PDP-10 Real-Time Multiprogramming System

Digital Equipment Corporation

July 1970

TABLE OF CONTENTS

I. Introduction

New Real-Time Software

II. PDP-10 Real-Time Software

- classes of service
- response
- interrupt driven tasks
- assignment of job priority
- high priority scheduler
- background scheduler
- timesharing
- batch
- core mangement

III. PDP-10 Hardware

- General Features
- Equipment List
- Computer Systems Group

IV. Applications

- High Energy Physics
- Analytical Instrumentation
- Data Acquisition

V. Typical Installations

- MIT - Project MAC
- Brookhaven National Laboratories
- Oxford University
- University of Munich
- National Institutes of Health
- Faculte of Medicine of Paris
- University of Pittsburgh
- Rolls Royce
- Royal Aircraft Establishment
- Sikorsky Aircraft
- KinOTrol Inc.
- Pennsylvania State University
- Chase Brass and Copper Company Inc.
- Stanford University
- United Aircraft Corporation

Table of Contents (Continued)

VI. Support

. Software Support
. Training
. Maintenance

VII. Appendices

I Real-Time Trap Feature
II Tables

- Response Time
- System Performance

III Supplementary Documentation List

Specifications subject to change without notice.

I. INTRODUCTION

ANNOUNCING NEW REAL-TIME SOFTWARE FOR THE PDP-10

For several years, PDP-10 computer systems have been highly successful in the monitoring and control of real-time processes, serving science, physics, chemical, hybrid, and computer science laboratories, and many other important real-time industrial applications. The PDP-10's multiprogramming hardware and software, multi-level hardware interrupt system, versatile addressing capabilities and many other features are ideally suited for real-time work.

Now, with the Real-Time Monitor, the PDP-10 becomes even more effective in real-time applications. With higher throughput and faster response time, more real-time processes can be monitored and/or controlled simultaneously. Each task or subtask - from the most critical real-time requirement through background timesharing or batch processing operations - is handled according to its response requirements.

A PDP-10 is an integrated hardware and software system.

The focal point of every system is the PDP-10 central processor with its 36-bit word length, floating point hardware, multi-level interrupt system, 16-general registers, dynamic memory protection and relocation hardware and 366-instructions including any size byte, half word, full word, and stack operations.

Hardware organization is word parallel for reliability and speed.

Memory organization is word parallel, multiple ports, with memory timing independent and asychronous among memories and between memory and CPU.

PDP-10 systems offer a complete line of peripheral equipment including disk packs, drums, data terminals, communication equipment, CRT's, plotters, printers, computer-to-computer interfaces, tape and card equipment.

The PDP-10 Real-Time Monitor is an extension of the PDP-10 multiprogramming monitor and provides several levels of response while maintaining high throughput. A hallmark of the PDP-10 software system is its ease of use and the speed with which software may be written and debugged.

Two levels of service are provided for time-critical real-time tasks. Super priority tasks may be driven directly by the hardware interrupt system while other time critical processes may be run under the control of a high priority, software scheduler.

Within the PDP-10 system a group of related sub-jobs may form a single "job" initiated by a single "user". Core management routines ensure that jobs are located in core in strategic areas so as to avoid interference with other jobs or programs. Typically, several real-time jobs (as well as background jobs) reside in core simultaneously and run under control of the hardware interrupt system, high priority scheduler, and multiprogramming hardware.

Interactive time-sharing and batch processing proceed under the supervision of a background scheduler employing a "fairness" or computation time weighted algorithm.

The PDP-10 real-time software extends the capability of the system in several application areas requiring fast response. These applications are given privileged status in their use of the system and its resources. Privileged status includes locking the job in core, that is, to never be considered for swapping or shuffling. Some examples of these jobs are:

Real-time Devices - These devices require immediate access to the processor in response to an interrupt.

Display Systems - The display must be refreshed from a display file in the user's core area in order to keep the display picture flicker-free.

Batch - Batch throughput is enhanced by locking the Batch control CUSP in core.

Performance analysis - Jobs monitoring the activities of the system need to be locked in core so that when they are entered to gather data, they are aware of their state and therefore, can record activities of the monitor independent of the monitor.

Standard PDP-10 system software includes FORTRAN IV, MACRO-10, DDT, TECO, BASIC, COBOL, ALGOL* and many other language processors and programs.

*ALGOL is under development

II. THE PDP-10 REAL-TIME MONITOR

The Real-Time Monitor is a powerful multiprogramming executive which dynamically controls and supervises all ongoing operations of a PDP-10 system. The system provides an optimum environment for real-time, time-sharing, and batch operations. User programs may run in Executive (master), Privileged User, or User (slave) modes and may be scheduled to run by the interrupt system, a foreground scheduler, a time-sharing scheduler, or a batch processing supervisor. Core memory is dynamically allocated by the monitor. There is no need to set up fixed memory partitions as the PDP-10 dual memory protection and relocation hardware provides a means whereby user programs may be loaded in any order and position in memory and programs may share re-entrant code.

The monitor is sophisticated in its file system and in its storing and retrieval of disc files. Retrieval information is tree structured to minimize search time. Furthermore, retrieval information for a specifiable number of files is kept in core so that disc accesses can be kept to a minimum during search operations. When disc access is requested, the monitor optimizes disc latency thereby providing high throughput. For added reliability and security of stored files, retrieval information is always kept on the disc in two places. This ensures that even in the event of a software or hardware "crash" files will remain intact and the system can quickly be restored to service without costly and inconvenient delay. Disc I/O proceeds in parallel with computation so as to maximize CPU usage and provide a multiprogramming environment.

A PDP-10 system is undoubtedly one of the most convenient and easy to use in existence. A simple command language is used to control all operations.

Perhaps one of the most exciting system qualities for the programmer are the facilities provided to speed program development. Using a terminal and the text editing and program debugging languages a PDP-10 programmer can create and debug a program often in less time than the key punching time associated with other systems.

MAJOR PDP-10 REAL-TIME MONITOR FEATURES

- Real-Time Jobs Activated by Hardware Interrupts*

Super priority real-time tasks may be activated by the hardware priority interrupt system. The PDP-10 monitor provides monitor calls through which the programmer may link one or more real-time jobs

*See Appendix I for a complete discussion and specification of this class of service.

directly to the interrupt system. There is no requirement that this linkage be set up at system generation time. One monitor call has as arguments the name of the device which will generate the interrupt and the priority level to which the interrupt will be assigned. Programs, thus scheduled, are run in Privileged User Mode allows the user to do I/O directly, yet offers the security provided by the memory protection and relocation hardware.

When an interrupt occurs on the hardware level to which a real-time device is assigned, the monitor saves the state of the machine unless the unique PDP-10 instruction block-in or block-out is being used on the interrupt level. Saving the state of the machine, includes the sixteen general registers, memory protect and relocate, CPU flags and the contents of certain trap addresses.

The PDP-10 has a unique capability to deal efficiently and at low cost with block oriented devices such as A/D converters, CRT displays, DECtaj and so on. An instruction called Block-In (Block-Out), when executed, will transfer a block of data words to (from) memory with a total CPU overhead of 6.5 microseconds per word. If this instruction is placed in the interrupt trap location, the system will transfer one word of data per interrupt while simultaneously maintaining a word count and a memory address pointer. The service routine for a device so connected is, therefore, this single instruction Block-In (Block-Out). Block-In is analogous to the three-cycle data break connection available on other DEC computers. However, Block-In requires no extra hardware in the devi controller and any device may make use of it.

The advantages of using the PDP-10 Real-Time Monitor facilities to link jobs to interrupt levels are several:

1. Jobs may be initiated (and removed) without building or loading a different monitor.
2. No monitor modifications are required.
3. Jobs may be run in Privileged User Mode.
4. Jobs are not restricted as to priority level assignments and, in fact, maybe raised or lowered in priority as required.
5. Subroutines are provided so that real-time programs may be coded in FORTRAN.

Response time is limited only by the ability of the hardware to service interrupts. For example, if a job placed a BLKI on a given interrupt level, transfer of a full 36-bit word into the job's buffer area would be completed within 6.4μs after occurrence of the interrupt. (No higher priority interrupts in progress assumed.)

Typically, response time, including full context switching, would be on the order of 100μs.

- Programs Can Share Data Areas

The PDP-10 monitor provides a convenient method for sharing an in-core data area between programs. Typically, a real-time program locked in core would gather data into the shareable segment containing a data buffer. When the buffer was filled, the real-time program would notify the system to call in the non-resident analysis program from the disk. The analysis program would come in and start up in accordance with priority assigned to it previously by the programmer.

Programs can be written in FORTRAN with blank or labeled common located in the shareable data segment.

The technique for creating a shareable common area intimately involves the PDP-10 dual memory protection and relocation hardware registers. The shareable high segment contains the common data area with the write protect bit turned off (Monitor calls are provided which will pick up the program segments and manipulate the write protect bit. The FORTRAN programmer calls a MACRO subroutine to use these).

. Assignment of Job Priority*

Jobs may be assigned to any of several high priority queues maintained by the normal time-sharing scheduler. The number of high priority queues to be maintained for real-time tasks (jobs) is specified at system generation time.

There is a separate queue for each priority level. The scheduler scans the high priority queues first (in descending order according to priority) and if there is a runnable job in one of these queues, it will be run. Whenever a high priority job uses its allotted quantum of CPU run time, it will be re-queued to the rear of the priority queue to which it is assigned.

Jobs are assigned to a high priority queue by the execution of a specialized program which runs when the job is logged into the system. A priveleged monitor call (UUO) is executed by this program which specifies any of the allowed priority levels. The priority order of the run queues maintained by the monitor are as follows:

Priority Levels:

N = the highest of high priority levels, where N is specified at MONGEN time.
.
.
3
2
1 = lowest of the high priority levels
0 = time-sharing or normal priority

Because the monitor scans the same queues when deciding which jobs to swap into core that it scans when deciding which job to run, assignment of job priority is a very powerful PDP-10 feature.

One or more real-time programs can be out on the disk (drum) for a great percentage of the time with their relative priorities assigned by the system administrator. Normal time-sharing or batch processing can proceed undisturbed until a real-time clock or some other mechanism (interrupt) that one or more of the real-time program(s) should run. The swapper then scans its queues and swaps in the highest priority real-time job requesting service. When the swap is complete, the scheduler will run the job. Because the PDP-10 is a multiprogramming system, processing continues during the swap period.

---

* Under development

. High Priority-Scheduler* for Real-Time Programs

In a large scale real-time system environment involving multiple and independent processes, it is often desireable to schedule CPU usage. The processes or tasks typically are time critical, in that each has a fixed deadline when response must be complete and, furthermore, the tasks' arrival time (interrupts) are unpredictable and asynchronous. The objectives of the scheduler are to allocate CPU usage such that the maximum number of time critical tasks can be handled and to allow the mix of real-time tasks to change dynamically without stopping operation of the system.

The high priority scheduler operates on priority level 6 and determines which real-time job, among those requesting this class of service, shall run. Typically, a real-time device using this class of service would have a service routine built into the monitor for the purpose of filling one or more buffer areas with data. When one buffer became full the service routine would request that the data processing routine be scheduled by the high priority scheduler.

The purpose of the scheduler is to provide job start up times in the range of 3-30 ms. The scheduler allocates CPU usage to time critical jobs by means of a "relative urgency" or "least-time-to-go" algorithm. This is accomplished by means of a high resolution real-time clock (DKlØ) which the scheduler uses to maintain a table of jobs and their CPU time requirements.

This technique of using a high priority, "least-time-to-go" scheduler to run real-time processes within a timesharing system is unique to the PDP-10.

---

*Reference:

Multiprogramming for Hybrid Computation
by Mark S. Fineberg and Omri Serlin
Fall Joint Computer Conference 1967 (AFIPS Proceedings)

This monitor section is scheduled for release during the first quarter of 1971.

- Fault Protection

Among the primary objectives in any multiprogrammed system is the prevention of any one job interfering with the successful operation of others, and of the monitor. Additionally, real-time systems often have specialized hardware associated with a given real-time task and, of course, it is subject to failure.

The PDP-10 real-time system protects itself and other users against several types of software/hardware faults peculiar to any given job. Among the protection features are:

- Foreground jobs completely protected from background jobs
- Background jobs protected from illegal memory references by foreground jobs.
- Protection against program loops - Real-time clock
- Protection against hardware faults - Real-time clock (too frequent interrupts)
- Protection against execution of privileged instructions - Exec/User mode
- Protection against out of bounds memory references - Memory/Protect/Relocate
- Protection against unauthorized file access - File Management (user assignment of protection levels)

- Communication Between Real-Time and non-Real-Time Jobs

The PDP-10 Real-Time System enables a real-time task or job to communicate with a non-real-time job. This may be accomplished by means of sharing files stored on disk or DECtape if desired. However, immediate access can be gained by means of standard PDP-10 monitor calls (INPUT/OUTPUT) so as to allow, for example, a data analysis job to read or write a buffer in the real-time job's core area. In a typical case, a PDP-10 real-time job would be associated with data acquisition and, thus, core resident at all times. However, a large data analysis program associated with this real-time job would reside on disk and only become core resident when the real-time job had filled a core buffer with data.

Because of the flexibility of the PDP-10 hardware and software, neith the core space for the buffer area nor the back-ground space for the analysis job need be reserved at build or load time. Dynamic memory protection and relocation hardware operating in conjunction with the monitor's core management facilities provide optimum core usage. Thi is unique to the PDP-10.

. Multiprogramming and Memory Allocation

A PDP-10 system is ideally structured to take advantage of multipro-gramming. With a word parallel I/O structure and multi-port memories CPU and I/O processors are independent with regard to memory access.

Memory space is occupied by the resident monitor and, typically, a mix of real-time and non-real-time programs. The only fixed partitic is between the monitor and the rest of memory. The monitor dynamical controls the occupation of memory by the job mix. Real-time programs are locked in core and are not swapped out onto disk or drum. Back-ground programs are swapped out onto disk or drum when, and only when other programs out on disk or drum need to be run. The PDP-10 monito will attempt to set up this exchange of programs between core memory and secondary storage (disk or drum) so that at least one runnable pr gram will always remain in core during the swap. Thus, the actual ti taken by the independent data channel/disk control/disk to perform th exchange will be overlapped with useful computation.

Because the PDP-10 memory protection scheme does not involve fixed partitions, memory space can be efficiently utilized. When real-time core space demands are high, more memory is dynamically assigned to them. Conversely, as real-time demands recede, more space is availab for background (interactive time-sharing and batch) usage.

In addition to conserving core space, the PDP-10 multiprogramming sys reduces the demands placed upon the swapping device. For real-time systems limited to 8-10 interactive users, disk packs may be used for both swapping and file storage. The PDP-10 monitor is fully document in those publications mentioned in Appendix III.

