities of OS/32 are available to each user, including the powerful command language, an extended version of the CSS, and the utility programs.

MTM allows individual users to operate autonomously, with complete privacy from all other users of the system. Each user can create disk files inaccessible to any other user and cooperating users can define files accessible to any member of the group.

An MTM system, with as many as 65,535 separate accounts, can be readily tailored to meet the security and/or cooperative needs of any installation. A comprehensive range of privileges can be made available to or withheld from any particular account. Signon-time, CPU-time and I/O requests per device classification can be limited to preassigned maximums on an account basis.

Information regarding account privileges and limitations is maintained in the Authorized User File (AUF). This file identifies each authorized user of MTM by account number, relates individual accounts to group accounts and contains each user's password. MTM uses this file to assure the validity of user requests to signon to the system. The system operator builds the file by using the MTM Account Utility, which provides facilities to create, update and generate reports concerning the AUF. This file is accessible only from the system console.

An extensive accounting facility is provided. Through it, accounting statistics can be maintained, and charges assessed, for the utilization of system resources (such as CPU-time, disk storage and I/O) by individual and group accounts, or in summary form for the entire system. Reports can be generated for any given time period from archival files created and maintained via operator commands.

Access to these files can be gained only from the system console; they are unavailable to even the most privileged MTM users.

A key feature of the Accounting Facility is the ease with which modifications and enhancements can be made to make report contents and formats meet specific site requirements.

MTM provides concurrent support for all Perkin-Elmer language processing systems. Each user can develop programs in the language, or combination of languages, best suited to the application. If desired, each user can concurrently compile and execute an application system developed in any mixture of the supported languages.

4.2.3 Reliance PLUS

Reliance PLUS provides comprehensive facilities for the development, execution and maintenance of on-line data base management applications. It manages the real-time aspects of transaction processing so that communications interfaces are transparent to application development.

Reliance PLUS:

- provides simple generation, initiation and control of the application transaction processing system,

- provides data base management that affords both the application programmer and the query user a relational view of data, at the same time ensuring rapid response regardless of the mode of file usage or the volatility of the data base, and
- manages concurrent access and update of the application data base while providing an extremely high level of data integrity.

Applications can be written in COBOL or FORTRAN VII. The data dictionary provides ease of development and maintenance. Queries and reports can be produced, without programming, through the Relational Query Language/32 (RQL/32).

Reliance PLUS supports up to 240 local_ or remote interactive terminals within each application environment. Leased or switched lines can be supported in point-to-point or multidrop configurations.

Users communicate with an application and Reliance PLUS facilities through screen forms. A screen form usually consists of several phrases, each prompting the user to enter information. Each phrase is generally followed by a blank field in which the user enters the appropriate response.

The available screen forms are presented to the user through a menu. An application can have one or more menus and menus can be nested as required.

Each field within a screen form may have associated validation attributes. Validation attributes specify the type of data that can be entered, the range of values or specific values that can be entered and whether or not an entry is required.

Validation attributes are specified by the user during the screen development process. User-entered data is verified against the validation attributes, and if any data does not satisfy the predefined criteria, an error message is displayed and the application is not loaded.

Access to the system is restricted to users who have been allocated and who correctly specify a unique user identifier and a password. Access to specific areas of an application can be restricted to designated users, and users can be limited to specific transaction types.

Further restrictions can be placed on the transactions available at specific terminals. This is accomplished by associating user passwords with individual terminals within the environment, with respect to their physical location. Such restrictions are particularly useful for environments containing terminals that are not located in secure areas.

Unsuccessful attempts to signon to a terminal are monitored. If consecutive unsuccessful attempts are made, the system controller is notified and the terminal is removed from the environment.

Consistent system performance can be maintained by relegating lengthy processing activities to background processing. Interactive transactions have priority over background transactions. The number of background transactions that can be executing concurrently can be limited by the system controller. For increased flexibility, background transactions can be initiated at a specified time of day and date.

More than one environment can operate simultaneously in the same Model 3205 System. For example, a production environment and one or more development environments, or several production environments, can operate concurrently.

Batch programs executing in an MTM environment can access the application data base at the same time as the on-line transactions, with the same enforcement of integrity provisions.

The data base management system (DBMS) provides an indexed file organization, with records referenced by multiple keys. It allows random, sequential and dynamic access, continuous automatic reorganization and ensures access security.

The integrated transaction contoller (ITC) streamlines the transaction processing environment so that programmers can focus on particular applications without concern for communications details.

RQL/32 is the fully integrated query and report writing component of Reliance PLUS.

RQL/32 is a nonprocedural language. Since no programming knowledge is required, it can be used by all personnel. Operation is menu-driven, using screen forms similar to other Reliance PLUS transactions. Queries and reports always access the latest data and can be obtained at any time.

The data dictionary provides a single source for all data definitions in the system. Data definitions can be accessed and manipulated by the interactive user or by an application program. Because data definitions are standardized across applications, productivity improvements are realized through simplified programming and reduced errors.

4.2.4 System Networking

Distributed networking allows the standard facilities of any computer system within the network to be shared by any other member of the network. PENnet provides a true distributed data processing network for geographically dispersed organizations. It requires no previous experience with distributed networking; MTM and/or Reliance users are not forced to learn complicated protocols or even to alter programs or user interfaces. Through PENnet, Model 3205 Systems (or any other Perkin-Elmer 32-bit system) can be linked together using modern international data communications standards. Connections can be made via dedicated communications links (leased line or dial-up) or by an X25 packet-switching network such as PSS or Telenet.

