DECsystem-10 in the Real-Time Environment

The term "real-time", as used to describe requirements placed on a computer system, signifies that the responsiveness of the system must be sufficiently short so as to satisfy some time-related requirement. To some, a system that supports keyboard inquiry stations is a real-time system. While this and similar system application areas are important, the balance of this paper is devoted to describing DECsystem-10 capabilities which meet the needs of real-time applications requiring millisecond and even microsecond response.

In the early 1960's computers were available for use in systems dedicated to real-time applications such as hybrid simulation, data acquisition, control, graphics display, and so on. The system designer had to design and implement whatever software he deemed necessary.

As manufacturers developed operating systems for computers, the needs of the R/T user had to be considered. To that end, the "foreground-background" system emerged. A product of the middle-to-late 1960's, the foreground-background system provides a means to utilize excess computer capability while real-time requirements are light, while devoting the full capability of the system to the real-time requirements upon demand.

In those days the classic R/T system consisted of a fast central processor with tape, disk, card input, and R/T hardware. The operating system divided the memory space, basically, into two parts: foreground, containing the resident operating system together with space for R/T tasks or jobs; and background space to be used by various non-R/T jobs to be input from cards or tape. Normally, hardware was provided to erect a "fence" between the two so that background jobs containing errors could not address the foreground area and thereby "crash" the system.

While a considerable advance in technology at the time, the foreground-background is now fairly obsolete and is quite short on capabilities found desirable--sometimes mandatory--by today's system designer.

A typical R/T system of today (and tomorrow) must support data acquisition, communications with other computers, terminal interaction, on-line debugging, on-line program development, on-line data display, hardware interrupt driven tasks, software scheduled processing of jobs, file security, and so forth.

Digital Equipment Corporation, the world's leading innovator and supplier of mini-computers and related products, has undoubtedly delivered more computers dedicated to R/T use than any other company.

The DECsystem-10, largest of the company's computing systems, offers the R/T capabilities needed to meet the critical requirements of today's system designer.

DECsystem-10 is the name under which the family of PDP-10 computers are marketed, including single and multi-processor systems of large capacity, providing both hardware and software support of mini-computers, communications, data acquisition, batch processing, and time sharing.

DECsystem-10 R/T Hardware

To meet R/T computing needs, DECsystem-10 provides a powerful KI10 central processor with paging memory management hardware, multiple high-speed register blocks, single and double precision floating point hardware, and dynamically assignable multiple priority interrupt systems. The instruction set is generally acknowledged to be ideal for both logical and arithmetical manipulations. The longer 36-bit word length provides extra precision so important in scientific computation, as well as direct addressing of more memory space. A full compliment of high performance peripherals including several unique data communications products completes the system.

For many R/T applications the inclusion of one or more small computers in the system configuration is desirable; and, of course, DIGITAL is the leader in this area. Usually, DIGITAL can supply a complete DECsystem-10 with a direct-to-memory interface of a mini-computer for no more than others would charge for a smaller big machine alone.

By the inclusion in the configuration of a mini, the system designer gains, as hardware benefits, the low-cost, high-performance, R/T peripherals developed (and proven over many hundreds of installations) for A/D, D/A, digital input and output.

Additionally, the mini handles interrupts from R/T hardware and executes "tasks" and "service routines" to buffer the collected data and control the R/T equipment. Because the satellite computer has some intelligent standalone capability, system checkout and calibration phases can be performed independently of and without interrupting operations of the larger "host" system.

DECsystem-10 R/T Software

In the past, R/T software requirements were met by foreground-background systems--the concept being that R/T requirements (fast response) would be met by "tasks" (small programs) loaded in the foreground space (monitor area) and executed in "master" (executive) mode. This procedural code, loaded as absolute (non-relocateable) core images, was capable of addressing all of core and executing all instructions (therefore dangerous to system integrity.)