. Real-Time Monitor Calls (UUO's)

A series of monitors calls have been implemented to enable the real-time user to conveniently demand real-time services. They are:

- associate this job or process with a given real-time device.
- place data from this device into a specified buffer area(s) of a specified size.
- output data to a given real-time device.
- run a real-time process on a periodic basis.
- run a real-time process when its buffer is full.
- run a real-time process that this process specifies.
- terminate this process.
- suspend this process (or continue).
- read the time of day.
- specify the relative priority of a process to the real-time scheduler (specified as "D" time - the maximum allowable delay between the filling of a real-time buffer and completion of running the associated real-time process).
- enter/leave real-time mode. (entering locks job in core. See Lock UUO usage described in Appendix I)

. Dynamic Core Allocation

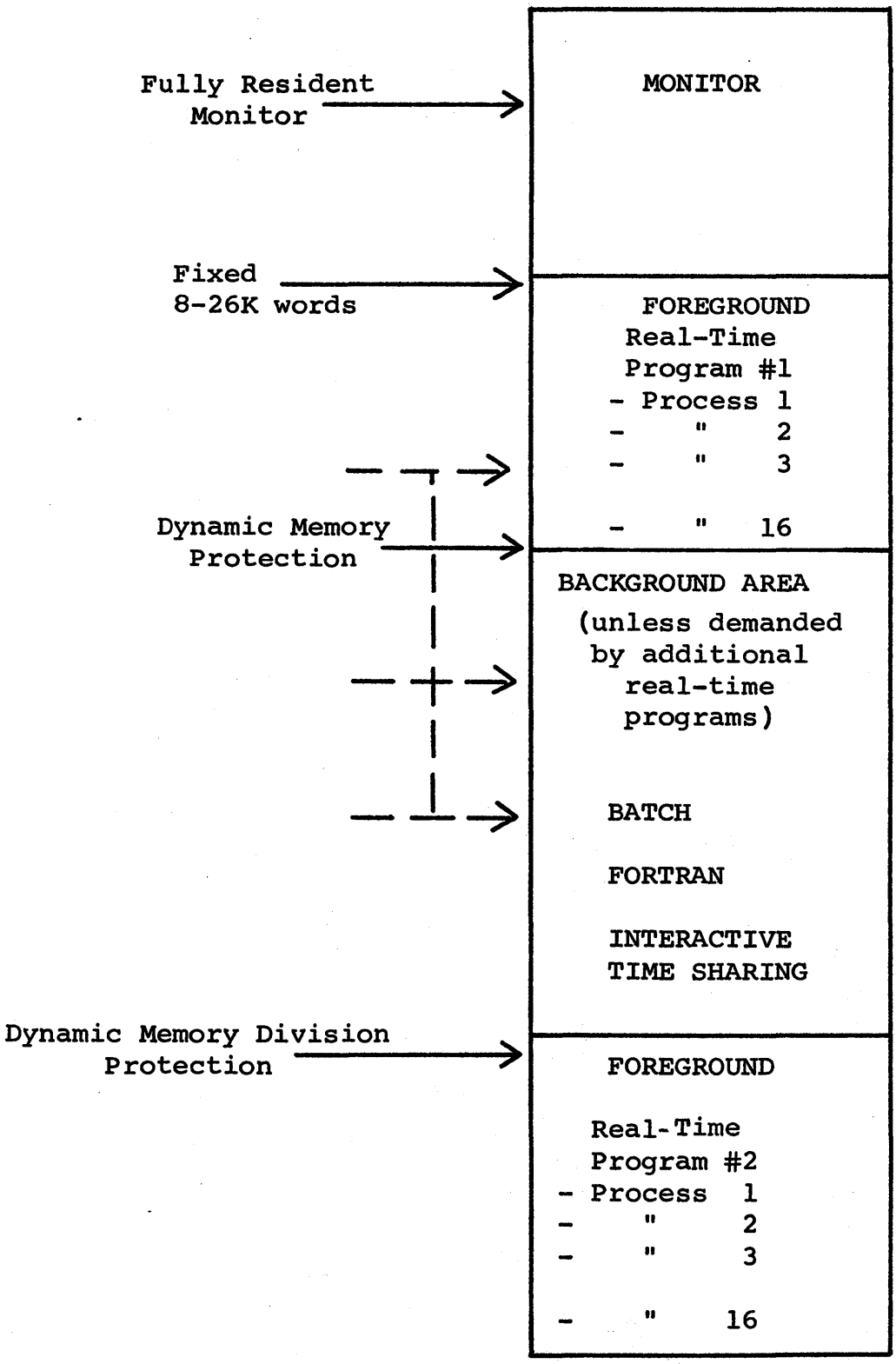
Dual memory protection and relocation registers divide memory into 1024-word blocks. A User mode program can be assigned to an area of memory or to two separate areas, one of which can be restricted to read only. Core space may be dynamically assigned by the monitor. In no sense is memory divided by a fixed partition.

Real-time programs are locked in core during their active period, thereby preventing the possibility of their being swapped out of core at some undesirable point.

Core management routines are provided to place real-time jobs into the least obstructive position in core, that is, either at the top or adjacent to the monitor.

PDP-10 REAL-TIME MONITOR

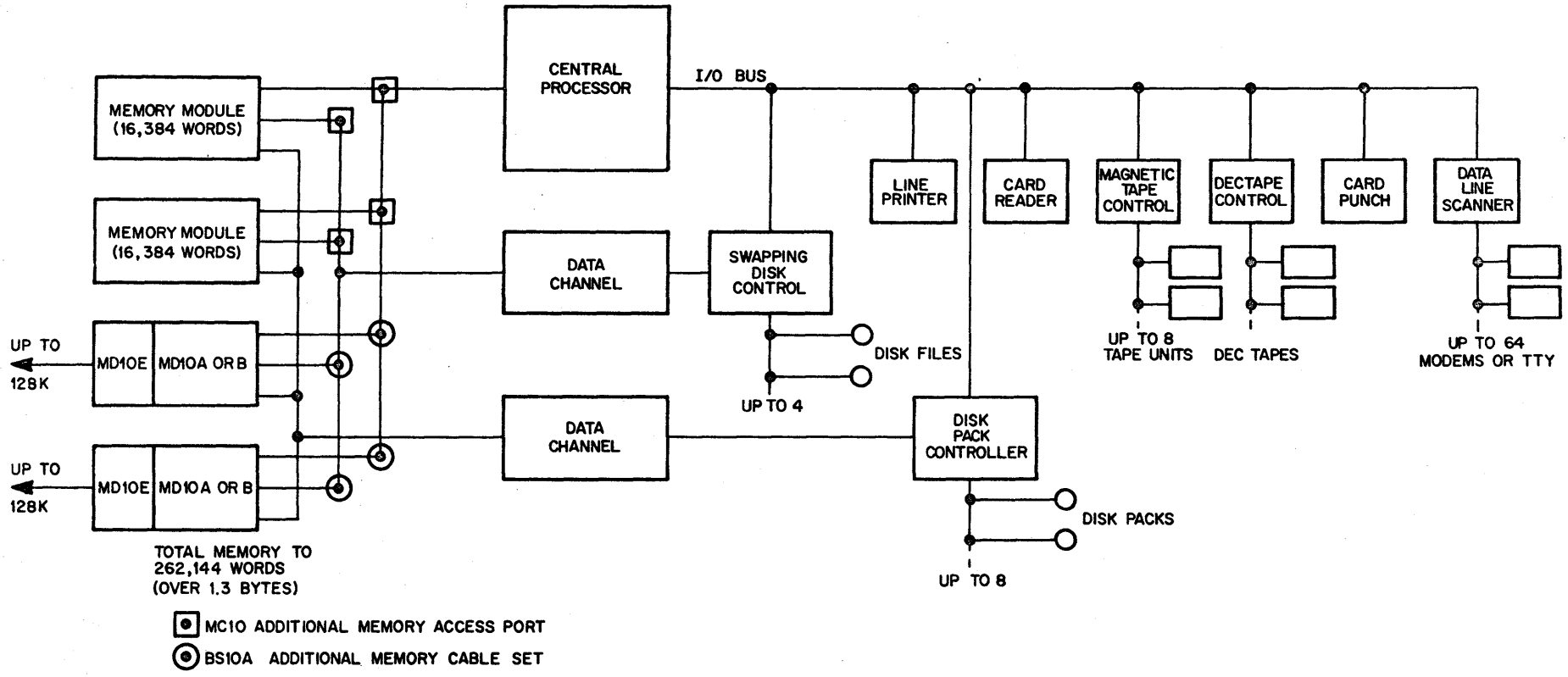
Core Memory Layout



PDP-10 SYSTEM OPERATIONS

Type of Operation	Initiated by	Mode	Comments
1. Monitor Functions incl. . Foreground scheduler . Background scheduler . Swapper (when present) . I/O Service (Std. I/O) . Special I/O . Command Decoding . Call Decoding	Real-Time Clock or Monitor Call or Hardware Interrupts	Exec	Background functions have low priority.
2. Real-Time Tasks	Hardware interrupts Other real-time tasks I/O service routines	Privileged User	Response time ≤ 100μs
3. Real-Time Jobs or Processes	Foreground scheduler	User	Response time of few milliseconds
4. Interactive Time Sharing	Background scheduler	User	Response times of few seconds or less
5. Batch Stream	Background scheduler	User	Longer run times but run less frequently

TYPICAL PDP-10 SYSTEM CONFIGURATION



III. PDP-10 HARDWARE FEATURES

Digital Equipment Corporation's PDP-10 incorporates advanced features providing for a multi-usage capability. PDP-10 systems perform real-time, batch and time sharing simultaneously. PDP-10's make possible a dedicated departmental computer with unprecedented performance.

In addition, the following PDP-10 characteristics result in fast, flexible and efficient system operation - whether in a dedicated or true multiprogramming real-time environment.

. Sixteen General-Purpose Registers - Whether implemented as 150 nsec cycle time TTL integrated circuit registers or as the first 16 locations of core memory, these are 16 general-purpose registers addressable as accumulators, index registers and memory locations. They may even contain executable code. An important speed advantage is obtained when iterative program loops are executed from fast memory.

. Fifteen Index Registers - The right half of each general-purpose register 1 through 15 may be used as an index register. Indexing may take place to any level.

. Multilevel Indirect Addressing - The calculations of an effective address for any instruction may involve any depth of indirect addressing, with indexing at each level if specified.

. Immediate Mode Addressing - An immediate mode is provided for in any of the instruction classes, including floating point instructions. In the immediate mode, the result of the effective address calculation, which may involve indirect addressing and indexing to any depth, is used directly as an operand.

. Sixty-Four Programmable Operators - The PDP-10 provides 64 "programmable operators," codes which trap to fixed (or relocated) locations with the effective address calculation performed. Half of these programmable operations are interpreted by the monitor. Such monitor interpreted calls, consistent throughout all PDP-10 systems, accomplish a variety of functions, including all input/output. The remaining half of the programmable operators traps to the user's own area, and may be user defined.

. Multilevel, Hardware Implemented, Priority Interrupt System - The PDP-10 priority interrupt system is designed to meet the dynamically changing requirements of the typical real-time environment. Seven levels of hardware priority are a standard feature of every PDP-10. Devices may be assigned to any hardware priority through the execution of assignment I/O instructions. It is not necessary, as with other systems, to rearrange cables in order to accomplish this necessary function.

Also, under program control it is possible to enable (or disable) the interrupt system as a whole, individual levels, or individual devices.

Devices uniquely assigned to one level may make use of the BLKI/O (Block-In - Block-Out) feature, which in one step, requiring no programmer attention, will automatically:

1. Identify the source of the interrupt.
2. Update word count and data address.
3. Transmit or receive a word of information.
4. Dismiss the interrupt or, if the block has been transmitted, execute an instruction transferring control to the appropriate end of transmission subroutine.

Interrupts may be generated by execution of appropriate instructions. This is often desirable in order to simulate the presence of real-time hardware which, for some reason, might be unavailable, there being certain development work in progress, for example. Many times software interrupt generation capability is desirable so as to permit the high priority recognition of real-time device interrupts yet permit lengthy related computations to be executed at lower level. Seven hardware priority levels are standard.

. Dynamic Memory Protection and Relocation - In order to protect the overall system from possible errors in the code of the currently running task and to provide for dynamic allocation of core among jobs, dual memory protect and relocation registers are provided. Each User mode program is assigned either one or two segments of memory. One segment may be write protected if desired so as to protect critical data from errors committed by the User himself. One segment may be shared with other users or other programs. Segments are a multiple of 1024 words, may be located anywhere in core, and may contract and expand as required.

Core management techniques, employed by the monitor, ensure the efficient use of core. System programs are reentrant, thus a single shareable copy of FORTRAN IV, for example, is sufficient for all users.

Should a program attempt to reference memory outside of its assigned limits, a hardware trap occurs and the running program is stopped in favor of another.

. Multiplexed I/O Bus - The PDP-10 multiplexed I/O bus provides a 36-bit, full word parallel path between memory and an I/O device controller. Each data transmission instruction will move one word of data between memory and the device controller buffer.

Certain multifunction BLKI/O (Block Input/Output) instructions will automatically move a block of data to or from a device without further programmer attention.

The maximum data rate of the bus is 200 KH_Z words.

. Memory Interleave/Overlap - All memories and processors are capable of asynchronous, overlapped operation. Memories are equipped with two-way interleave switches which, when used, cause consecutive memory addresses to be located in alternate physical memories. The resulting memory overlap often can increase the execution speed of a given program by reducing the effective cycle time of memory.

. High-Speed Channel - One or more high-speed selector channels may be included in any system. Each channel will service up to eight devices. The data paths for both channel commands and data transmission are 36-bit word parallel. Data transmission rates are limited only by memory speeds, i.e., 1M words per second per memory.

. Direct Addressing - The entire core memory, up to 262,144 words, is directly addressable without resort to base registers, displacement addressing or indirect addressing.

. Optimum Instruction Format - The 36-bit PDP-10 instruction word provides for 512 operation codes, of which 366 are wired instructions and 64 trap as programmable operators. The remaining codes are reserved for future expansion and act as no-ops. In all but eight instructions, one of 16 accumulators may be specified. The eight exceptions are I/O instructions, which do not reference an accumulator but, in these, one of 126 I/O devices is specified. All instructions contain an indirect bit through which indirect addressing may be carried to any depth. All permit the specification of one of 15 index registers. All instructions are capable of addressing a full 262,144 words of memory or, alternatively, the address field may contain immediate operands up to 18 bits.

. Logically Complete Order Code - 36 basic instructions divide logically into specific classes and are easily learned. Instruction mnemonics are modular, that is, the characters comprising them are concatenated to form the various instruction mnemonics. Many instructions operate in several modes, with the mode defining the direction of data transfer, where a result is to be left, or defining the condition of test in any of the 128 arithmetic and logical test instructions.

. Variable Length Byte Manipulation - In addition to the instructions which manipulate 36-bit words, there are 64 half-word instructions and a set of 5 byte manipulation instructions which operate on bytes of any size. (Byte sizes are not restricted to submultiples of the word length.) The hardware permits automatic packing, unpacking

and sequential access of any size bytes. Characters are represented throughout the system in ASCII which is a 7-(not 8)level code. Five characters pack conveniently into a character set (as in the representation of symbols). In this case, 6-bit characters are stored six to a word.

. Push-Down Stack Operations - The PDP-10 incorporates instructions for storage and retrieval of data on push-down lists. Even more importantly, there are subroutine call and return instructions which preserve and restore the program counter and the state of the machine flags using a push-down list. These instructions are valuable in implementing reentrant and recursive procedures.

. Floating Point Hardware - Every PDP-10 has full floating point hardware with which the PDP-10 36-bit word length gives a precision of 8 decimal digits in single precision.

. Power Fail Safe - Every PDP-10 is equipped with a power failure detection circuit which, if enabled, will cause an interrupt should power go out of limits. The program can then save all volatile registers.

. Temperature Protection - Strategically placed thermostats detect overly high temperature conditions and, operating in conjunction with their logic, are capable of generating an interrupt.