Local area networks (LANs) provide communication among locally grouped computer systems. For example, an organization may have several computer systems scattered throughout a building or cluster of buildings. Perkin-Elmer's implementation of the popular Ethernet LAN provides a solution to the data communications needs of such an organization. Ethernet provides high-speed (up to 10Mbits per second) data transfer over coaxial cable for as many as 1,024 stations.

4.3 APPLICATION ENVIRONMENTS UNDER EDITION VII WORKBENCH

Edition VII Workbench is a standard version of the UNIX™ time-sharing system, seventh edition, and includes the Source Code Control System (SCCS) facilities of programmer's workbench (PWB).

Edition VII Workbench is a multiprogramming time-shared system suited to installations with as many as 128 interactive users. Typical applications include business and scientific computing, telecommunications and text publishing. Available programming languages include:

- C
- FORTRAN VII D
- FORTRAN 77

UNIX™ is a trademark of Bell Labs, Inc.

- Pascal

Edition VII Workbench provides a highly structured operating system with a proven record for improving the productivity of systems and applications programmers. Its structure consists of a central kernel responsible for system administration and resource allocation, a shell that interprets user requests by performing functions that communicate with programs using software channels called "pipes", and a variety of utilities.

Overall, Edition VII Workbench is one of the most versatile of all operating environments. The extensive use of UNIX systems by universities has produced many graduates in the computer science and engineering fields who are experienced and comfortable with UNIX. Another benefit of Edition VII Workbench is its high degree of portability. This provides the user with built-in protection against any loss of software due to hardware obsolescence.

Responsibility for system maintenance belongs to those personnel entrusted with the "superuser" password. The superuser is responsible for the day-to-day operation of the system. This includes such chores as bringing the system up and down as required, adding and deleting users, maintaining file system archives and analyzing the output from various accounting programs.

The kernel maintains control of resources, supervises process switching, scheduling and I/O synchronization and provides memory management.

The shell is a high-level programming and command language that allows the user to control the operating environment. It includes features such as flow-control statements, variables and subroutines. Shell commands are used to read and write files, redirect I/O, and invoke processes that communicate through software channels known as pipes. Various functions can be performed by combining shell commands to create new programs.

The utilities available under Edition VII Workbench provide programs for interactive graphics, electronic mail, phototypesetting, text editing and formatting, and setting mathematical equations in type.

The system also contains facilities for updating and maintaining data bases.

A number of programs, known collectively as Programmer's Workbench (PWB), have been added to serve as software development tools. These tools monitor program changes, recreate earlier versions of programs, collect specifications for jobs and maintain records of dependencies between program parts.

The most distinguishing characteristic of Edition VII Workbench is its file system. Under Edition VII Workbench, a file is simply a collection of bytes with no special format. This feature simplifies the creation and use of files. Files are organized into sets known as directories and each user has his own file directory. The management of files is facilitated by the hierarchical organization of the file directories.

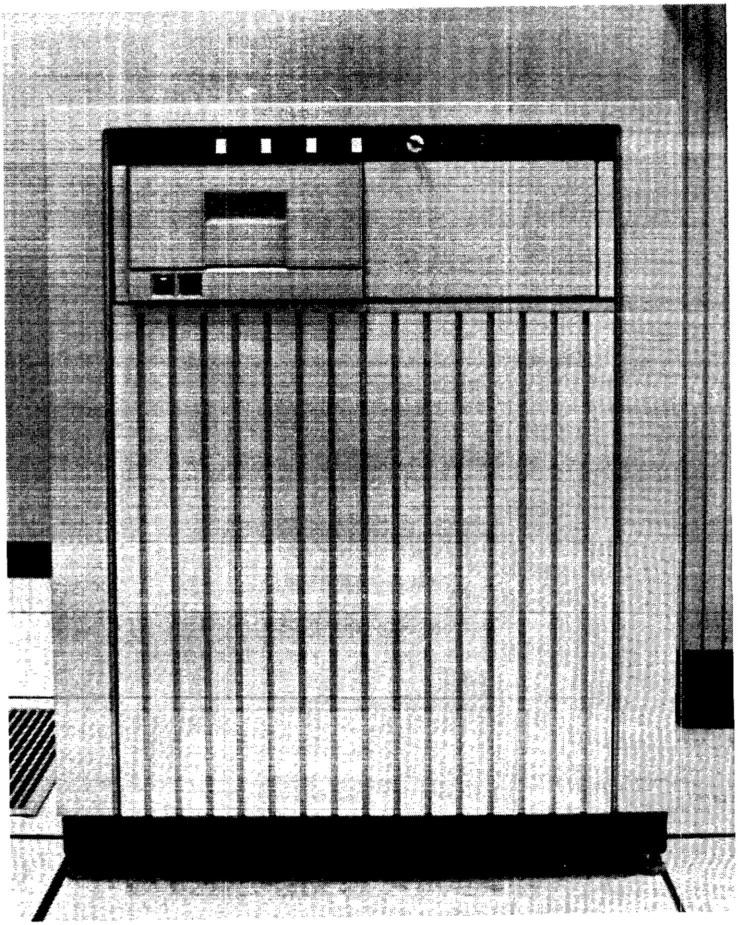
Special files are associated with I/O devices and the system file structure allows directories to contain both ordinary and special files. Each file has an associated protection key that allows individual, group or global access.

The SCCS provides facilities to store, update and retrieve all versions of a source code module. As source files are modified, the SCCS records who made each change and why it was made. To assure maximum efficiency, SCCS stores only the changed data for each derivative. SCCS, in effect, manages source files and provides an audit trail for software projects. The system serves as a custodian of files, permitting files to be numbered using a hierarchical technique.