Tasks would respond directly to hardware interrupt (not "scheduled" by software) typically for the purpose of inputting data to be processed by other programs running in background (user) mode in the background (brought in from a disk or drum.) Communication (exchange of data) between the foreground and the background would be done through a "mailbox" area (reserved block of core set up at system generation time--not dynamic.)

Today the limitations and inflexibility of this software design have become obvious to system planners. The lack of protection, the lack of a good file system, and the inability to support on-line program development and debugging are definite shortcomings. Because nearly every significant R/T system now being planned includes major requirements for data communications, batch processing, and on-line terminal usage, a more highly developed operating system is essential.

The DECsystem-10 monitor and operating system supporting time-sharing, communications, batch processing, and extensive R/T capabilities--all on one system--provides this extension to previous systems.

Summarized, the features of the DECsystem-10 real-time operating system are: fast response, high throughput, and no fixed memory partitions. Both single and dual processor systems are supported.

All demands for system resources, ranging from the most time-critical real-time task to batch processing and timesharing, are handled efficiently. Real-time programs may be written in FORTRAN or MACRO-10 (assembly language.)

In order to obtain fastest response, a job may be driven directly in response to the priority interrupt system and be run in Executive mode, providing response times limited only by the ability of the hardware to respond to interrupts (typically under 10 microseconds KA10, 3 microseconds KI10.)

Response times of 100 microseconds for the KA10 or 3 microseconds for the KI10 may be obtained by programs running in User mode by means of the real-time trap monitor call providing fast response while simultaneously offering protection to the rest of the system in the event that the R/T program is not entirely without possibility of error.

Response times of a few milliseconds or less are typical for jobs placed in any one of 15 high-priority queues maintained by the system scheduler.

Using high-priority queues, system response may be biased to favor real-time tasks, batch processing or any other program or group of programs--a very useful tool for the system administrator or batch operator.

Jobs may be run on a periodic basis under control of the system clock. Also, one job may request that another be run--a feature useful when a block of data has been input and an analysis job on the disk system is selectable. This is accomplished by means of hibernate and wake monitor calls.

REAL-TIME SOFTWARE FEATURES

Real-time programmers are able to call upon unique DECsystem-10 monitor features for the various privileges and services listed below.

- Fast Response to Interrupts--Real-time programmers may dynamically connect and disconnect their time-critical real-time equipment and tasks to the DECsystem-10 priority interrupt system.

- Response to interrupts is immediate. Real-time programs run at interrupt level in either Privileged User or Executive hardware modes. A spectrum of real-time privileges is provided to be administered by the system manager.

  The interrupt service time for each data word in fast boock mode is 6.4 microseconds and in normal block mode, 14.6 microseconds (4.4 and 8.0 microseconds KI10.) User programs can get control in 3.1 microseconds, "super" Executive mode; in less than 10 microseconds, Executive mode; and in 100 microseconds with maximum security Privileged User mode. (1.8, 3, 3 microseconds KI10.) In all cases the real-time code is completely protected from the coding errors or program bugs of timesharing and batch processing users.

- Core Management Control--The real-time programmer, by means of a monitor call, can cause his program to be "locked" into core and, thereby, to become protected against being "swapped" or rolled out onto the disk (drum.) Prior to performing the "lock", the monitor positions the real-time code so as to ensure the efficient use of core by maximizing contiguous space in the remaining core area.

- Job Priority Assignments--DECsystem-10 programmers, with privileges granted by the system administrator, can assign any of fifteen priorities to their job (s). The system may be biased to favor real-time, timesharing, and/or batch jobs of special interst. An analogous console command allows a terminal user to make these assignments. The job scheduler always scans high-priority queues first and in descending priority order.

- Jobs May Share a Common Data Area--Programs (more than one job) may share a common area in core. Thereby, a data collection or filtering program can pass data to a more extensive analysis program for processing. This technique, provided by the DECsystem-10 protection and relocation hardware, allows one to lock the smaller program in core, with the large analysis program remaining disk resident until needed.