III. PDP-10 EQUIPMENT LIST

KA10 Arithmetic Processor - central processing unit with floating point and byte manipulation instructions and including:

- 300 character/second photoelectric paper tape reader
- 50 cps paper tape punch
- 10 cps console teleprinter, LT35A
- functional operator console
- multiplexed input/output processor(IOP)
- seven levels of priority interrupt
- real time clock
- power fail-safe
- multiprocessor offset trap locations

KM10 Fast Registers - 16 36-bit high-speed integrated circuit registers used as multiple accumulators and/or index registers and for highly interactive program loops. Replaces the first 16 locations of main core memory.

KT10A Dual Memory Protection & Relocation Registers - multiprogramming hardware for dynamic protection and relocation of reentrant and non-reentrant code.

MA10 Core Memory - 16,384 words, 1.00μs cycle time. Each is supplied with one memory port with cables. Up to three MC10 Additional Memory Access Ports may be added, allowing access to a total of four processors and/or channels. Access time 570ns.

MC10 Additional Memory Access Port - provides the cables and logic to connect an additional processor/channel to a MA10 or MB10 memory.

MD10 Core Memory - 32,768 words, 1.8μs cycle time. Supplied with four memory access ports and a memory cable set for one of these ports. Three additional BS10A memory cable sets are optional to activate the remaining three ports.

MD10E Core Memory Expansion Module - 32,768 words, 1.80μs cycle time. Up to three may be added to each MD10.

BS10A Additional Memory Cable Sets - for the MD10.(Required to activate ports)

DF10 Data Channel - permits data transfers between disk controllers and core memory. It will service up to eight high-speed devices such as RC10, RP10. TM10B also uses DF10.

NOTE: Memory modules may be interleaved so as to establish overlapped operation with resultant reduction of effective cycle time.

DISK SYSTEMS

RC10 — Swapping Disk Control - provides control for up to 4-RD10 disk files or 4-RM10B fast drums. Drums and disks may be mixed on a single RC10 control. Requires a DF10 data channel. Requires at least one RD10 or RM10.

RM10B — High-Speed Fixed Head Drum - high transfer rate, fast access time for program swapping and/or data storage. Each drum provides 345,600 36-bit words of storage. Transfer rate is 4.1μs per 36-bit words (142.6 M bits per second). Average access time 8.3 ms. A total of four RM10B drums per controller are allowed.

RP10 — Disk Pack Control - provides control of up to eight RP01 or RP02 Disk Pack Drives (intermixing is allowed). Requires the DF10 Data Channel. Also requires at least one RP01 or RP02.

RP01 — Disk Pack Drive - each RP01 provides storage for up to 1,299,20(36-bit words on removable , industry compatible, disk packs. Average access time is 62.5 ms. including 12.5 ms average rotational latency. Transfer rate is 30μs/word. Requires RP10 Control. Includes one RP01P pack.

RP02 — Disk Pack Drive - the RP02 provides storage for up to 5,196,800 36-bit words on interchangeable disk packs. Average access time is 62.5 ms including 12.5 ms average rotational latency. Transfer rate is 15μs/word. Requires RP10 control. Includes one RP02P pack.

RP01P — Disk Pack - Pack for RP01 Disk Pack Drive.

RP02P — Disk Pack - Pack for RP02 Disk Pack Drive.

TD10 — DECtape Control - provides control for up to eight TU55 DECtape transports. Requires at least two TU55 transports. (One TD10 control is required with every PDP-10 system).

TU55 — DECtape Unit - reads and writes magnetic tape at 15K 6-bit characters/second rate. (Tapes are 3½" diamater, 260 ft. long and 3/4" wide.) Tape units are bi-directional and redundantly recorded. Each tape has a directory, allowing random access to up to 22 user files. (Two DECtape units are required per PDP-10 system.)

TM10A Magnetic Tape Control - controls up to eight tape transports. Either 7- or 9-channel (or combination of both). Requires at least one magnetic tape unit of the types shown below. Magnetic tape unit types may be intermixed on a single control.

TM10B Magnetic Tape Control - same as TM10A but provides for data channel operation. Requires a DF10 Data Channel.

TM10C TM10B Modification Kit - provides necessary components for converting a TM10A Magnetic Tape Control to a TM10B.

TU20A Magnetic Tape Unit - reads and writes 9-channel USASI standard magnetic tape at 45 inches/second and a density of 800 bits/inch.

TU20B Magnetic Tape Unit - reads and writes 7-channel industry standard tape at 45 inches/second and densities of 200, 556, and 800 bits/inch (36K characters/second).

TU30A Magnetic Tape Unit - reads and writes 9-channel USASI standard magnetic tape at 75 inches/second and density of 800 bits/inch (60K characters/second).

INPUT - OUTPUT DEVICES

Punched Card Equipment

CR10A Card Reader - reads 80-column punched cards at up to 1,000 cards/min. (800 cards/min. in systems using 50Hz power.) Card hopper and stacker capacities are 1,000 cards.

CP10A Card Punch - punches cards at a rate of 200 cards/min. when punching in all 80 columns. A maximum rate of 365 cards/min. is possible when only the first 16 columns are punched. Card hopper and stacker capacities are 1,000 cards.

Line Printers

		Characters	Lines/Min.	Columns/Line
LP10A	Line Printer	64	300	132
LP10C	Line Printer	64	1,000	132

Plotters

XY10 Plotter Control - interface for CalComp 500 and 600 series digital incremental plotters.

		CalComp Plotter Model	Step Size	Speed (Steps/ Minute)	Paper Width (Inches)
XY10A	Plotter and Control	XY10 (565)	0.01 inch	18,000	12
			0.005 inch	18,000	
			0.1 mm.	18,000	
XY10B	Plotter and Control	XY10 (563)	0.01 inch	12,000	31
			0.005 inch	18,000	
			0.1 mm.	18,000	

DATA COMMUNICATION EQUIPMENT

Data Line Scanner - Data line scanner provides on-line service of up to 64-communication lines. Accommodates any device which uses eight level serial teletype code at speeds up to 100 kilo-baud. Full duplex with local copy and half duplex data modes are available on each line serviced.

DC10A Control Unit - the scanner and control unit for the DC10 communication controller provides 4-units of cabinet space and power supplies for various combinations of line equipment.

DC10B 8-Line Group Unit - provides teletype interface for up to 8 local lines, full duplex. May be used with duplex or full duplex with local copy data sets. When used with data sets, data set will provide automatic answer but no control over data set is possible. Requires one unit of cabinet space in a DC10A or DC10E.

DC10C 8-Line Telegraph Relay Assembly - provides conversion from local to long lines using full or half-duplex facilities. Requires two units of cabinet space in a DC10A or DC10F.

DC10D Telegraph Power Supply - the standard line voltage supply used with DC10C (120V dc at 2 amperes). No additional cabinet space required.

DC10E Expanded Data Set Control - provides expanded control of eight data sets in the DC10 system. Requires two units of cabinet space in a DC10A or DC10F. Needed for control of data sets.

DC10F Expander Cabinet - provides eight units of cabinet space and power supplies for expansion beyond capacity of DC10A.

TELETYPES AND TERMINALS

LT33A Teleprinter - 33TS machine (KSR-33, friction feed)

LT33B Teleprinter - 33TY machine (ASR-33, sprocket feed, automatic reader control XON/XOFF feature).

LT35A Teleprinter - VSL312HF machine (KSR-35, sprocket feed).

LT37AC Teleprinter - KSR-37, sprocket feed, 60Hz operation only. Also suitable for use with Bell System 103-type data set or equivalent.

DISPLAY SYSTEMS

VP10 Point Plotting Display Control - operates at either of two maximum plotting rates. Low rate 10 KC (one point every 100μs). High rate is 50 KC (one point every 20μs). Number of addressable points along each axis is 1024. Control interfaces to a customer-supplied oscilloscope (Tektronix Type RM503 or equivalent) or to a CRT display.

370 High Speed Light Pen - for use with VP10.

Other Display Equipment - See following section (Computer Systems Group) and LDS-1 Brochure.

MISCELLANEOUS

DA10 PDP-8 or PDP-9 to PDP-10 Interface.

DK10 Programmable Real-Time Clock - unit is supplied with a crystal oscillator which provides a resolution of 10us. Interrupt interval can be program selected from 10μs to 2.65 s (other crystals and frequencies may be specified). For other clocks, see also following section (Computer Systems Group).

GP10M General Purpose Interface Cabinet - includes cabinet, two 728 power supplies, one 844 power control, indicators, end panels, fan, convenience outlet with fans, and BS10A/15 foot cable set.

BS10A Cable Set - provides cables for interconnecting I/O devices. Set includes two BC10A, one BC10B and one BC10C cables. Available in 5,10,15,25 and 35-foot lengths (1.5, 3, 4.5, 7.5, 10.5 meters).

CAB-9A Cabinet - with full-length single doors front and back, without indicator panel or end panels.

CAB-9B Cabinet - with full-length single doors front and back, with indicator panel but without end panels.

CAB-9C Cabinet - with snap-on covers on front and single door on rear, without indicator panel or end panels.

CAB-9D Cabinet - with snap-on covers on front and single door on rear, with indicator panel but without end panels.

COMPUTER SYSTEMS GROUP

DIGITAL's Computer Systems Group provides custom and limited production computer peripheral equipment and custom computer sytems to meet customer requirements in a variety of applications areas. The groups of technical staff members work with customers and DIGITAL sales engineers in the development of specifications and the design of customized systems for virtually any customer application.

Customized computer peripherals and systems are fully tested and carry the standard DIGITAL warranty. Custom equipment is fully documented with operational and service manuals and is supplied with completely documented test routines. All standard and custom equipment designed by DIGITAL can be included in standard service contracts, if desired, at the expiration of the warranty period.

Multi-processor system applications are recognized by DIGITAL as presenting particularly stringent hardware and software system design requirements. The Computer Systems Group has an experienced team of specialists dedicated to multi-processor applications who can provide extensive applications assistance in: system configuration and design, system specifications, system program design, system test, documentation and installation. In-house assembly of multi-processor system in the exact layout of the final system and close liason with the customer during the phases of extensive testing assures the cutomer a smooth installation process and a minimum time to system operational status on-site.

- Priority Interrupt Level Expansion

General

Interrupt expansion of the PDP-10 may be accomplished by the attachment of an external electronics package. The device is tied to both the I/O and memory buses and requires one memory port into the low memory. It consists of an External Interrupt Control (EIC) and an External Interrupt Group of 16 sublevels. External interrupts are arranged in descending priority with priority of individual lines determined by cable assignment. The unit, itself, is assigned to one of the standard seven priority levels under program control. The concept of the EIC is that electronics within the unit will find the highest priority flag currently requesting an interrupt. Upon finding a requesting flag the EIC uses the memory bus to store in memory an address unique to that flag (and a jump to subroutine instruction). Thereafter, the EIC requests an interrupt on its assigned priority level. The CPU then traps (subject to the usual rules with regard to higher priority levels being not active, etc.) and executes the instruction previously stored in memory by the "EIC". 3.1 microseconds later the program counter (PC) and processor flags will have been stored and the PC will be sitting at the first instruction of the subroutine associated with the interrupt flag.

Nested interrupt servicing requires a minimum of overhead. Performance is summarized below.

Features

- Overhead to reach an interrupt related subroutine is minimized. (3.1μs for non-nested interrupts)
- Contains hardware to detect and identify 16 interrupts arranged in descending priority.
- Unit may be enabled or disabled by assignment to any of the 7 standard priority levels.
- Standard PDP-10 PI levels may be individually enabled or disabled.
- Individual interrupts may be set or reset, enabled or disabled.
- There are 3 flip-flops per external interrupt. They are: enable/disable, interrupt request, interrupt recognized.
- Unit accepts both pulse and level change inputs from external devices. (-3 volts = 0; 0 volts = 1)

Timing

Nested Interrupts:

Entry Overhead = 12.5μs

Non Nested Interrupts:

Entry Overhead = 3.1μs

. PDP-10, PDP-11 Interface

The PDP-10/PDP-11 Interface may be specified to connect up to five PDP-11 computers directly to PDP-10 memory. The interface is not a block transfer type of connection, but rather allows a PDP-11 to store/retrieve data and execute programs in PDP-10 memory directly.

The interface is divided into two logical parts; a control section containing connections to the PDP-10 memory bus, multiplexing elements, and control elements. In addition, each PDP-11 has a complete set of control and data transmission logic. Data is passed 36-bits (plus parity) in parallel so that each PDP-11 can operate with a minimum of PDP-10 memory cycles.

The PDP-10/PDP-11 Interface is designed for those applications involving high data rates and maximum programming flexibility.

The system consists of the following units:

10 I/O Bus Converter

This unit, converts the standard PDP-10 I/O bus from negative levels to TTL compatible levels of ØV and +3V, enabling the attached devices to be built from M-Series modules. This unit also contains the CROBAR inhibits required during the Power-up and Power-down sequence of the PDP-10.

Memory Multiplexer and Level Converter

This section of the interface contains the negative to M-Series level-conversion logic, the parity generation and checking logic and the memory multiplexing and priority allocation logic. This unit, invisible to both the PDP-10 and PDP-11 software, will be connected to the -10 Memory Bus only.

- PDP-11 Interface (continued)

As the major section of the interface, the PDP-11 interface is connected to the converted (+v) 1Ø I/O bus, the PDP-11 Unibus, and the Memory Multiplexed bus which connects, daisy chain fashion, from the Memory Multiplexer, to each of the PDP-11 Interfaces. (Maximum of eight)

One of these interfaces is provided for each PDP-11.

It contains all the main registers, available to the programmer. Which are:

1) Protection and Relocation Registers A & B
2) Control Register
3) Address & Data Registers A
4) Address & Data Registers B

The units are connected as shown in figure 1.

- Operation (proposed)

INITIAL The PDP-10 software treats each of the attached PDP-11's as separate units, as each interface has a unique device number. (This in no implies that the service routines will not share a lot of common code.) The 11 is incapable of interrupting the 10 in any way until the 10 enables the interface with a CONO. The data path to the 10's memory is similarly closed until the 10 opens the path by setting the appropriate protection and relocation registers.

DA15/10

PDP-15 to PDP-10 Memory Interface DA15/10 - This interface makes it possible for a PDP-15 to read and write PDP-10 memory directly. The PDP-15 addresses the PDP-10 memory eighteen bits at a time and PDP-15 peripheral equipment can store (or retrieve) data directly into (out of) PDP-10 memory. In effect, PDP-10 core acts as an extension of PDP-15 memory. Thus, for example, with a DA15/10 interface PHA software which normally requires an 8K word PDP-15 can be run in a 4K PDP-15 with only 2K words of memory being used to replace the otherwise missing 4K of PDP-15 memory.

The PDP-10 can define core bank addresses for PDP-15 under program control. A control buffer is provided to provide the necessary interprocessor communications and interrupt capability.

Requires GP10M.

Display Equipment

VB10I Interactive Graphics Terminal - The VB10I Graphics Terminal is a highly interactive display system for the PDP-10. The VB10I system was designed to allow complete user freedom and flexibility under normal PDP-10 time sharing. The basic hardware system consists of a display connected directly to PDP-10 memory through a special memory interface. Several important features included in the VB10I display are memory protection and relocation, character mode operation, vector mode, and subroutining.To allow complete user-display interaction, the VB10I graphics terminal has a Function Box, Light Pen and a Rand Tablet as options. However, the important feature of the VB10I system is the extensive software package available from DECUS to control the display. This package permits a user unfamiliar with the hardware instructions for the display to generate and manipulate display pictures from a higher level language (FORTRAN or LISP).