Another valuable feature of Edition VII Workbench is its wide variety of office automation facilities. These include electronic mail, a variety of editors, a spelling dictionary and a style-checking program. Text formatting facilities include TROFF and NROFF, which provide sophisticated typesetting and table generation capabilities.

Users can communicate with one another through electronic mail. Mail can be transferred to any computer system in the network. When a user logs onto the system, he is automatically informed of any mail awaiting his attention. A user can also communicate interactively with another user, or can opt to send an electronic message to be reviewed at the recipient's convenience. The sender can optionally request confirmation that the mail has been read.

This is a blank page inserted to retain pagination sequence.



CHAPTER OVERVIEW

The Model 3205 System is supported by a complete line of hardware and software documents. This chapter is designed to assist users who wish to determine available documentation and explains how the overall documentation package is divided by subject matter.

CHAPTER 5
MODEL 3205 SYSTEM SUPPORT DOCUMENTS

5.1 INTRODUCTION

No data processing system is complete without a full complement of support documents. Users of the Model 3205 System can depend on a set of comprehensive support documents - both reference manuals and user's guides - that are designed with the user in mind. From managers to programmers to data-entry personnel, all levels of technical expertise are addressed in these documents. The underlying principle of all Perkin-Elmer support documentation is that the users of the system should understand, and as a result, take advantage of all of the features that make the Model 3205 System unique in today's minicomputer marketplace.

The Model 3205 System documentation package reflects more than a decade of experience in the creation of support documents for each of the models in the Perkin-Elmer Series 3200 line of processors.

The sections of this chapter present an overview of the major support documents supplied with the Model 3205 System. The documents are grouped together based on the general user category that each serves.

Within these groupings, subgroups are identified; these subgroups represent more specific areas of related subject matter. The title of each manual is indicated and the corresponding Perkin-Elmer part number appears in parentheses.

All of the manuals identified in this chapter can be obtained by contacting your Perkin-Elmer sales representative. Readers interested in other Perkin-Elmer support documents or in a more detailed description of the contents of the documents listed in this chapter should see the 32-Bit Systems User Documentation Summary, Publication Number 50-003.

The Model 3205 System documents identified in this chapter are arranged into the following functionally related groups:

- Hardware reference documents
- OS/32 reference documents
  - Operations reference documents
    - Administrative
    - Operational
  - System programmer reference documents
    - Instruction sets
    - Programmer reference
    - Data communications protocols
  - Application programmer reference documents
  - Language reference documents

- Edition VII reference documents

5.2 HARDWARE REFERENCE DOCUMENTS

Whether installing, performing maintenance or upgrading the configuration of a system, the engineers depend on the hardware reference documents to supply the information necessary to complete the job. Users of the Model 3205 System receive a comprehensive set of hardware reference documents. These documents provide information detailing installation procedures, upgrade procedures, system preventative maintenance, troubleshooting procedures, and theory of operation analysis. Also, a complete schematic and assembly drawing reference volume, printed on oversize 11" by 17" sheets, is provided.

Figure 5-1 identifies the hardware reference documents that support the Model 3205 System.

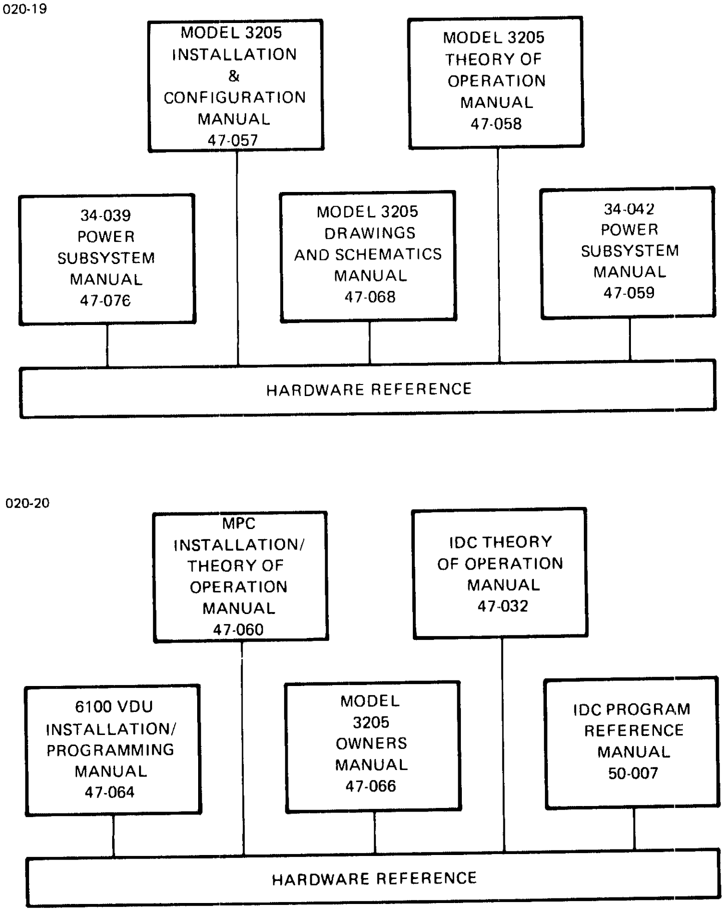

020-19
MODEL 3205 INSTALLATION & CONFIGURATION MANUAL 47-057
MODEL 3205 THEORY OF OPERATION MANUAL 47-058
34-039 POWER SUBSYSTEM MANUAL 47-076
MODEL 3205 DRAWINGS AND SCHEMATICS MANUAL 47-068
34-042 POWER SUBSYSTEM MANUAL 47-059
HARDWARE REFERENCE
020-20
MPC INSTALLATION/ THEORY OF OPERATION MANUAL 47-060
IDC THEORY OF OPERATION MANUAL 47-032
6100 VDU INSTALLATION/ PROGRAMMING MANUAL 47-064
MODEL 3205 OWNERS MANUAL 47-066
IDC PROGRAM REFERENCE MANUAL 50-007
HARDWARE REFERENCE

Figure 5-1 Hardware Reference Documents

5.3 OS/32 REFERENCE DOCUMENTS

The following sections outline the reference documents available for users of the Model 3205 System operating under OS/32.