- Jobs May go to Sleep Until Awakened--When a program is not needed until another task has been completed or until a specified length of time has passed, it may declare that it wishes to "hibernate." If a time is specified, it will

awaken when that period has elapsed. If not time is given, the job will wait indefinitely until called by another job. "Waking" a suspended job is accomplished by executing the monitor call WAKE with the job number specified. Waking is immediate if the hibernating job has previously been given highest priority.

- Real-Time Jobs are Completely Protected from Background--Exec, Privileged User, and User hardware modes, plus memory protection and relocation hardware, provide protection from coding errors and program "bugs."

- Protection Against Loss of Data Files--Redundant recording of file retrieval information ensures against loss of data files.

- Protection Against Loss of Power--Power Failsafe hardware provides system security against loss of power.

- No Fixed Memory Partitions--DECsystem-10 dual memory protection and relocation hardware provides absolute address protection and relocation of jobs throughout core memory--dynamically.

- Re-entrant Systems Software--Conserves core space in the multi-program environment. Jobs may share core space.

- Dual Processor System--Master-slave dual processor systems supported by system software.

- Unified Job Control Language--One job control language for real-time, batch and timesharing users.

- Higher Level Language Control of Real-Time Tasks--FORTRAN routines provided for real-time system usage.

- Dynamic Scheduler--Does not use fixed time slot or round robin algorithm.

FILE HANDLING

File service for disk packs, drums, and disks is designed for maximum convenience and system efficiency. Each user may have as many files as he desires on any of the files storage devices in the system. The only limit on file size is a quota, which the

installation can set for each user, or the physical capacity of the installation-defined file structure, which can include storage on several like devices. Each file is referred to by name so that the user is not required to know where his file is physically located.

For user convenience, file organization is independent of access method. Therefore, it is not necessary to reorganize a file completely to change from sequential-access to random-access methods. The user may even change his access methods during file processing. For example, he could use random-access methods to find a pointer block, and then user sequential-access for the remainder of his processing.

File storage is dynamically allocated during program operations, so there is no need to preallocate a certain number of blocks before a file is established. This feature is especially useful during program development and debugging, when the final size of the file is still unknown. However, a user is not limited to automatic allocation. If he wants to, he can reserve a contiguous area on a drum or disk pack to make sequential and random processing even more efficient. When processing is completed, he can keep his preallocated file space for future use...or return some or all of it to the public pool.

For convenience and flexibility in system design, files can be shared concurrently (even with different access methods) among specified users through the use of protection codes. These codes, which are assigned when the file is created, describe the access privileges of the person who created the file, members of the same project, and all other system users. These persons or groups may be assigned any number of privileges such as EXECUTE ONLY: READ AND EXECUTE: READ, EXECUTE, and MODIFY: or any of the groups may be completely excluded from file access.

Protection Level	Access Code	Access Privileges
Greatest Protection	7	No access privileges
	6	EXECUTE ONLY
	5	READ, EXECUTE
	4	APPEND, READ, EXECUTE
	3	UPDATE, APPEND, READ, EXECUTE
	2	WRITE, UPDATE, APPEND, READ, EXECUTE
	1	RENAME, WRITE, UPDATE, APPEND, READ, EXECUTE
Least Protection	0	CHANGE PROTECTION, RENAME, WRITE, UPDATE, APPEND, READ, EXECUTE

Files are assigned protection levels for each of three classes of users: self; users with a common project number; and all users. Each user class may be assigned a different access privilege, so that there are 21 possible combinations:

File Access Code	User Class: Self	Common Project	All Users
7			7
6			
5		5	
4			
3			
2			
1			
0	0		

The typical file system protection code is 057. This protection excludes all general users, but permits users on the same project to READ and EXECUTE on the file. The file owner has all privileges.