The VB10I software package contains all of the basic routines for displaying vectors, points, and text, and for controlling the scale and intensity of the picture. The user can define his own number space or sets of number spaces and the viewports on the display CRT where the pictures are to be displayed. Pictures can be generated as a set of sub-pictures or nested sub-pictures. Any part of a picture which falls outside of the user's number space or window will not be displayed. Sub-pictures and pictures can be blanked from the display and unblanked at a later time. Core space used by a picture can be released and reused for generating new pictures. The VB10I package also contains all of the routines necessary to create three-dimensional pictures. The user can rotate the picture by changing the pitch, yaw, and roll coefficients of the picture or zoom-in or back off from a picture by changing the viewport value.

The VB10I Graphics Terminal provides several levels of user interaction. Simple data inputs can be entered through the function box to the teletype. The identification of lines or points within a picture can be accomplished with the light pen. Perhaps the most important interactive feature of the VB10I terminal is that a user can input graphical data through the Rand Tablet and "see" that data on the CRT in real-time. Applications are numerous. An architect could draw new structures, a mathematician could input new curves, an electrical engineer could add new connections during circuit design, all with no more difficulty than drawing on a piece of paper. The VB10I Graphics Terminal can solve almost all complex graphics requirements with hardware and software that is already available and running in production environments today.

A complete description of VB10I software is available from both DIGITAL and DECUS.

Alphanumeric Terminals - CRT based terminals may be connected to the PDP-10 through PDP-10 communications equipment. Both storage tube and refreshed alphanumeric only systems are available. The most compatible units for connection are those following full-duplex, ASCII, RS-232-B specifications. Line widths should be at least 73 characters.

AD10 Analog Input Subsystem - The type AD10 Analog Input Subsystem combines a reliable solid state multiplexor with a high resolution analog-to-digital converter for real-time applications of data acquisition and reduction of time dependent analog voltage signals from such laboratory devices as gas chromatographs.

The differential input multiplexor provides complete programmable gain selection over the entire input range of ± 10 mv to ± 10.24 volts full scale. An automatic gain selector feature allows two gain selection modes of operation to be controlled by computer command. In the program mode, the software designates one of eleven available gain ranges. In the auto-ranging mode, the multiplexor automatically selects the gain range for optimum subsystem performance.

The AD10 subsystem, under software control, selects the desired analog input channel, provides the proper gain, and digitizes the input voltage at a maximum rate of 10,000 samples per second. Resolution is 14 bits including sign (15 bits including sign is optional).

The basic subsystem can service 64 input channels with expansion capabilities to 192 channels. Input gate modules provide plug-in channel activation, allowing field add-on capabilities.

Expansion - The basic Analog Input Subsystem contains 16 active channels, but expands to 64 channels with the installation of one Type AD10C Dual Channel Input Module for each additional pair of channels. Expansion to more than 64 channels requires the Type AD10B 64-Channel Multiplexor Expander Unit. For example, a system that requires 192 channel capacity would include two Type AD10B Expander Units. One Type AD10C module must be installed for each pair of channels to be activated in an expander unit.

The following designations are assigned to the subsystems components:

AD10A The basic 64-channel multiplexor and analog-to-digital converter, including eight AD10C dual channel input modules (16 active channels) and the computer interface.

AD10B The 64-channel multiplexor expander unit; no AD10C modules are included.

AD10C The dual channel input modules with 20 Hz filters. Filters with cut-off frequencies of 10 or 100 Hz are available if specified at the time of order. Each channel filter on the same modules must have the same cut-off frequency.

CAB-9A The cabinet which houses the computer local interface.

CAB-9A The cabinet which houses the basic AD10A.

CAB-9C An additional cabinet (same as above) which houses the optional multiplexor expander unit, AD10B. A maximum of two units may be installed in this cabinet.

Does not require a GP10M.

Nuclear Instrumentation

PHA Pulse Height Analysis ADC's

The Computer Systems Group is well qualified to interface any of several nuclear physics oriented analog to digital converters to PDP-10 and/or other PDP computers. The ADC's themselves are supplied by the customer and shipped to our factory for installation in the system. Connection may be made to the PDP-10 over the I/O bus, Direct-to-memory, or through a small computer. Data input may take the form of a list of conversions stored in alternate buffer areas in core or a memory increment mode where the ADC word specifies a memory address and the contents of that memory location is incremented by one. Both singles and coincidence modes are available.
One extremely attractive concept is to interface the ADC's to a PDP-15 and interface the PDP-15 directly to PDP-10 Memory with a DA15/10.

FI10 Flag Input Unit - This 36-bit pulse input unit generates a program interrupt request whenever one or more of the lines connected to it changes from a zero to a one. The central processor will respond to this interrupt by reading in the entire 36-bit word and then locating the bit(s) which are ones. The PDP-10 instruction JFFO (find first one and jump) is particularly useful in performing this function.

This unit recognizes the absolute voltage changes on the input lines and variations in rise times are unimportant. The unit contains a 3-bit hardware register to allow its assignment to one of the 7 PDP-10 priority levels. The data register is double-buffered so as to allow synchronization and thus avoid loss of any information.

Customer equipment should make connection to the logic by means of the W031 cable connectors which are provided.

Requires GP10M.

DS03C Contact Interrogation Unit - One of the significant costs incurred in the scanning of multiple contacts can be the cost of wiring. To minimize this expense, the DS03C uses a bus-system in which the contacts are interrogated and the data lines are shared, thereby reducing the number of lines needed to transmit the data to the central processor. This contact interrogation approach dictates that both sides of the contact must be available. Each interrogation provides up to 36 bits of contact status information.

Up to 128 words of contact information can be handled by the control section of a DS03C. Each word is both sequentially and randomly addressable.

Requires a GP10M.

Multiprocessor Hardware

DT01C I/O Bus Switch - The DT01C is an Electronic Switch which enables two PDP-10 central processors to share peripherals. Each processor may request the use of the switch, and may have it when it becomes available. An override feature is built into the DT01C for emergency switching. The purpose of the 36-bit test buffer simulates a common peripheral shared by both computers but accessed only by the first to select the bus switch. The second purpose of the buffer is to act as an interprocessor buffer accessible by both computers.

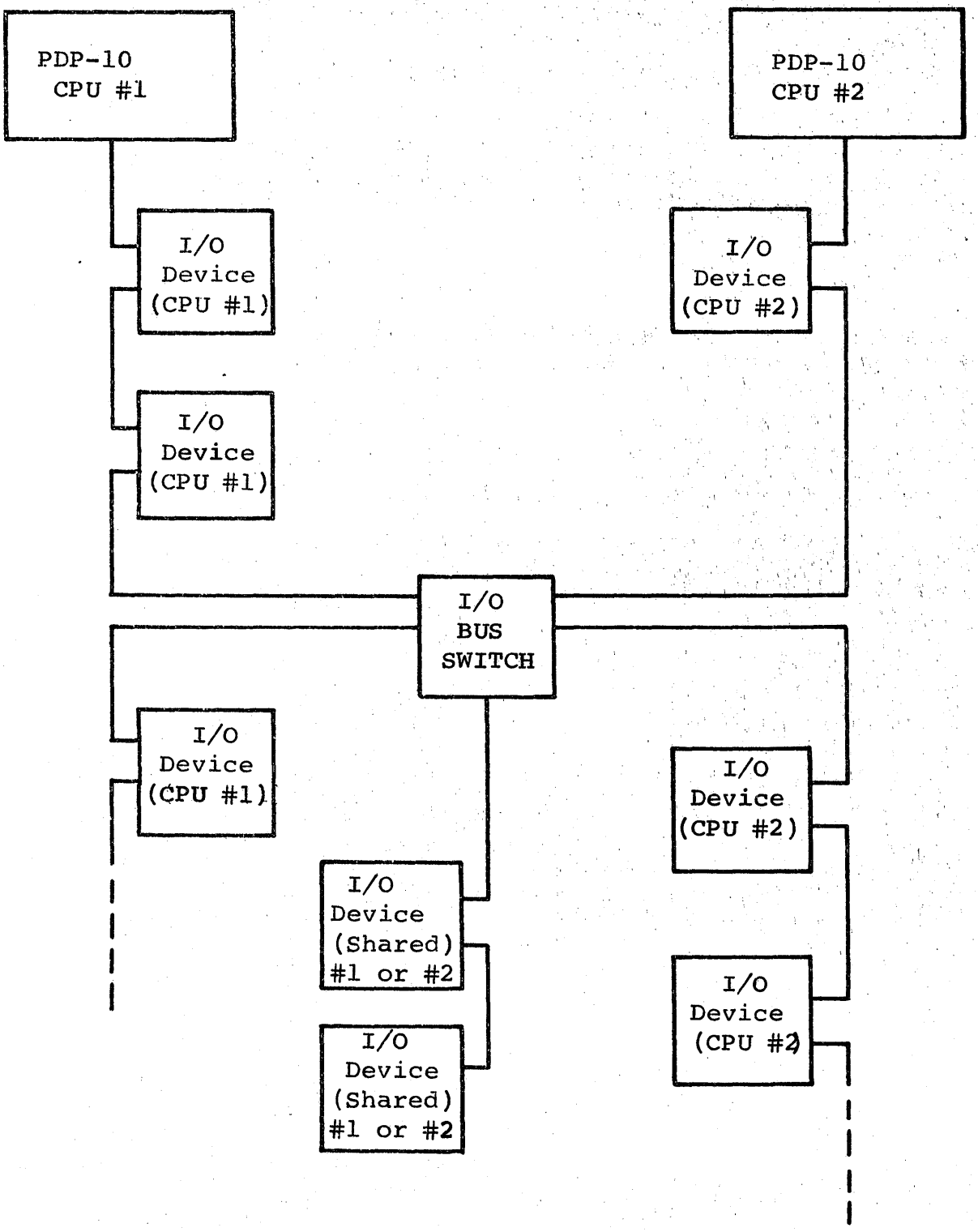

PDP-10
CPU #1
PDP-10
CPU #2
I/O
Device
(CPU #1)
I/O
Device
(CPU #1)
I/O
Device
(CPU #2)
I/O
BUS
SWITCH
I/O
Device
(CPU #1)
I/O
Device
(Shared)
#1 or #2
I/O
Device
(Shared)
#1 or #2
I/O
Device
(CPU #2)
I/O
Device
(CPU #2)

Figure 1

DT02A Multi Processor Peripheral Access Control Unit - The DTØ2A Peripheral Access Control Unit provides all control functions required by two central processors such that peripherals may by accessed by both processors. Each peripheral to be accessed by both central processors must be provided with DTØ2B Peripheral Switches. Up to seven peripherals may be controlled by the basic DTØ2A.

DT02 Peripheral Switch - The DTØ2B Peripheral Switch operating under the control of a DTØ2A will select the peripheral with which it is associated and connect it to a central processor's I/O bus.

There must be one DTØ2B switch for every central processor that is to have access to the shared peripheral. (See Figure 2)

Installations with more than two central processors will require an expenaded Peripheral Access Control Unit.

IV. APPLICATIONS OF PDP-10 REAL-TIME SYSTEMS

There are a number of typical PDP-10 system configurations that serve to illustrate the real-time capabilities of both the PDP-10 hardware and software. Because of the modularity and expansion capabilities of PDP-10, DIGITAL can supply a hardware-software combination unusually well suited to the application.

Historically, computers have had to be dedicated to a single task in order to achieve optimum performance. Real-time systems have often been inadequate in their computational power. Computational strengths have implied a lack of real-time capability. Monitor software has been either non-existent or comprehensive but so restrictive as to offer no possibility of doing real-time work. PDP-10 hardware and software, supplied by DIGITAL, combine a multiple task real-time capability with spooled batch capability and on-line, interactive, time-sharing.

• High Energy Physics

A great deal of effort, applied on an international scale, has gone into the process of automating the measurement of data contained in bubble chamber film. The problems involved are especially difficult because bubble chamber film is "noisy" and the measurements to be made require great accuracy.

The measurement system known as PEPR (Precision Encoding and Pattern Recognition) uses a precision CRT system and controller, interfaced to a PDP-10 computer, to "steer" a line segment, produced by the CRT, along the bubble track images on the film. Using pattern recognition techniques, the computer program acquires a digital image of the useful information held on the film. As data is acquired, it is pre-processed. The extracted information is used to further direct the measuring equipment. Speed is essential and speed can only be attained with a computer system capable of both high speed on-line I/O operations and computational power of the highest order. PDP-10 is perfectly described by these requirements.

PEPR systems are installed both in the U. S. and in Europe. DIGITAL EQUIPMENT CORPORATION has supplied not only computers but also several digital controllers for the CRT systems.

. Analytical Instrumentation

PDP-10 provides some unique features to meet the needs of a multi-instruments multi-experiment analytical system. Many laboratories have several different types of instruments whose characteristics are quite dissimilar. Furthermore, the computer system must provide data reduction as well as data acquisition capability. A growing laboratory will have expansion requirements for more or new instruments, more processing and more on-line storage.

Due to the modular nature of PDP-10 hardware and software, expansion of the system is completely practical. The PDP-10 CPU characteristics include 36-bit word length, floating point hardware, plus over 360 instructions so one can immediately recognize that a PDP-10 is a large machine.

A typical analytical instrument system would include a multiplexed A/D converter with wide dynamic range. Analog signal paths are three-wire; two shielded signal leads plus analog ground. Up to 192 channels can be accommodated. The A/D is 12-bit plus sign with an auto-ranging option to provide automatic selection of gain range.

As instruments input data to the system, other processing simultaneously takes place. The multiprogramming real-time monitor provides concurrent computation and I/O operations.

With the PDP-10 hardware priority interrupt structure, any I/O device can be assigned to any hardware priority. As system requirements dictate, these assignments can be made so as to optimize the system responsiveness to real-time tasks. Other competing designs require that cabling actually be rearranged in order to accomplish the same results.

After data has been accumulated, data analysis programs produce the desired results. The file structure concepts implemented by the PDP-10 monitor include the ability to do tree searches and the ability to choose several different levels of protection for a user's files.

High-speed I/O paths make possible efficient overlapped computation with I/O. Data paths are full-word parallel.

. Data Acquisition

The term data acquisition as applied to a real-time computer system implies a system capable of not only the on-line placement of data in core storage but also the output of this data to secondary storage for later analysis.

PDP-10 data acquisition systems are capable of both the on-line acquisition of data by a variety of means and, simultaneously, the reduction of raw data so as to produce final results.

PDP-10 offers an unusual combination in that great computational capability can be combined with high-speed overlapped I/O operations.

In a typical system, high-speed data acquisition might proceed in either of two ways. First, data might be input directly to memory. If several data streams were involved, a high-speed multiplexor could be interfaced to the word parallel memory bus. A single memory cycle only would be required to input or output a 36-bit word. Even high-speed transfers would result in no CPU interference as long as the CPU and high-speed multiplexor were accessing different memories. Should the same memories be accessed by both the CPU and I/O multiplexor (channel-processor), the memory bus priority hardware would resolve the conflict with, at most, the CPU losing a single memory cycle as I/O overhead.

Slower speed devices (to about 20KC word rate) can practically and economically be interfaced to the multiplexed I/O bus.

Device service routines within the monitor oversee the input (or output) of data, provide control and status checks, assign the interrupt structure and detect error conditions. Real-time User mode programs communicate (through UUO's and monitor commands) with the data acquisition equipment device service routine.