5.3.1 Operations Reference Documents

The operation and administration of a system such as the Model 3205 System is a complex responsibility. Perkin-Elmer recognizes this fact and supports the system manager and system operator with a number of documents designed to make these tasks easy to understand and, consequently, easy to use.

Such functions as system generation (sysgen), console use, administration of disk storage, establishment and maintenance of the system user account structure, and system usage and accounting are all thoroughly documented in easy-to-follow user guides and reference manuals.

Figure 5-2 identifies the operation reference documents that support the Model 3205 System operating under OS/32.

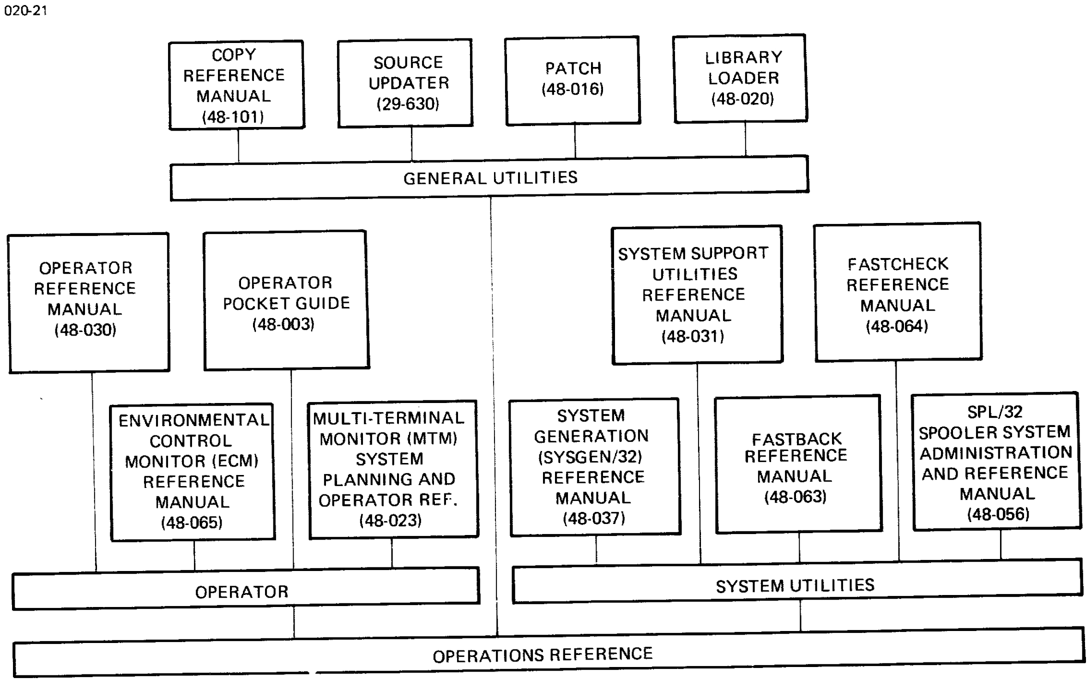

020-21
COPY REFERENCE MANUAL (48-101)
SOURCE UPDATER (29-630)
PATCH (48-016)
LIBRARY LOADER (48-020)
GENERAL UTILITIES
OPERATOR REFERENCE MANUAL (48-030)
OPERATOR POCKET GUIDE (48-003)
SYSTEM SUPPORT UTILITIES REFERENCE MANUAL (48-031)
FASTCHECK REFERENCE MANUAL (48-064)
ENVIRONMENTAL CONTROL MONITOR (ECM) REFERENCE MANUAL (48-065)
MULTI-TERMINAL MONITOR (MTM) SYSTEM PLANNING AND OPERATOR REF. (48-023)
SYSTEM GENERATION (SYSGEN/32) REFERENCE MANUAL (48-037)
FASTBACK REFERENCE MANUAL (48-063)
SPL/32 SPOOLER SYSTEM ADMINISTRATION AND REFERENCE MANUAL (48-056)
OPERATOR
SYSTEM UTILITIES
OPERATIONS REFERENCE

Figure 5-2 Operation Reference Documents

5.3.2 System Level Programmer Reference Documents

The Model 3205 System documentation package contains a number of support documents designed to aid the system level programmer. Detailed information regarding such topics as system instruction sets, programming considerations, operating system services, task linkage, virtual task management and data communications protocols are covered in these documents. When used in conjunction with the appropriate language reference documents (see Section 5.3.5), a thorough foundation of system level documents is available to the user.

Figure 5-3 identifies the major support documents available to system level programmers using OS/32.