Updating a file is performed by either of two methods: superseding or updating in place. When a person makes a change by superseding a shared file, concurrent users are not affected until they finish using the file and later reaccess it. In situations where users need immediate access to the most current data, such as in management information systems, data bases are updated in place. In this method, concurrent users receive changes as soon as they are made.

In some applications it is necessary for a person or group of users to have complete control of a file structure, such as when file processing depends on the mounting or dismounting of disk packs. A disk drive can be designated as private and assigned to one user, so that packs can be mounted or dismounted without disturbing other users.

On the DECsystem-10, file processing is highly efficient. In sequential processing, the monitor checks to see if the next block of data requested by the user physically follows the current block; if it does, the monitor uses command chaining, a hardware feature of the data channel, and loads additional buffers scattered throughout the user's core area. The system optimizes accesses, I/O requests are queued and processed on the basis of minimum access time. To assure retrieval of data positioned farthest from the moving head, both seek and latency algorithms employ a fairness clause, which means that after a given number of retrievals, the data that has waited the longest is accessed next. The number of retrievals is an installation parameter and can be adjusted to meet specific operating conditions. When multiple moving-head disks are employed, the monitor also overlaps seeks on all devices and allows simultaneous transfers to and from core memory via separate data channels.

The DECsystem-10 monitor avoids unnecessary file accesses by maintaining retrieval information for recently active files in an in-core data base. For reliability, the retrieval information is recorded in two separate locations on the file device, reducing the probability of destroying both directory locations simultaneously.

Swapping space (random-access storage which receives core memory overflow) may be allocated on any file storage device. The monitor employs the fastest device first. When its allocated space is full, the next fastest device is employed for swapping. No special device is required for operation; the system can use disk packs as the sole storage media just as easily as it can employ a full complement of disks, drums, and disk packs.

In addition to those features of the DECsystem-10 operating system aimed directly at the support of real-time, immediate response applications, there is a wide spectrum of system software for general purpose time-sharing, terminal inquiry, data base management, simultaneous file access, etc. The DECsystem-10 Technical Summary (new edition) can provide the reader with a clear and complete description of these important facilities.

REAL-TIME SYSTEMS COMPARED

FEATURE	DECsystem-10 KA10	DECsystem-10 KI10	XDS SIGMA 8	DEC RSX-11D
1. Can R/T tasks be linked directly to hardware interrupts (hardware scheduled?)	Yes	Yes	Yes	
2. Can R/T tasks be run in either "Exec" or "User" modes?	Yes	Yes	No	
3. Can jobs (tasks) issue monitor calls to lock themselves in core, attach to interrupt level, and thereby become R/T?	Yes	Yes	No	
4. Are there separate queues for disk files so that R/T task requests for disk usage will receive priority over background (batch and T/S requests?)	Yes	Yes	No	
5. Are R/T programs (foreground) protected from background?	Yes	Yes	Yes	
6. Are background programs protected in any way from foreground program errors?	Yes	Yes	No	
7. Are R/T programs relocateable?	Yes	Yes	No	
8. Are there fixed memory partitions?	No.	No	Yes	
9. Is time sharing supported?	Yes	Yes	No	

	DECsystem-10 KA10	DECsystem-10 KI10	XDS SIGMA 8	DEC RSX-11D
10. Are there high-priority queues for software scheduling time important jobs	Yes	Yes	No	
11. Is there a highly developed scheduler, swapper for the efficient scheduling of jobs and management of memory?	Yes	Yes	No	
12. Is multi-programming supported?	Yes	Yes	No	
13. Rotational latency optimization for disk and drum transfers?	Yes	Yes	No	
14. File retrieval information protected in the event core is over-written due to software or hardware failure?	Yes	Yes	No	
15. Sophisticated file structure with				
- Tree-structured file directories	Yes	Yes	No	
- Sequential and random access of files	Yes	Yes	No	
- Multiple levels of file protection for simultaneous access by many users	Yes	Yes	No	
- Automatic disk file space allocation and expansion	Yes	Yes	No	