As data is accumulated, it is formatted for output to the disk. PDP-10 disk file structure provides a master file directory containing individual user file directories. User file directories provide the tree structure necessary to efficiently locate and output information. Real-time and other users of the system may assign several different levels of protection to their disk files. Access may be limited to members of one project only, for example, or read only access might be granted.

Background programs, whether part of a Batch stream or not, may pick up the data for analysis upon notification of a real-time process.

V. TYPICAL INSTALLATIONS

Massachusetts Institute of Technology - Project MAC

Project MAC Artificial Intelligence Laboratory uses a dual processor PDP-10/PDP-6 system that includes shared memory and an I/O bus switch which allows sharing of peripherals. The system includes many special devices. Currently, the PDP-6 controls the motions of a mechanical arm through a digital servo network, while the PDP-10 operates a color display built by DIGITAL.

Dual processor software is under development by laboratory personnel. A type 340/346 display is used to study system loading and observe the passage of jobs through the system. The various job queues and their occupancy - maintained by the scheduler - may be displayed.

A chess playing program of national fame has been developed for the PDP-6/10 by laboratory personnel. They have also produced a music compiler capable of producing compiled output for six simultaneous voices. With this compiler, tempo, pitch, key, and musical note embellishments are all programmable. Unlike normal instruments, the computer is self tuning. The chess and the music compiler are available from DECUS.

Brookhaven National Laboratories

Brookhaven Laboratories, Upton (Long Island), New York, have established a National Neutron Cross Section Center based on a PDP-10 time-shared computer and a high speed CRT display system.

Of major consideration in the purchase of the PDP-10 was the ability of PDP-10 hardware and software to directly complement and support graphics and graphics terminals. The PDP-10 provides large scale graphics systems with the support, maintenance, and simultaneous access of a data base by multiple users. In addition, it provides high speed data busses, a powerful instruction set, and time-sharing and interactive languages.

Other graphics systems include: Case Institute, National Institutes of Health, Royal Aircraft Establishment, Kodak, Project MAC, various PEPR installations, Stanford's CAI project, and University of Utah.

The Brookhaven On-Line Data Facility was set up within the Physics Department to perform high-energy counter and spark chamber hodoscope on-line computer experiements. The equipment, together with a dual-processor PDP-10/PDP-6 system, is set up in vans located in the main experimental hall of the AGS accelerator.

Since mid-1965, the facility has been running five to ten real-time jobs simultaneously. With the addition of the PDP-10, the number will increase. Thus the time-sharing aspects of the PDP-10 are of significant importance to the customer.

This on-line use of a powerful computer in high energy physics has demonstrated that research can be much more efficient that the traditional off-line analysis of bubble chamber film, especially when the experiment has high data rates and low cross section.

Oxford University

The PDP-10 at the Department of Nuclear Physics, Oxford University, operates with a PDP-7 in nuclear research. The PDP-7 supports three vector displays and monitors a linear accelerator, while the 48K PDP-10 provides computational power for complex m thematical modeling. The PDP-7 also includes customer built interfaces to devices of other manufacturers.

University Munich

The University of Munich, in Garching, Germany, uses a PDP-10 for nuclear physics research. The 32 K timesharing system operates a 10/40 monitor with a PDP-8/I for data acquisition. A direct memory-to-memory interface (DA25) built by Computer Special Systems couples the PDP-10 and PDP-8.

Communications software to support the interface was provided by DIGITAL. The software allows high speed transfers in both directions so that data acquired by the PDP-8I can be passed to the PDP-10 for processing, and formatted data can be sent back to the PDP-8/I for real-time CRT display.

The A/D converter data is collected in the PDP-8I memory using a double buffering scheme. When one buffer is filled, the A/D converter starts filling the second buffer as the PDP-10 empties the first one. The PDP-10 is used to analyze the incoming data and create real-time display information which is displayed by the PDP-8 concurrently with data acquisition.

Pulse Height Analysis software was supplied with the system by a vendor employed by the University, and tested in Maynard prior to shipment of the PDP-8/I system. The application software allows data acquisition in both singles and group mode from four A/D converters as well as various types of PHA display formats with light pen routines to permit moving of marker, locating peaks, etc.

FORTRAN data reduction programs developed at Rochester University will be used to process the data and to obtain final results.

As an additional benefit the PDP-8/I teletype can be used as a PDP-10 terminal for program development. PDP-8 programs can be assembled by the PDP-10 using PAL-10 and, after assembly, may be loaded into the PDP-8/I for execution.

The Munich system includes a KA10 processor with fast registers and dual memory protection and relocation registers, 32 K of MB10 memory, 2 DECtapes, 2 Magtapes, the memory-to-memory interface (DA25), the PDP-8/I with 8 K of memory, CRT Data Break Display with a VR12, a data break multiplexor, 10-ND A/D converters, 8 scalers, even registers and A/D converter configuration selection switches. System expansion is planned.

National Institutes of Health

The National Institutes of Health, Bethesda, Maryland, has installed a large PDP-10 swapping system to serve as the hub of a laboratory-wide computer complex. In addition to providing time-sharing services for the institute, the PDP-10 is coupled to a powerful CRT display processor, type VB10, for research in biomedical image processing.

The VB10 is a highly interactive display processor which operates directly on the PDP-10 memory bus. N. I. H. has written a complete software package, containing all the routines needed for generating vectors, points, and text. With display service routines - supplied by DIGITAL - added to the swapping monitor, the system provides dynamic displays in real-time simultaneously with time-sharing.

The software package lets the user generate and manipulate display files from FORTRAN and LISP; there is no need to understand display hardware. A Rand tablet, for the input of graphical data, is supported by a complete software package that links the tablet input with the user's display file. Complete software documentation is available from DECUS.

The installation includes 64K of core memory, a line printer, a card reader, two swapping disks, 6 DECtapes, 3 magnetic tapes, and will be expanded to include more memory and disk packs.

Faculte De Medicine De Paris

This PDP-10, located in a computing center of a medical research laboratory, performs statistical analysis of data stored in massive magtape files as well as other computing functions under time-sharing. Future plans call for links between the system and PDP-8 terminals located up to 150 miles away.

The blood bank is currently developing an information storage and retrieval system so that the procedure of maintaining records and scheduling appointments for several thousand blood donors may be automated. When fully implemented, a computer local to the blood bank will be interfaced to blood analyzers and communicate over a high speed link with the PDP-10. Remote terminals will be concentrated by the blood bank's computer so that operating personnel can interact directly with the PDP-10 and its data files.

University of Pittsburgh

The PDP-10 at the University of Pittsburgh is in the Graduate School of Public Health. It is being used in biomedical image processing currently related to chromosome studies, but the customer hopes to branch out to other areas such as X-ray processing, X-ray enhancement, etc.

An automatic microscope locates metaphase white blood cells (a stage in cell division when chromosomes are most easily observed) on slides and signals a PDP-7 which digitizes the image with a flying spot scanner. The PDP-7 then signals the PDP-10 which accepts the data and stores it on an RD10 disk. The PDP-10 then performs the analysis. The PDP-7 and PDP-10 are interfaced through a GP-10, with logic built by the customer.

The system has 32 K of core memory of mixed speeds, DECtape, a card reader and a line printer. A swapping monitor is used although only three terminals are employed, one of them being the console teletype on the PDP-7. The PDP-7 software operates in two modes: local PDP-7 mode and PDP-10 terminal mode.

Rolls Royce

The system at Rolls Royce Bristol Engine Division, Bristol, England, uses a PDP-10 in conjunction with PDP-8's and two ICL computers. The PDP-8's monitor and control the test beds for the Concorde's jet engine as well as other engines readied for on-line testing. The data they collect is fed to the PDP-10 which, in turn, routes it to the ICL machines which provide the computational power. Results of the engine analyses are stored in files which are managed by the PDP-10 and easily accessible to researchers using time-sharing terminals. The terminals can also be used to change the parameters of the tests by initiating appropriate commands to the PDP-8's.

Royal Aircraft Establishment

At the Royal Aircraft Establishment at Farnborough, England, a PDP-10 monitors the structural testing of the Concorde supersonic commercial transport. The results of these tests will lead to the craft's certificate of airworthiness. The project has been labeled one of the "most elaborate structural tests ever carried out."

The tests dynamically apply loads to the fuselage of the plane, under continuous computer control, and continuously monitor the stress effects. The test program will continue throughout the service life of the Concorde.

The testing system also uses two of DIGITAL's PDP-8/I computers, one to control the operation of the loading jacks and the other to monitor operations.

A VB10 CRT display will be used to examine accumulated data and assist technical personnel in their efforts to analyze and interpret test results. The VB10 includes hardware for generation of vectors, characters, and increments. A high-speed light pen and remote slave display are included. The VB10 is interfaced directly to memory and is fully capable of generating a display without processor overhead. DIGITAL provided the monitor software to drive the display. A package which allows the user to program the display in FORTRAN IV is available from DECUS.

Sikorsky Aircraft

The PDP-10 at Sikorsky is interfaced to an Applied Dynamics AD-4 100 V analog computer and is used to perform helicopter design simulations. Software for the hybrid interface was supplied by Applied Dynamics and written by Applied Programming Technology, Sudbury, Mass.

KinOTrol Incorporated

KinOTrol Incorporated, a consulting firm in Houston, Texas, has interfaced a Hybrid Systems, Inc., SS-100 general-purpose analog computer to a PDP-10. The interface was designed and implemented by KinOTrol personnel.

At rates exceeding 1000 per second, the system provides interactive solutions to differential equations encountered in the chemical, oil, and steel industries; in oceanographic and biomedical research; and in engineering and construction. The system is also used in simulation and optimization studies.

A/D and D/A data is transferred directly to and from PDP-10 memory and the SS-100 while control and command information is transmitted and received over the PDP-10 I/O bus. FORTRAN IV callable subroutines, written by the customer, complete the package.

Pennyslvania State University

The PDP-10 in the Electrical Engineering Department's "Hybrid Computing Laboratory" is interfaced to an EAI 680 analog computer. The PDP-10 functions as a logic controller for the analog computer, and serves in simulation studies - for research and instruction - performed under time sharing.

Engineering applications of the system include high-speed analog data conversion from tape and remote inputs, control optimization studies, system simulation, network design, biomedical simulation, pattern recognition, and solution of complex systems of partial differential equations.

The system is also being used to research such diverse subjects as weather forecasting, human performance, building design, ionosphere wave propagation, and mining systems.

For instructional purposes, the system can be operated from 8 console booths, each with a KSR-33 teletype. Through time-sharing, students can work on a common problem, previously set up on the analog plug-board, supplying parameters and conditions of their own choice. Or, the students may work on different problems by using various sections of the plugboard. The students, who are required to have previous analog and digital (FORTRAN) programming experience, work on such problems as parameter optimization, Monte Carlo methods, and difference approximations to partial differential equations.

The system includes 16K of core memory, a 346/340 display with a light pen, DECtape, a teleprinter, 8 teletypes, and the hybrid interface.

The hybrid interface, together with the MACRO subroutines callable from FORTRAN IV, were supplied by EAI. The interface is an EAI 693 system with 24 A/D channels (multiplexed), 14 bits with 20 μsec conversion time and 8 D/A (4multiplying type), 15 bits with 40 μsec conversion time.

Chase Brass & Copper Company, Inc.

The PDP-10 at Chase Brass in Montpelier, Ohio, acts as a time-shared management information system. Its major functions are to: maintain a central data bank, perform manufacturing control functions (including production scheduling, demand forecasting, order shipment, etc.), provide management reports, and perform miscellaneous management and engineering calculations.

Through time-sharing, management has immediate access to the status of any order, process, or inventory. At the same time, this data is available to programs which control, schedule, or forecast plant needs. The system eliminates manual record keeping by collecting, on-line, the data need for sound management decision-making.

Orders, schedules, etc. are updated directly from console teletypes and from an on-line data collection system tied directly into plant processes. Plan personnel using the teletypes are requested to enter specific data in a fixed format. Thus the system is self-tutoring and does not require extensive employee training. Response is so fast that each operator appears to have exclusive use of the computer.

The on-line data collection sub-system includes relay contacts to determine the status of plan devices and a multiplexed low-level integrating digital voltmeter to read the voltages of a variety of measuring instruments. The relay contacts are interfaced to the PDP-10 via the Contact Interrogation Unit (CIU) which can be expanded to 4608 individual contacts. The status of 36 of these contacts may be read by a single PDP-10 instruction.

The type AF04 Integrating Digital Voltmeter and Scanner (IDVM) was chosen for this system for its ability to read many low-level signals while providing excellent noise isolation. The Privileged User IOT mode is employed within the PDP-10 time-sharing monitor system to allow user programs to have direct access to the CIU and IDVM without causing calls to the monitor. This allows real-time user programs to be run in a time-sharing mode.

System control functions include production scheduling, demand forecasting and shipping. Like the data acquisition programs, the control programs run under the time-sharing monitor; they are resident in core only when active. The programs may be written in any PDP-10 programming language - FORTRAN IV, COBOL, BASIC, or MACRO-10. Control programs may have access to the central data file as data upon which to base control decisions. Control programs also have access to all system peripherals such as line printers, DECtape, and disk.

The management reporting system is implemented in the same manner as the system control functions, via user-level programs written at Chase Brass. The system is designed to provide specific information upon request or when exceptional conditions exist which require management decisions. Its basic philosophy minimizes the volume of reports and maximizes the value of information generated; overall emphasis is upon management control by exception.

VI SUPPORT

DIGITAL EQUIPMENT CORPORATION owes a large measure of its success to the high level of technical competence of its staff. All field support personnel are factory trained in their various specialties.

Customer personnel are encouraged to attend training courses given by competent instructors at our plant. PDP-10 documentation includes all relevant drawings, manuals, program listings, etc. PDP-10 software specialist are available for on-site assistance at all times. Particular emphasis is placed on having a software specialist available on-site for several weeks immediately following delivery. Factory personnel are available for consultation and for backup support.

. Training

Formal courses are held at our factory on a scheduled basis. Computer time and all course materials are provided.

Programming/Operation Course (4 weeks)	- Covers all PDP-10 programming systems concepts, including monitor details.
Maintenance Course (5 weeks)	- Covers central processor and memory. Familiarization with DIGITAL logic diagrams, flow charts. Covers I/O systems, hardware priority interrupt.

. Documentation

Maintenance Manuals - Two sets of maintenance manuals are provided with each system.

Engineering Diagrams - A complete set of engineering drawings, flow diagrams, logic diagrams, and wire lists for each system will be provided. Microfilm copies are available at extra cost.

Software - Two complete sets of manuals are provided. A complete set of maintenance (diagnostic) programs are provided. As software is improved or new programs are released, customers may elect to receive them.

. Software Support

Personnel are factory trained and located near installations. Personne are available upon request to assist with familiarization of standard PDP-10 software and programming techniques. Support personnel are responsible for customer satisfaction with PDP-10 software and work closely with both the customer and our factory to ensure that any problems are quickly remedied.

. Field Service

DIGITAL maintains a large, well-trained field service staff in order to provide customers with continued reliable hardware performance.

Maintenance can be provided either through a maintenance contract or by means of on-call hourly service. Several different maintenance contracts are available. All contract customers pay only a monthly charge covering all labor and all parts. Field service contracts include scheduled preventive maintenance.

A complete explanation of DIGITAL field service is detailed in a document available from the local field service organization and is entitled "DEC Computer Maintenance Services and Prices."