020-22

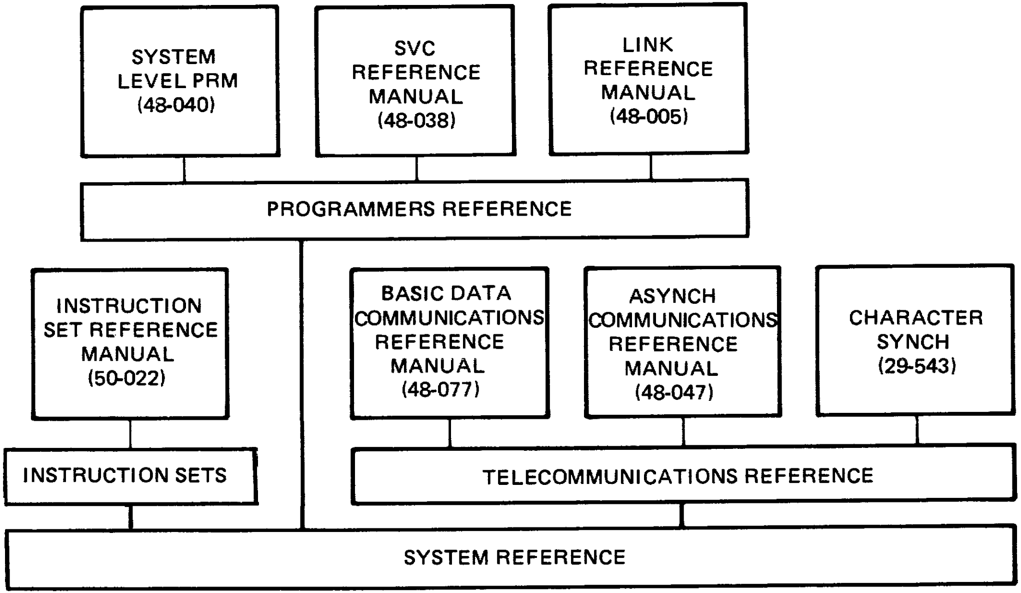

SYSTEM LEVEL PRM (48-040)
SVC REFERENCE MANUAL (48-038)
LINK REFERENCE MANUAL (48-005)
PROGRAMMERS REFERENCE
INSTRUCTION SET REFERENCE MANUAL (50-022)
BASIC DATA COMMUNICATIONS REFERENCE MANUAL (48-077)
ASYNCH COMMUNICATIONS REFERENCE MANUAL (48-047)
CHARACTER SYNCH (29-543)
INSTRUCTION SETS
TELECOMMUNICATIONS REFERENCE
SYSTEM REFERENCE

Figure 5-3 System Level Programmer Reference Documents

5.3.3 Application Level Programmer Reference Documents

Programmers assigned to produce application packages on the Model 3205 System have a number of support documents available to them. Documents dealing with such topics as text editing, program development environments, task linkage, operating system services, programming considerations and real-time environment commands are covered in detail in these documents. When used in conjunction with the appropriate language reference documents, they provide application level programmers with a complete set of support documents.

Figure 5-4 identifies the major documents available to the application level programmer using OS/32.

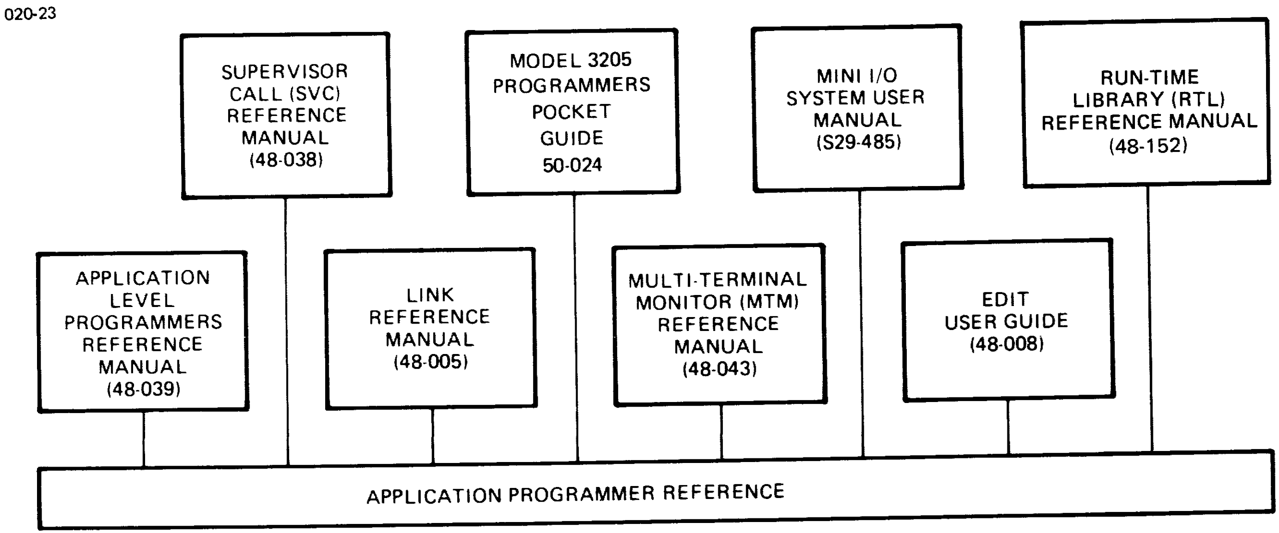

020-23
SUPERVISOR CALL (SVC) REFERENCE MANUAL (48-038)
MODEL 3205 PROGRAMMERS POCKET GUIDE 50-024
MINI I/O SYSTEM USER MANUAL (S29-485)
RUN-TIME LIBRARY (RTL) REFERENCE MANUAL (48-152)
APPLICATION LEVEL PROGRAMMERS REFERENCE MANUAL (48-039)
LINK REFERENCE MANUAL (48-005)
MULTI-TERMINAL MONITOR (MTM) REFERENCE MANUAL (48-043)
EDIT USER GUIDE (48-008)
APPLICATION PROGRAMMER REFERENCE

Figure 5-4 Application Level Programmer Reference Documents

5.3.4 Language Reference Documents

The Model 3205 System supports a variety of programming languages. Each language is thoroughly documented with user guides and reference manuals. Two of the languages feature enhancements specifically designed to maximize the hardware and operating features of the Model 3205 System - Common Assembly Language (CAL) and FORTRAN VII.

Figure 5-5 presents a summary of the language reference documents available to users of the Model 3205 System running under OS/32.

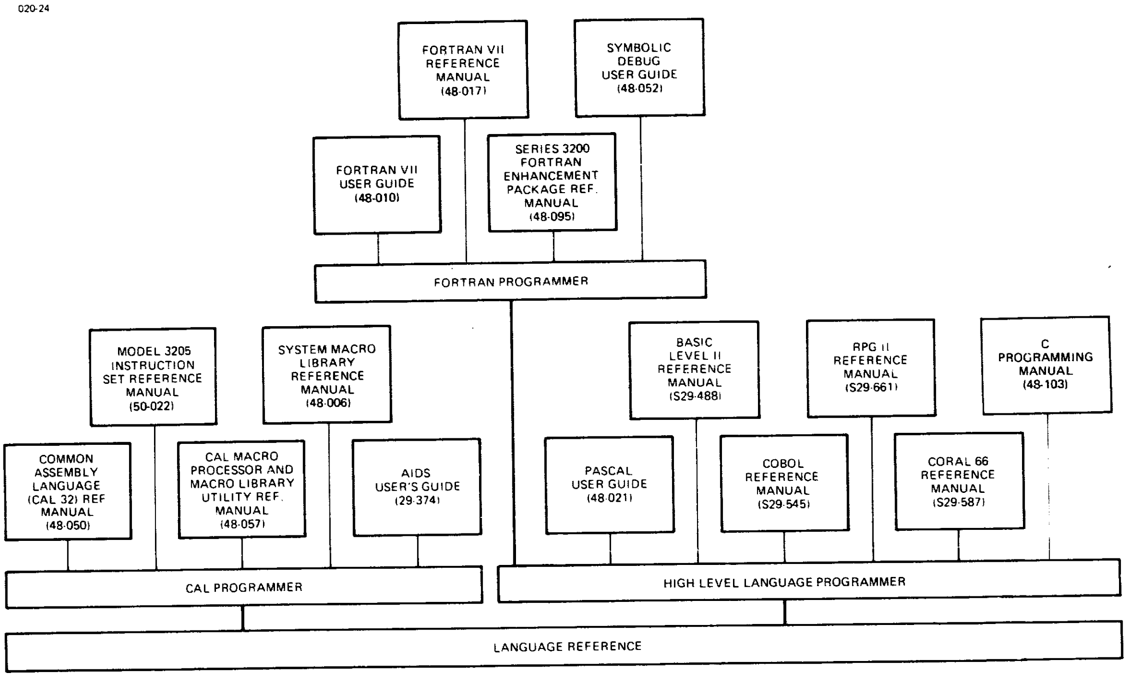

020-24
FORTRAN VII REFERENCE MANUAL (48-017)
SYMBOLIC DEBUG USER GUIDE (48-052)
FORTRAN VII USER GUIDE (48-010)
SERIES 3200 FORTRAN ENHANCEMENT PACKAGE REF. MANUAL (48-095)
FORTRAN PROGRAMMER
MODEL 3205 INSTRUCTION SET REFERENCE MANUAL (50-022)
SYSTEM MACRO LIBRARY REFERENCE MANUAL (48-006)
BASIC LEVEL II REFERENCE MANUAL (S29-488)
RPG II REFERENCE MANUAL (S29-661)
C PROGRAMMING MANUAL (48-103)
COMMON ASSEMBLY LANGUAGE (CAL 32) REF MANUAL (48-050)
CAL MACRO PROCESSOR AND MACRO LIBRARY UTILITY REF. MANUAL (48-057)
AIDS USER'S GUIDE (29-374)
PASCAL USER GUIDE (48-021)
COBOL REFERENCE MANUAL (S29-545)
CORAL 66 REFERENCE MANUAL (S29-587)
CAL PROGRAMMER
HIGH LEVEL LANGUAGE PROGRAMMER
LANGUAGE REFERENCE

Figure 5-5 Language Reference Documents

5.4 EDITION VII REFERENCE DOCUMENTS

The reference documents available to users of Edition VII on the Model 3205 System provide information that ranges from tutorial to reference material and includes a large collection of articles written by a number of authors.

Figure 5-6 identifies the reference documents that support the Model 3205 System running under Edition VII.