RTTRP - REAL TIME TRAPPING UUO

THE RTTRP UUO ALLOWS TIME-SHARING USERS TO DYNAMICALLY CONNECT REAL-TIME DEVICES TO THE PRIORITY INTERRUPT SYSTEM, AND TO RESPOND TO THESE DEVICES AT INTERRUPT LEVEL FROM THEIR USER PROGRAMS (WRITTEN EITHER IN MACRO-10 OR FORTRAN-IV). REAL TIME JOBS CAN CONTROL THEIR DEVICES IN TWO WAYS: BLOCK MODE OR SINGLE MODE. IN BLOCK MODE, AN ENTIRE BLOCK OF DATA IS READ BEFORE THE USER INTERRUPT PROGRAM IS RUN, WHERE AS IN THE SINGLE MODE, THE USER INTERRUPT ROUTINE IS RUN EVERY TIME THE DEVICE INTERRUPTS. THERE ARE TWO TYPES OF BLOCK MODES: FAST BLOCK MODE AND NORMAL BLOCK MODE. THE RESPONSE TIME TO READ A WORD OF DATA IS 6.5 US FOR FAST BLOCK MODE AND 14.6 US FOR NORMAL BLOCK MODE (N.B. THESE ARE THE TIMES TO COMPLETELY SERVICE THE INTERRUPT). IN SINGLE MODE THE RESPONSE TIME MEASURED FROM THE RECEIPT OF A REAL-TIME DEVICE INTERRUPT UNTIL THE START OF THE USER CONTROL PROGRAM IS 100 US. A NOTE OF CAUTION: THE RTTRP UUO IS NOT FOOLPROOF. IMPROPER USE OF ANY OF THE RTTRP FEATURES COULD CAUSE THE SYSTEM TO HANG. THE DESIGN OF THIS FUNCTION GIVES THE USER AS MUCH FLEXIBILITY AS POSSIBLE. HOWEVER, TO MAKE THIS FEATURE POWERFUL, SOME SYSTEM INTEGRITY HAD TO BE SACRIFICED. SINCE THE USER PROGRAM IS RUN IN USER MODE WITH ALL OF THE AC'S SAVED, THE MOST COMMON ERRORS ARE PROTECTED AGAINST. A LIST OF THE RULES AND POSSIBLE PITFALLS APPEARS BELOW.

THE RTTRP UUO ALLOWS A REAL TIME JOB EITHER TO PUT A BLKI OR BLKO INSTRUCTION DIRECTLY ON A PI LEVEL OR TO ADD HIS DEVICE TO THE FRONT OF THE MONITOR PI CHANNEL CONSO SKIP CHAIN. WHEN AN INTERRUPT OCCURS FROM THE REAL TIME DEVICE IN SINGLE MODE OR AT THE END OF A BLOCK OF DATA IN BLOCK MODE, THE MONITOR SAVES THE CURRENT STATE OF THE MACHINE (THE AC'S, THE APR FLAGS, THE PROTECTION-RELOCATION REGISTER, THE UUO TRAP ADDRESSES 40 AND 41, AND THE RESERVED INSTRUCTION TRAP ADDRESSES 60 AND 61), AND THEN SETS UP THE NEW PROTECTION-RELOCATION REGISTER AND APR FLAGS AND TRAPS TO THE USER INTERRUPT ROUTINE. AFTER SERVICING HIS DEVICE THE USER RETURNS CONTROL TO THE MONITOR TO RESTORE THE PREVIOUS STATE OF THE MACHINE AND DISMISS THE INTERRUPT.

IN FAST BLOCK MODE THE MONITOR PLACES THE BLKI/O INSTRUCTION DIRECTLY IN THE PI LOCATION. THIS REQUIRES THAT THE PI CHANNEL BE DEDICATED TO THE REAL TIME JOB DURING ANY DATA TRANSFERS. IN NORMAL BLOCK MODE THE MONITOR PLACES THE BLKI/O INSTRUCTION DIRECTLY AFTER THE REAL TIME DEVICE'S CONSO INSTRUCTION IN THE CONSO SKIP CHAIN. ANY NUMBER OF REAL TIME DEVICES USING EITHER SINGLE MODE OR NORMAL BLOCK MODE CAN BE PLACED ON ANY AVAILABLE PI LEVEL. THE AVERAGE EXTRA OVERHEAD PER REAL TIME DEVICE ON THE SAME CHANNEL IS 5.5 US PER INTERRUPT.

THE REAL TIME TRAP UUO CAN BE CALLED FROM UUO LEVEL OR FROM INTERRUPT LEVEL. IT CAN BE USED TO ADD DEVICES TO THE INTERRUPT SYSTEM, TO REMOVE THEM FROM THE INTERRUPT SYSTEM, OR TO CHANGE THEIR PI LEVEL ASSIGNMENT. THE RTTRP UUO IS A PRIVELEGED UUO AND REQUIRES THAT THE JOB HAVE REAL TIME PRIVELEGE (GRANTED BY LOGIN) AND BE LOCKED IN CORE (ACCOMPLISHED BY THE LOCK UUO).

A) RTTRP UUO FORMAT:

```
        MOVEI AC,RTBLK          ;AC CONTAINS ADDRESS OF DATA BLOCK
        CALLI  AC,57            ;PUT DEVICE ON PI LEVEL
        ERROR RETURN            ;RETURN HERE IF AN ERROR OCCURED
                                ;ERROR CODE IS RETURNED IN AC
        NORMAL RETURN           ;RETURN HERE IF PI SET UP PROPERLY

        ;SINGLE MODE
RTBLK:  XWD  PICHL,TRPADR       ;PI CHANNEL (1-6) AND TRAP ADDRESS
        EXP      APRTRP         ;APR ENABLE BITS AND APR TRAP ADR
        CONSO DEV,BITS          ;CONSO CHAIN INSTRUCTION
        0                       ;NO BLKI/O INSTRUCTION

                OR

        ;FAST BLOCK MODE
RTBLK:  XWD  PICHL,TRPADR       ;PI   AND TRAP ADR WHEN BLKO DONE
        EXP      APRTRP         ;APR TRAP CONDITIONS
        BLKO DEV,BLKADR         ;BLKO OR BLKI INSTRUCTION
        0                       ;BLKADR POINTS TO THE IOWD OF
                                ;BLOCK TO BE SENT OUT.

                OR
        ;NORMAL BLOCK MODE
RTBLK:  XWD      PICHL,TRPADR   ;CHANNEL   AND TRAP ADDRESS
        EXP      APRTRP         ;APR TRAP ADDRESS
        CONSO    DEV,@BITMSK    ;CONTROL BIT MASK FROM USER AREA
        BLKI     DEV,BLKADR     ;BLKI INSTRUCTION
```

B) DISMISSING PROCEDURE:

ANY UUO OTHER THAN THE RTTRP UUO WILL DISMISS THE INTERRUP ANY INSTRUCTION WHICH TRAPS TO ABSOLUTE LOCATION 60 WI DISMISS THE INTERRUPT VERY QUICKLY. THE STANDARD METHOD DISMISSING AN INTERRUPT IS WITH A UJEN INSTRUCTION (OPCO 100). THIS TRAPS TO LOCATION 60 GIVING THE FASTEST POSSIB DISMISSAL. THE USE OF OPCODE 100 IS DESIRABLE TO GUA AGAINST PROGRAM OBSOLESCENCE DUE TO HARDWARE EXPANSION.

C) INTERRUPT LEVEL USE OF RTIRP:

THE RTTRP UUO FORMAT AT INTERRUPT LEVEL IS THE SAME AS AT USER LEVEL EXCEPT THAT AC'S 16 AND 17 CANNOT BE USED IN THE UUO CALL. (I.E. CALLI 16,57 IS ILLEGAL AT INTERRUPT LEVEL) ALSO, EXCECUTION OF THE RTTRP UUO AT INTERRUPT LEVEL CAUSES ALL AC'S TO BE OVERWRITTEN. THUS THE USER MUST SAVE ANY DESIRED AC'S BEFORE ISSUING THE RTTRP UUO AT INTERRUPT LEVEL. THIS IS TO SAVE TIME AT INTERRUPT LEVEL.

CAUTION: IF AN INTERRUPT LEVEL ROUTINE EXECUTES AN RTIRP UUO WHICH AFFECTS THE DEVICE CURRENTLY BEING SERVICED (FOR INSTANCE TO CHANGE FROM A BLKI TO A BLKO), THEN NO MORE UUO'S OF ANY KIND, RTTRP INCLUDED, CAN BE EXECUTED. AT THIS POINT ANY SUBSEQUENT UUO WILL DISMISS THE INTERRUPT.

D) REMOVING DEVICES FROM A PI CHANNEL:

IF PICHL IN THE RTBLK IS ZERO (SEE UUO FORMAT DESCRIPTION ABOVE), THEN THE DEVICE SPECIFIED IN THE CONSO INSTRUCTION IS REMOVED FROM THE INTERRUPT SYSTEM.

A RESET UUO FROM TIME SHARING LEVEL WILL REMOVE ALL DEVICES FROM THE INTERRUPT LEVELS. THIS UUO CAUSES A "CONO DEV,0" TO BE EXECUTED BEFORE THE DEVICE IS REMOVED.

E) ERROR RETURN CODES:

BIT 35	1	PI CHANNEL NOT AVAILABLE (RESTRICTED USE BY SYSTEM)
BIT 34	2	PI CHANNEL NOT CURRENTLY AVAILABLE FOR BLKI/O'S
BIT 33	4	TRAP ADDRESS OUT OF BOUNDS
BIT 32	10	ERROR ADDRESS OUT OF BOUNDS
BIT 31	20	BLKADR OR POINTR WORD ILLEGAL
BIT 30	40	ILLEGAL FORMAT OF CONSO, BLKO, OR BLKI INSTRUCTION
BIT 29	100	SYSTEM LIMIT FOR REAL TIME DEVICES EXCEEDED
BIT 28	200	JOB NOT LOCKED IN CORE
BIT 27	400	ILLEGAL AC USED DURING RTTRP UUO AT INTERRUPT LEVEL (AC'S 16 AND 17 ARE ILLEGAL)
BIT 26	1000	DEVICE ALREADY IN USE BY ANOTHER JOB

F) EXPLANATION OF RTTRP MNEMONICS:

1. "PICHL" IS THE PI LEVEL WHERE THE DEVICE IS TO BE PLACED. LEVELS 1-6 ARE LEGAL DEPENDING ON SYSTEM CONFIGURATION. PICHL IS 0, THE DEVICE IS REMOVED FROM ALL LEVELS. NORMAL WHEN A DEVICE IS PUT ONTO A PI LEVEL, ALL OTHER INSTANCES THE SAME DEVICE ON ANY OF THE PI LEVELS ARE REMOVED. I HOWEVER, THE USER WANTS TO PUT THE SAME DEVICE ON MORE TH ONE PI LEVEL AT THE SAME TIME (I.E. A DATA LEVEL AND ERROR LEVEL) THEN HE CAN GIVE AN RTTRP UUO WITH PIC NEGATIVE. THIS TELLS THE SYSTEM NOT TO REMOVE ANY OTH OCCURENCES OF THIS DEVICE FROM ANY OTHER (OR THE SAME) LEVELS. NOTE THAT THIS COUNTS AS ANOTHER REAL TIME DEVI TAKING ONE OF THE POSSIBLE REAL TIME DEVICE SLOTS.

2. "TRPADR" IS THE LOCATION TO WHICH A REAL TIME INTERRUPT TRA (JRST TRPADR). BEFORE A TRAP OCCURS ALL AC'S ARE SAVED THE MONITOR AND CAN THUS BE OVERWRITTEN WITHOUT CONCERN F THEIR CONTENTS.

3. "APRTRP" IS THE TRAP LOCATION FOR ALL APR TRAPS. THE MONIT SIMULATES A "JSR APRTRP" WHEN AN APR TRAP OCCURS. THE US GETS CONTROL FROM AN APR TRAP ON THE SAME PI LEVEL THAT H REAL TIME DEVICE IS ON. THE MONITOR WILL ALLWAYS TRAP TO T USER PROGRAM ON ILLEGAL MEMORY REFERENCES, NON-EX-MEM'S, A PUSH DOWN OVERFLOWS. THIS IS TO ALLOW THE USER TO PROPER TURN OFF HIS REAL TIME DEVICE IN CASE IT HAS RUN WILD. T MONITOR WILL ALSO TRAP ON THOSE CONDITIONS SPECIFIED BY T APRENB UUO. IF THE INTERRUPT ROUTINE IS ON A PI LEVEL HIGH THAN OR EQUAL TO THE APR INTERRUPT LEVEL, THEN NO APR ERRO WILL BE DETECTED.

4. "DEV" IS THE REAL TIME DEVICE CODE NUMBER.

5. "BITS" IS THE BIT MASK OF ALL INTERRUPT BITS OF THE REAL TI DEVICE THIS BIT MASK MUST CONTAIN ALL INTERRUPTING BITS A NO OTHERS. IF THE USER WANTS TO CONTROL THIS BIT MASK FR HIS USER AREA, HE MAY SPECIFY ONE LEVEL OF INDIRECTION IN T CONSO INSTRUCTION (NO INDEXING). I.E. "CONSO DEV,@MAS WHERE MASK IS A LOCATION IN THE USER AREA OF THE BIT MASK F THAT REAL TIME DEVICE. MASK MUST NOT HAVE ANY BITS IN T INDIRECT OR INDEX FIELDS SET.

6. "BLKADR" IS THE ADDRESS IN THE USER'S AREA OF THE BLKI/O POINTER WORD. THE MONITOR WILL ADD THE PROPER RELOCATION FACTOR TO THE RIGHT HALF OF THIS POINTER WORD BEFORE RETURNING TO THE USER. DATA CAN BE READ ONLY INTO THE LOW SEGMENT AND ONLY ABOVE THE PROTECTED JOB DATA AREA (LOC. 20-114). THE POINTER WORD WAS LEFT IN THE USER'S AREA TO ALLOW HIM TO SET UP A NEW POINTER WORD VERY QUICKLY WHEN THE WORD COUNT GOES TO Ø AT INTERRUPT LEVEL. THUS WHEN THE USER WISHES TO SET UP HIS OWN POINTER WORD, HE MUST MAKE THE ADDRESS IN THE RIGHT HALF OF THE WORD BE AN ABSOLUTE ADDRESS INSTEAD OF A RELATIVE ADDRESS. THE RELOCATION VALUE IS RETURNED FROM BOTH THE LOCK UUO AND FROM THE FIRST RTTRP UUO EXECUTED TO SET UP THE BLKI/O INSTRUCTION. FAILURE TO ADD THE PROPER RELOCATION VALUE TO THE RELATIVE ADDRESS WILL UNDOUBTEDLY CAUSE THE MONITOR TO BE OVERWRITTEN. THE NEGATIVE WORD COUNT IN THE LEFT HALF OF THE POINTER WORD SHOULD BE CHECKED AGAINST BEING TOO LARGE. IF THE WORD COUNT EXTENDS BEYOND THE USER'S OWN AREA THEN THE DEVICE COULD OVERWRITE ANOTHER JOB. IF THESE PRECAUTIONS ARE OBSERVED, THIS METHOD OF SETTING UP THE POINTER WORD IS MUCH FASTER AND MORE FLEXIBLE THAN GIVING AN RTTRP UUO AT INTERRUPT LEVEL. (SEE EXAMPLES 2 AND 3)

7. BEFORE GIVING AN ERROR RETURN TO THE USER, RTTRP WILL SCAN THE ENTIRE DATA BLOCK TO FIND AS MANY ERRORS AS POSSIBLE.

8. RTTRP RETURNS TO THE USER WITH THE USER IOT PRIVILEGE ENABLED SO THAT THE USER CAN EXECUTE ALL I/O INSTRUCTIONS, AND ALL OTHER RESTRICTED INSTRUCTIONS.