020-25

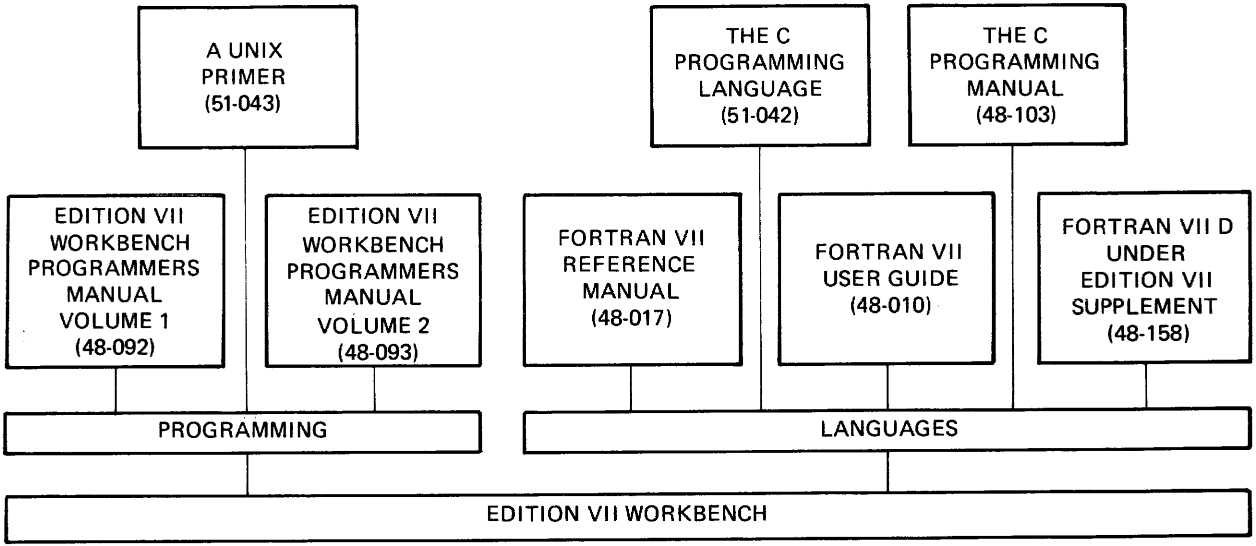

A UNIX PRIMER (51-043)
THE C PROGRAMMING LANGUAGE (51-042)
THE C PROGRAMMING MANUAL (48-103)
EDITION VII WORKBENCH PROGRAMMERS MANUAL VOLUME 1 (48-092)
EDITION VII WORKBENCH PROGRAMMERS MANUAL VOLUME 2 (48-093)
FORTRAN VII REFERENCE MANUAL (48-017)
FORTRAN VII USER GUIDE (48-010)
FORTRAN VII D UNDER EDITION VII SUPPLEMENT (48-158)
PROGRAMMING
LANGUAGES
EDITION VII WORKBENCH

Figure 5-6 Edition VII Reference Documents

GLOSSARY

absolute address

An actual address in main memory. It is also known as real or machine address. An absolute address can be interpreted directly by the control unit and need not be processed by the memory address translator (MAT).

ADCCP

Advanced Data Communications Control Procedures. A full-duplex, bit-oriented protocol used in packet switching networks.

algorithm

A series of instructions or procedural steps used to solve a problem.

arithmetic logic unit (ALU)

Registers and associated circuitry designed in such a manner that arithmetic and logic operations can be performed.

The arithmetic section of the ALU performs arithmetic operations (addition, subtraction, multiplication and division) and logic operations (shift, mask, etc.)

ASYNCH

Asynchronous transmission method. Each transmitted character is preceded by a start bit and followed by a stop bit.

background task

A low-priority task that is serviced only when all high-priority interactive or real-time tasks are inactive.

backpanel

The backpanel is a wiring bus common to all printed circuit (PC) boards.

batch processing

Work that is performed only as facilities are freed from the demands of real-time processing.

BISYNCH

A protocol used for controlling synchronous transmission. Specifies the data link control characters for starting and stopping a message, the type of error checking required, the bytes subject to inclusion or exclusion, and the method by which the data link escape character is used.

central processing unit (CPU)

The CPU of a system reads and interprets coded instructions and data stored in a computer's memory and performs operations as required by a program. The CPU is divided into two sections: the control section and the arithmetic and logic section.

consolette

The consolette provides the user with information on the system's status and control of the system functions.

control section

The control section of the central processing unit (CPU) is the center of the processor system and controls most of the components of the computer.

diagnostic task (d-task)

A user-written task that can execute privileged instructions. It uses relative addressing.

direct memory access (DMA)

The DMA provides fast access to memory for high-speed devices, independent of the processor.

distributed network

A system of geographically dispersed but coordinated computers linked via telecommunications technology.

executive task (e-task)

A user-written task that can execute privileged instructions and executive-level code. It uses real (absolute) addresses, and therefore, has access to operating system data structures.

foreground processing

The normal mode of processing in a real-time system, whereby foreground tasks take precedence, via interrupts, over lower priority or background tasks.

HDLC

High-level Data Link Controller. A full-duplex, bit-oriented protocol used in packet switching networks.

impure segment

A routine composed of instructions and data areas subject to modification during execution. It is also known as nonreentrant code.

input/output (I/O) modules

Printed circuit (PC) boards that enable the user to interface to the system. Examples are 2-line and 8-line communication multiplexor (COMM MUX) module and line printer interface controller.

intelligent disk controller (IDC)

The IDC provides the processor with the capability of writing to and reading from disk media.

interactive processing

A procedure that allows a give-and-take, or conversational, process to be carried on between user and computer. Under a system such as MTM or Edition VII Workbench, resources are apportioned among many simultaneous users in such a way that each appears to apply a dedicated system to the task at hand.

intertask communication

A method by which a task operating in a Model 3205 System can communicate with, and/or control another task in the system.

local area network (LAN)

A method by which closely grouped computer systems (generally within 17km) can communicate at relatively high data transmission rates.

main memory bus

This bus interfaces to memory all components of the system that require memory access.

memory address relocation and protection/memory address translator (MAT)

When a memory address is referenced to a physical address location, the memory address relocation and protection/MAT is disabled and becomes transparent to the memory address. When the address to memory is a virtual address, the memory address relocation and protection/MAT is enabled and translates the virtual address into a physical address location.

memory controller

The memory controller controls the address/data transfers between the processor/direct memory access (DMA) and memory.

memory controller bus

The memory controller bus interfaces the memory controller to the storage device.

memory storage module (STM)

The system memory module (STM) typically consists of 16, 64 or 256 kilobit MOS RAM chips as memory storage elements. The STM memory capacity varies from 25Mb to 2Mb as required.

microprogram

A set of machine-level instructions that directly affect the hardware.

monitor task

A task that provides coordination and control over other tasks by making use of intertask communication and/or SVC interception.

multidrop

A telecommunication configuration whereby a single channel or line serves multiple terminals.

multiprogramming

A method by which several tasks are permitted to share system resources.

multitasking

Procedures in which several separate but interrelated tasks operate under a single program identity. This differs from multiprogramming in that common routines and data space may be used.

Multi-Terminal Monitor (MTM)

A software system available to users of Perkin-Elmer processors that is used to simultaneously monitor activity and provide services to a number of terminals operating concurrently.

nonreentrant code

A routine composed of instructions and data areas subject to modifications during execution. It is also known as an impure segment.

operating system

A collection of software instructions designed to control and coordinate a computer system. It also facilitates the performance of routine operations and maintenance.

operating system space

A contiguous portion of Model 3205 System memory beginning at real address 0 and large enough to contain the operating system.

paging

The process of transferring pages between main and secondary storage. It is useful for allocating a limited amount of main storage among a number of concurrently executing tasks.

point-to-point

A telecommunications configuration whereby a single channel or line services only two terminals, one at each end.

priority scheduling

A method by which tasks are scheduled for servicing according to importance, which is determined by a user-assigned number.

process control

Automatic control of continuous operations or processes achieved by using computers.

program

A plan for the automatic solution of a problem, including a set of instructions or steps that tells the computer exactly what to do.

program address

An address relative to the beginning of the memory that a task occupies. Requires translation by the memory address translator (MAT). It is also known as relative address.

pure segment

A routine composed of instructions and data areas that is not subject to modification during execution. It is also known as reentrant code.

queue

A sequence of items waiting for attention by one or more servicing entities.

real address

An actual address in main memory. A real address can be interpreted directly by the control unit and need not be processed by the memory address translator (MAT). It is also known as absolute or machine address.

real-time

Describes a system under which computations are performed within the actual time that the related process transpires. It allows the results of computations to be used in guiding the process.

receive acknowledge/transmit acknowledge (RACKO/TACKO)

Acknowledge signal to the user that activated the attention signal request.

reentrant code

A routine composed of instructions and data areas that are not subject to modification during execution. It is also known as a pure segment.

registers

A temporary storage device for the information the computer is using in its current operation. A register consists of one or more storage devices to store information. The arrangement of the logic circuits permits input, output and possible information alteration.

relative address

An address relative to the beginning of the memory that a task occupies. It requires translation by the memory address translator (MAT). It is also known as program address.

routine

A sequence of instructions that are designed to perform a well-defined function or series of operations.

run-time library (RTL)

A set of precoded routines that execute in support of a program language environment.

segmentation

Is the technique of dividing a task into functionally distinct portions, such as pure and impure segments.

selector channel (SELCH)

A DMA device capable of servicing up to 16 I/O controllers, independent of the processor.

subchannel controller (SCC)

A device that regenerates the MUX bus for extension of the MUX bus to another cabinet.

subtask

A task belonging to a subsystem and controlled or coordinated by a monitor task.

supervisor call (SVC)

A request to the operating system to perform certain privileged services, such as I/O operation.

symbolic debugger

Software that provides symbolic commands used to assist program debugging.

synchronous data link controller (SDLC)

Uses synchronous data transmission techniques and protocols to transfer data in point-to-point or multidrop telecommunications configurations.

system clocks

System clocks are designed to provide time of day and programmable timer interrupts.

system control modules

Printed circuit (PC) boards required for system control functions in multiboard processor systems; that is, universal clock (UCLOCK), loader storage unit (LSU) and hardware communication assist.

system loading program

The system loading program automatically initializes/restarts the computer system.

system space

A portion of main memory used as a work area by the operating system.

system video display unit (VDU)

The system VDU provides the operator with the means to interface to the system. The VDU can be connected to a remote computer via telephone data lines.

task

A logically related sequence of instructions that has its own unique identity and performs a unique function when executed.

task common

A type of reentrant segment consisting of data available to more than one task.

task control block (TCB)

A logical construction in the Model 3205 System containing various information on the status of a given task.

task management

Controls the use of system resources (other than I/O) by tasks.

telecommunication

The transmission and reception of signals by various electromagnetic means.

time-share

The use of a system for two or more purposes during the same overall time interval.

time-slicing

The allocation of a time interval during which a given task may execute on a processor. At the end of this interval, the task is suspended and another task of equal priority is given execution rights to the processor. Time-slicing is designed to ensure that all tasks of equal priority receive the same level of service.

transaction processing

Processing of a group of related actions entered by one or more terminal operators.

uniprocessing

The method used in a conventional computer system consisting of a single central processing unit (CPU) in which instructions from one or more programs are executed sequentially.

user space

The portion of main memory dedicated to the storage and execution of user-written tasks.

user task (u-task)

A user-written task on the Model 3205 System that uses relative addressing and does not make use of privileged instructions.

virtual memory

Secondary storage that the user can regard as main memory.

ZDLC

Zero-bit Insertion and Deletion Data Link Controller.

2-Line and 8-Line Communication Multiplexor (COMM MUX)

The 2-line and 8-line COMM MUX interfaces the MUX bus of the processor system to half-duplex or full-duplex asynchronous data sets or local terminals.

INDEX