9. A USER CAN OBTAIN JUST THE USER IOT PRIVILEGE BY ISSUEING AN RTTRP UUO WITH PICHL=Ø.

G) RULES OF OPERATION:

1. DEVICES CAN BE CHAINED ON TO ANY PI CHANNEL NOT USED FOR BLKI/O INSTRUCTIONS BY THE SYSTEM OR BY OTHER REAL-TIME USERS USING FAST BLOCK MODE (THIS INCLUDES THE APR CHANNEL). HOWEVER, PI LEVEL 7 IS ALWAYS RESERVED FOR THE SYSTEM. NORMALLY PI LEVELS 1 AND 2 ARE RESERVED BY THE SYSTEM FOR DECTAPES AND MAGTAPES.

2. BLKI OR BLKO INSTRUCTIONS CAN BE PUT IN THE PI LOCATIONS ON ANY UNUSED PI LEVEL (FAST BLOCK MODE), THIS DEDICATES THAT PI LEVEL TO THE REAL TIME JOB FOR THE DURATION OF HIS DATA TRANSFER.

3. JOBS DESIRING THE USE OF THIS REAL-TIME FEATURE MUST BE LOCKED INTO CORE (ACCOMPLISHED BY USING THE LOCK UUO).

4. JOBS WHICH HAVE SUCCESSFULLY EXECUTED A RTTRP UUO ARE GIVEN USER IOT PRIVILEGE. THIS ALLOWS THE USER TO EXECUTE THE NECESSARY I/O INSTRUCTIONS TO CONTROL HIS DEVICE. IOT PRIVILEGE MUST BE USED WITH CAUTION. IMPROPER USE OF I/O INSTRUCTIONS COULD HANG THE SYSTEM (I.E., CONO APR,0, CONO PI,0 OR HALT).

5. A USER PROGRAM MUST ALWAYS DISMISS AN INTERRUPT WITH ANY OF THE DISMISS UUO'S (ANY UUO OTHER THAN THE REAL TIME UUO'S OR ANY INSTRUCTION WHICH TRAPS TO ABSOLUTE LOCATION 60). THIS ALLOWS THE MONITOR TO PROPERLY RESTORE THE STATE OF THE MACHINE.

6. A REAL TIME DEVICE MUST BE CHAINED ON TO A PI LEVEL BY A RTTRP UUO BEFORE THE USER PROGRAM ISSUES THE CONO DEV,PIA TO THE DEVICE. FAILURE TO OBSERVE THIS RULE OR FAILURE TO ASSIGN THE DEVICE TO THE SAME PI LEVEL THAT WAS SPECIFIED IN THE RTTRP UUO COULD HANG THE SYSTEM.

7. THE USER MUST BE AWARE THAT IF THE CONSO BIT MASK IS SET UP BUT THE DEVICE HAS NOT PHYSICALLY BEEN PUT ON ITS PROPER PI LEVEL (CONO DEV,PIA), AND IF ONE OF THE CORRESPONDING FLAGS IN THE DEVICE IS ON, THEN A TRAP MAY OCCUR TO THE USER INTERRUPT SERVICE ROUTINE. THE REASON FOR THIS IS AS FOLLOWS: THERE IS A CONSO SKIP CHAIN FOR EACH PI LEVEL, AND IF A DEVICE INTERRUPTS WHOSE CONSO INSTRUCTION IS FURTHER DOWN THE THE CHAIN THAN THE REAL TIME DEVICE, THE CONSO ASSOCIATED WITH THE REAL TIME DEVICE IS EXECUTED. IF ONE OF THE HARDWARE DEVICE FLAGS IS SET, AND THE CORRESPONDING BIT IN THE CONSO BIT MASK IS ALSO SET, THEN THE CONSO WILL SKIP AND A TRAP WILL OCCUR TO THE USER PROGRAM EVEN THOUGH THE REAL TIME DEVICE WAS NOT CAUSING AN INTERRUPT ON THAT CHANNEL. TO AVOID THIS SITUATION, THE USER CAN KEEP THE CONSO BIT MASK IN HIS USER AREA BY USING A CONSO INSTRUCTION WITH THE INDIRECT BIT SET IN THE RTTRP UUO, I.E., CONSO DEV,@BITMSK, WHERE BITMSK IS THE LOCATION IN THE USER PROGRAM OF THE CONSO BIT MASK. THIS FEATURE MAKES IT POSSIBLE TO CHAIN A DEVICE ONTO THE INTERRUPT LEVEL, KEEPING THE CONSO BIT MASK ZERO UNTIL THE DEVICE IS ACTUALLY PUT ON THE PROPER PI LEVEL WITH A CONO INSTRUCTION. (SEE EXAMPLE 1) THIS PROBLEM WILL NEVER ARISE IF THE DEVICE FLAGS ARE TURNED OFF UNTIL THE CONO DEV,PIA CAN BE EXECUTED. (SEE EXAMPLE 3)

8. IF THE USER REMOVES A DEVICE FROM A PI CHAIN, HE MUST MAKE SURE THAT THE DEVICE HAS ALSO BEEN REMOVED FROM THAT PI LEVEL (I.E., CONO DEV,0).

9. THE USER SHOULD NOT PUT PROGRAMS WHICH EXECUTE FOR A LONG PERIOD OF TIME ON A HIGH PRIORITY INTERRUPT LEVEL. SUCH PROGRAMS COULD CAUSE OTHER REAL-TIME PROGRAMS AT LOWER LEVELS TO LOSE DATA.

10. SINCE THE USER IS RUNNING AT INTERRUPT LEVEL AND FULL CONTEXT SWITCHING IS NOT PERFORMED, THE USER PROGRAM MUST NOT CHANGE ANY LOCATIONS IN THE PROTECTED JOB DATA AREA (LOC. 20-114). THIS COULD KILL THE SYSTEM.

11. IF THE USER INTERRUPT ROUTINE ISSUES AN RTIRP UUO AT INTERRUPT LEVEL WHICH AFFECTS THE DEVICE WHOSE INTERRUPT IS BEING SERVICED, THEN NO FURTHER REAL TIME UUO'S CAN BE EXECUTED DURING THE REMAINDER OF THIS INTERRUPT. AT THIS POINT ALL UUO'S WILL DISMISS THE INTERRUPT.

12. IF THE USER IS USING THE BLKI/O FEATURE, HE MUST RESTORE THE BLKI/O POINTER WORD BEFORE DISMISSING ANY END-OF-BLOCK INTERRUPT. THIS CAN BE DONE WITH ANOTHER RTIRP UUO, OR BY DIRECTLY CHANGING THE ABSOLUTE POINTER WORD SUPPLIED BY THE FIRST RTTRP UUO. FAILURE TO RESET THE POINTER WORD COULD CAUSE THE DEVICE TO OVERWRITE ALL OF MEMORY.

H) EXAMPLES:

********** EXAMPLE 1 **********
SINGLE MODE

```
        TITLE RTSNGL - PAPER TAPE READ TEST USING CONSO CHAIN

        OPDEF UJEN [1B2]        ;DISMISS INSTRUCTION (OP CODE 100)

        RTTRP=57                ;RTTRP UUO NUMBER
        LOCK=60                 ;LOCK JOB IN CORE
        RESET=0                 ;IO RESET
        PIOFF=400               ;TURN PI SYSTEM OFF
        PION=200                ;TURN PI SYSTEM ON
        TAPE=400                ;NO MORE TAPE IN READER IF TAPE=0
        BUSY=20                 ;DEVICE IS BUSY READING
        DONE=10                 ;A CHARACTER HAS BEEN READ

PDATA:  Z                       ;LOCATION WHERE DATA IS READ INTO

PTRTST: CALLI RESET             ;RESET THE PROGRAM
        CALLI LOCK              ;LOCK THE JOB IN CORE
        JRST FAILED             ;LOCK UUO FAILED
        SETZM PTRCSO            ;MAKE SURE CONSO BITS ARE ZERO
        SETZM DONFLG            ;INITIALIZE DONE FLAG
        MOVEI RTBLK             ;GET ADDRESS OF REAL TIME DATA BLOCK
        CALLI RTTRP             ;PUT REAL TIME DEVICE ON THE PI LEVEL
        JRST FAILED             ;RTTRP UUO FAILED
        MOVEI 1,DONE            ;SET UP CONSO BIT MASK
        HLRZ 2,RTBLK            ;GET PI NUMBER FROM RTBLK
        TRO 2,BUSY              ;SET UP CONO BITS TO START TAPE GOING
        CONO PI,PIOFF           ;GUARD AGAINST ANY INTERRUPTS
        MOVEM 1,PTRCSO          ;STORE CONSO BIT MASK
        CONO PTR,(2)            ;TURN PTR ON
        CONO PI,PION            ;ALLOW INTERRUPTS AGAIN
        MOVEI 5                 ;SET UP TO SLEEP FOR 5 SECONDS
        CALLI 31                ;SLEEP
        SKIPN DONFLG            ;HAVE WE FINISHED READING THE TAPE
        JRST .-3                ;NO GO BACK TO SLEEP
        CALLI 0                 ;RESET AND UNLOCK THE JOB
        CALLI 12                ;EXIT

RTBLK:  XWD 5,TRPADR            ;PI CHANNEL AND TRAP ADDRESS
        EXP APRTRP              ;APR ERROR TRAP ADDRESS
        CONSO PTR,@PTRCSO       ;INDIRECT CONSO BIT MASK = PTRCSO
        Z                       ;NO BLKI/O INSTRUCTION
```

```
PTRCSO:  Ø                       ;CONSO BIT MASK
DONFLG:  Ø                       ;PI LEVEL TO USER LEVEL COMM.
RTBLK1:  Ø                       ;DATA BLOCK TO REMOVE PTR
         Ø                       ;FROM PI CHANNEL
         CONSO PTR,Ø
         Ø

TRPADR:  CONSO PTR,TAPE          ;END OF TAPE?
         JRST TDONE              ;YES, GO STOP JOB
         DATAI PTR,PDATA         ;READ IN DATA WORD
         UJEN                    ;DISMISS THE INTERRUPT

APRTRP:  Ø                       ;APR ERROR TRAP ADDRESS
TDONE:   MOVEI RTBLK1            ;SET UP TO REMOVE PTR
         CONO PTR,Ø              ;TAKE DEVICE OFF HARDWARE PI LEVEL
         CALLI RTTRP             ;REMOVE FROM SOFTWARE PI LEVEL
         JFCL                    ;IGNORE ERRORS
         SETOM DONFLG            ;MARK THAT READ IS OVER
         SETZM PTRCSO            ;CLEAR CONSO BIT MASK
         UJEN                    ;DISMISS THE INTERRUPT

FAILED:  TTCALL 3,[ASCIZ/RTTRP UUO FAILED!/]
         CALLI 12                ;EXIT

         END PTRTST
```

********** EXAMPLE 2 **********
FAST BLOCK MODE

```
        TITLE RTFBLK - PAPER TAPE READ TEST IN BLKI MODE

        OPDEF UJEN [1B2]          ;DISMISS INSTRUCTION (OP CODE 100)

        RTTRP=57                  ;RTTRP UUO NUMBER
        LOCK=60                   ;LOCK THE JOB IN CORE
        RESET=0                   ;RESET THE IO DEVICES
        TAPE=400                  ;NO MORE TAPE IN READER IF TAPE=0
        BUSY=20                   ;DEVICE IS BUSY READING
        DONE=10                   ;A CHARACTER HAS BEEN READ

BLKTST: CALLI RESET               ;RESET THE PROGRAM
        CALLI LOCK                ;LOCK THE JOB IN CORE
        JRST FAILED               ;LOCK UUO FAILED
        SETZM DONFLG              ;INITIALIZE DONE FLAG
        MOVEI RTBLK               ;GET ADDRESS OF REAL TIME DATA BLOCK
        CALLI RTTRP               ;PUT REAL TIME DEVICE ON THE PI LEVEL
        JRST FAILED               ;RTTRP UUO FAILED
        HLRZ 2,RTBLK              ;GET PI NUMBER FROM RTBLK
        TRO 2,BUSY                ;SET UP CONO BITS TO START TAPE GOING
        CONO PTR,(2)              ;TURN PTR ON
        MOVEI 5                   ;SET UP TO SLEEP FOR 5 SECONDS
        CALLI 31                  ;SLEEP
        SKIPN DONFLG              ;HAVE WE FINISHED READING THE TAPE
        JRST .-3                  ;NO GO BACK TO SLEEP
        CALLI 0                   ;RESET AND UNLOCK THE JOB
        CALLI 12                  ;EXIT

RTBLK:  XWD 6,TRPADR              ;PI CHANNEL AND TRAP ADDRESS
        EXP APRTRP                ;APR ERROR TRAP ADDRESS
        BLKI PTR,POINTR           ;READ A BLOCK AT A TIME
        Z

POINTR: IOWD 5,TABLE              ;POINTER FOR BLKI INSTRUCTION
OPOINT: IOWD 5,TABLE              ;ORIGINAL POINTER WORD FOR BLKI
TABLE:  BLOCK 5                   ;TABLE AREA FOR DATA BEING READ
DONFLG: Z                         ;PI LEVEL TO USER LEVEL COMM.
RTBLK1: Z                         ;DATA BLOCK TO REMOVE PTR
        Z                         ;FROM PI CHANNEL
        CONSO PTR,0
        Z
```

```
TRPADR:  CONSO PTR,TAPE          ;END OF TAPE?
         JRST TDONE              ;YES, GO STOP JOB
         MOVE OPOINT             ;GET ORIGINAL POINTER WORD
         MOVEM POINTR            ;RESTORE BLKI POINTER WORD
         UJEN                    ;DISMISS THE INTERRUPT

APRTRP:  Z                       ;APR ERROR TRAP ADDRESS
TDONE:   MOVEI RTBLK1            ;SET UP TO REMOVE PTR
         CONO PTR,0              ;TAKE DEVICE OFF HARDWARE PI LEVEL
         CALLI RTTRP             ;REMOVE FROM SOFTWARE PI LEVEL
         JFCL                    ;IGNORE ERRORS
         SETOM DONFLG            ;MARK THAT READ IS OVER
         UJEN                    ;DISMISS THE INTERRUPT

FAILED:  TTCALL 3,[ASCIZ/RTTRP UUO FAILED!/]
         CALLI 12                ;EXIT

         END BLKTST
```

********** EXAMPLE 3 **********
NORMAL BLOCK MODE

```
        TITLE RTNBLK - PAPER TAPE READ TEST IN BLKI MODE

        OPDEF UJEN [1B2]        ;DISMISS INSTRUCTION (OP CODE 100)

        RTTRP=57                ;RTTRP UUO NUMBER
        LOCK=60                 ;LOCK THE JOB IN CORE
        RESET=0                 ;IO RESET
        TAPE=400                ;NO MORE TAPE IN READER IF TAPE=0
        BUSY=20                 ;DEVICE IS BUSY READING
        DONE=10                 ;A CHARACTER HAS BEEN READ

BLKTST: CALLI RESET             ;IO RESET
        CALLI LOCK              ;LOCK THE JOB IN CORE
        JRST FAILED             ;LOCK UUO FAILED
        MOVEI RTBLK1            ;GET ADDRESS OF REAL TIME BLOCK
        CALLI RTTRP             ;GET USER IOT PRIVILEGE
        JRST FAILED             ;UUO FAILED!
        CONO PTR,0              ;CLEAR ALL PTR FLAGS
        SETZM DONFLG            ;INITIALIZE DONE FLAG
        MOVEI RTBLK             ;GET ADDRESS OF REAL TIME DATA BLOCK
        CALLI RTTRP             ;PUT REAL TIME DEVICE ON THE PI LEVEL
        JRST FAILED             ;RTTRP UUO FAILED
        MOVE POINTR             ;GET RELOCATED POINTER WORD FOR LATER
        MOVEM OPOINT            ;STORE FOR INTERRUPT LEVEL USE
        HLRZ 2,RTBLK            ;GET PI NUMBER FROM RTBLK
        TRO 2,BUSY              ;SET UP CONO BITS TO START TAPE GOING
        CONO PTR,(2)            ;TURN PTR ON
        MOVEI 5                 ;SET UP TO SLEEP FOR 5 SECONDS
        CALLI 31                ;SLEEP
        SKIPN DONFLG            ;HAVE WE FINISHED READING THE TAPE
        JRST .-3                ;NO GO BACK TO SLEEP
        CALLI 0                 ;RESET AND UNLOCK JOB
        CALLI 12                ;EXIT

RTBLK:  XWD 6,TRPADR            ;PI CHANNEL AND TRAP ADDRESS
        EXP APRTRP              ;APR ERROR TRAP ADDRESS
        CONSO PTR,DONE          ;WAIT ONLY FOR DONE FLAG
        BLKI PTR,POINTR         ;READ A BLOCK AT A TIME

POINTR: IOWD 5,TABLE            ;POINTER FOR BLKI INSTRUCTION
OPOINT: Z
TABLE:  BLOCK 5                 ;TABLE AREA FOR DATA BEING READ
DONFLG: Z                       ;PI LEVEL TO USER LEVEL COMM.
RTBLK1: Z                       ;DATA BLOCK TO REMOVE PTR
        Z                       ;FROM PI CHANNEL
        CONSO PTR,0
        Z
```

```
TRPADR:  CONSO PTR,TAPE          ;END OF TAPE?
         JRST TDONE              ;YES, GO STOP JOB
         MOVE OPCINT             ;GET ORIGINAL POINTER WORD
         MOVEM POINTR            ;STORE IN POINTER LOCATION
         UJEN                    ;DISMISS THE INTERRUPT

APRTRP:  Z                       ;APR ERROR TRAP ADDRESS
TDONE:   MOVEI RTBLK1            ;SET UP TO REMOVE PTR
         CONO PTR,0              ;TAKE DEVICE OFF HARDWARE PI LEVEL
         CALLI RTTRP             ;REMOVE FROM SOFTWARE PI LEVEL
         JFCL                    ;IGNORE ERRORS
         SETOM DONFLG            ;MARK THAT READ IS OVER
         UJEN                    ;DISMISS THE INTERRUPT

FAILED:  TTCALL 3,[ASCIZ/RTTRP UUO FAILED!/]
         CALLI 12                ;EXIT

         END BLKTST
```

I) FORTRAN USAGE OF THE REAL TIME TRAPPING FEATURE

FORTRAN PROGRAMS CAN CONNECT REAL TIME DEVICES TO THE PRIORITY INTERRUPT SYSTEM BY USING THE REAL TIME LIBRARY SUBROUTINES. THESE SUBROUTINES ENABLE THE FORTRAN PROGRAMMER TO WRITE REAL TIME CODE WITHOUT BEING REQUIRED TO LEARN ASSEMBLY LANGUAGE CODING OR HARDWARE CHARACTERISTICS. A LIST OF THE REAL TIME SUBROUTINES AND THEIR FUNCTIONS APPEARS BELOW. AT PRESENT THESE SUBROUTINES ARE NOT REENTRANT AND CANNOT BE CALLED FROM TWO PI LEVELS SIMULTANEOUSLY. IF THIS IS A REQUIREMENT, THEN A DIFFERENT ROUTINE MUST BE CALLED AT EACH LEVEL.

1, RTINIT (UNIT, DEV, PI, TRPADR, MASK)

UNIT - REAL TIME DEVICE UNIT NUMBER
DEV - DEVICE CODE FOR REAL TIME DEVICE
PI - PI LEVEL ON WHICH THE REAL TIME DEVICE IS TO BE RUN
TRPADR - ADDRESS IN FORTRAN PROGRAM WHERE REAL TIME INTERRUPTS ARE TO TRAP. (LOADED BY "ASSIGN 100 TO TRPADR")
MASK - MASK OF ALL INTERRUPTING FLAGS FOR THE REAL TIME DEVICE. THIS IS SET UP BY "RTSTRT" AND SHOULD BE ZERO WHENEVER THE REAL TIME DEVICE IS INACTIVE.

RTINIT INITIALIZES ALL OF THE INTERNAL TABLES CONTROLING THIS REAL TIME DEVICE.

2, CONECT (UNIT, MODE)

MODE - -1 = WRITE A BLOCK OF DATA THEN INTERRUPT
0 = INTERRUPT EVERY WORD
+1 = READ A BLOCK OF DATA THEN INTERRUPT

CONECT TELLS THE SYSTEM TO CONNECT THIS REAL TIME DEVICE TO THE PROPER PI LEVEL AND TO SET UP THE TRAP ADDRESS FOR TRAPPING.

3, DISCON (UNIT)

DISCON DISCONNECTS THE REAL TIME DEVICE FROM THE PI LEVEL.

4. RTSTRT (UNIT, START, INTMSK)

START - FLAGS NECESSARY TO START THE DEVICE
"CONO DEV, START"
INTMSK - MASK OF ALL INTERRUPTING BITS. THESE BITS ARE
LOADED INTO "MASK".

RTSTRT STARTS THE REAL TIME DEVICE WITH A "CONO DEV, START". RTSTRT CAN ALSO BE USED TO STOP THE REAL TIME DEVICE AND ZERO THE CONSO MASK I.E. "CALL RTSTRT(UNIT,0,0)". THIS IS THE PREFERED METHOD OF STOPPING THE DEVICE.

5. BLKRW (UNIT, COUNT, ADR)

COUNT - NUMBER OF WORDS TO BE READ OR WRITTEN
ADR - ARRAT INTO WHICH THE DATA IS COMING DEPENDING ON
THE "MODE" SETTING.

BLKRW SETS UP THE SIZE OF THE BLOCK AND THE STARTING ADDRESS OF THE BLOCK OF DATA. AFTER THE SPECIFIED NUMBER OF WORDS ARE READ OR WRITTEN, A TRAP OCCURS TO THE INTERRUPT TRAP ROUTINE. A NEW COUNT AND STARTING ADDRESS MUST BE SET UP EACH TIME THE PRESENT ONE RUNS OUT.

6. RTREAD (UNIT, ADR)

ADR - ADDRESS OF WHERE TO STORE THE DATA READ.

RTREAD READS A SINGLE WORD OF DATA FROM THE REAL TIME DEVICE (DATAI DEV, ADR).

7. RTWRIT (UNIT, ADR)

ADR - LOCATION OF DATA WORD TO BE SENT TO THE REAL TIME DEVICE.

RTWRIT SENDS A SINGLE WORD TO THE REAL TIME DEVICE (DATAO DEV, ADR).

1. STATO (UNIT, ADR)

ADR - LOCATION OF THE STATUS BITS TO BE SENT TO THE REAL TIME
DEVICE (CONO DEV, @ADR).

STATO SENDS THE ARGUMENT TO THE STATUS REGISTER OF THE DEVICE.

9. STATI (UNIT, ADR)

ADR - LOCATION INTO WHICH THE DEVICE STATUS BITS ARE TO BE READ.

STATI READS THE CURRENT DEVICE STATUS BITS INTO ADR (CONI DEV, ADR).

10. RTSLP (TIME)

TIME - SLEEP INTERVAL IN SECONDS.

RTSLP IS CALLED FROM THE TIME SHARING LEVEL AND WILL CAUSE THE JOB TO SLEEP UNTIL RTWAKE IS CALLED FROM INTERRUPT LEVEL. THE BACKGROUND PORTION OF THE PROGRAM GOES TO SLEEP FOR THE SPECIFIED NUMBER OF SECONDS (UP TO 60). WHEN IT WAKES UP IT CHECKS TO SEE IF RTWAKE HAS BEEN AT INTERRUPT LEVEL. IF RTWAKE HAS BEEN CALLED THEN RTSLP RETURNS TO THE CALLING PROGRAM. IF RTWAKE HAS NOT BEEN CALLED THEN THE BACKGROUND JOB GOES BACK TO SLEEP AGAIN.

11. RTWAKE

RTWAKE IS CALLED AT INTERRUPT LEVEL TO WAKE UP THE BACKGROUND PART OF THE FORTRAN PROGRAM.

12. LOCK

LOCK LOCKS THE JOB IN CORE. LOCK MUST BE CALLED BEFORE ANY OTHER ROUTINES CAN BE EXECUTED.

```
          ********** EXAMPLE 4 **********
             FORTRAN REAL TIME PROGRAM


          IMPLICIT INTEGER (A-Z)

C         LOCK THE JOB IN CORE
          CALL LOCK

          ASSIGN 20 TO TRPADR
          UNIT=1
          PI=5
          DEV="104
          MASK=0

C         INITIALIZE PTR FOR PI LEVEL 5, AND TRAP ADDRESS OF 20
          CALL RTINIT(UNIT,DEV,PI,TRPADR,MASK)

          MODE=0

C         CONNECT THE DEVICE TO PI LEVEL 5 IN SINGLE MODE
          CALL CONECT(UNIT,MODE)

          CONO="25
          CONSO="10

C         START THE PTR READING
          CALL RTSTRT(UNIT,CONO,CONSO)

C         SLEEP UNTIL THE PAPER TAPE RUNS OUT
          CALL RTSLP(5)
          CALL EXIT

C         REAL TIME TRAP SECTION
20        CALL STATI(UNIT,J)
          IF (J-"400,GE.0)GO TO 21

C         PAPER TAPE IS FINISHED, WAKE UP BACKGROUND JOB
          CALL RTSTRT(UNIT,0,0)
          CALL DISCON(UNIT)
          CALL RTWAKE
          CALL DISMIS

C         READ NEXT WORD AND DISMISS
21        CALL RTREAD(UNIT,X)
          CALL DISMIS

          END
```

J) MISCELLANEOUS INFORMATION

1. RTTRP PRIVELEGES

IN ORDER TO EXECUTE AN RTTRP UUO THE USER MUST BE A PRIVELEGED USER. THESE PRIVELEGES ARE ASSIGNED BY THE SYSTEM MANAGER AND ARE OBTAINED BY THE MONITOR FROM ACCT.SYS. THE TWO PRIVELEGE BITS WHICH ARE REQUIRED ARE:
PVLOCK - THIS ALLOWS THE JOB TO BE LOCKED IN CORE.
PVRTT - THIS ALLOWS THE RTTRP UUO TO BE EXECUTED.

2. RTTRP ALGORITHMS

A) RTTRP CHAINS EACH SINGLE MODE DEVICE ONTO THE FRONT OF THE MONITOR CONSO SKIP CHAIN. THEREFORE THE LAST DEVICE ASSIGNED TO A PI LEVEL GETS THE FASTEST RESPONSE.

B) IN NORMAL BLOCK MODE THE BLKI OR BLKO IS PUT ON THE SKIP CHAIN IN THE FOLLOWING MANNER:

```
CONSO DEV,BITS
JRST NXTDEV
BLKI DEV,POINTR
JRST <CONTEXT SWITCHER>
JEN @CH'PI
```

C) IN THE FAST BLOCK MODE THE BLKI OR BLKO IS PLACED DIRECTLY IN THE PI TRAP LOCATION, FOLLOWED BY A JSR TO THE CONTEXT SWITCHER.

3. CORE REQUIREMENTS

A) IF FTRTTRP = 0, NO EXTRA CORE.

B) IF FTRTTRP = -1, AND NUMBER OF REAL TIME DEVICES = 0
THEN THE EXTRA CORE = 6 LOCATIONS.

C) IF FTRTTRP = -1, AND NUMBER OF REAL TIME DEVICES > 0 THEN THE EXTRA CORE (IN DECIMAL WORDS) IS AS FOLLOWS:
500 WORDS FOR RTTRP
24 WORDS FOR TABLE SPACE
20 WORDS FOR EACH REAL TIME DEVICE
1 WORD FOR EACH POSSIBLE JOB ON THE SYSTEM

FOR A 40 JOB SYSTEM WITH 5 REAL TIME DEVICES, THE ADDITIONAL CORE NEEDED IS:
500+24+20*5+1*40 = 664, WORDS

4. MONITOR BUILDING CONSIDERATIONS

A) RTTRP.MAC MUST BE ASSEMBLED WITH S.MAC.

B) MONGEN ASKS: "HOW MANY REAL TIME DEVICES?"
THE ANSWER TO THIS QUESTION IS THE MAXIMUM DECIMAL NUMBER OF REAL TIME DEVICES WHICH ARE TO BE RUN ON THE SYSTEM SIMULTANEOUSLY.

APPENDIX II

PDP-10 SYSTEM PERFORMANCE

- Number of Priority Interrupt Levels - 7 std, expandable to 128
- Number of Memory Ports - 4 std, expandable
- Memory Bus Bandwidth - 1 MH_Z word rate (max.)
- I/O Bus Bandwidth - 200KH_Z word rate (max.)
- Data Channel Bus Bandwidth - 1 MH_Z word rate (max.)
- All Data Busses are Word Parallel

Response Times	Min.	Max.
. Interrupt Driven Code (with context switching)	2.3μs	100μs
. High Priority Scheduled Jobs	0.5ms	20ms
. Normally Scheduled Jobs		
- Interactive Time-Sharing Jobs	0.5ms	10 sec
- Batch Jobs	0.5ms	20 sec

- Time to save general registers

2 saved	4.9 μs
16 saved	37.1 μs

- Time to service an interrupt using BLKI/O instructions
  a. one interrupt on priority level - 6.4μs
  b. multiple interrupts per priority level - 9.2μs+NDEV * 5.5μs where NDEV = the number of Real Time devices on that PI channel

APPENDIX III

Supplementary Documentation List

Memos

. Guide For 5 Series Monitor	-100-221-024-02
. Design Specification - PDP-1Ø Software Level D and Level E	-100-221-016-02
. Multi-Program Batch Functional Spec	-100-490-012-02
. Program Logic Manual	

Handbooks

- PDP-1Ø Timesharing Handbook
- PDP-1Ø Reference Handbook

Manuals

. PDP-1Ø System Site Preparation Guide	DEC-10-HAAB-D
. System Manager's Guide	DEC-10-NWZA-D(L)
. Science Library and Fortran Utility Subprogram	DEC-10-SFLE-D

APPENDIX IV

PDP-1Ø Real-Time Monitors and System Programs

Real-Time Monitors	Designation	Typical Size
. Multiprogramming - no disk	10/40N	9K
. Multiprogramming - disk system	10/40D	14K
. Multiprogramming and swapping	10/50	20K-8 jobs 26K-32 jobs

System Programs	Shareable	Non-Sharable
Fortran, re-entrant	9K	1K
Macro 1Ø, re-entrant	5K	1K
Basic, re-entrant	5K	3K
Teco, re-entrant	2K	1K
Editor, re-entrant	1K	1K
PIP, re-entrant	3K	1K
Batch, non-reentrant	0K	4K
DDT, non-reentrant	0K	2K