R.L.

SYSTEM BULLETIN/CONTROL DATA® 6400

David Lee

SYSTEM BULLETIN/CONTROL DATA® 6400

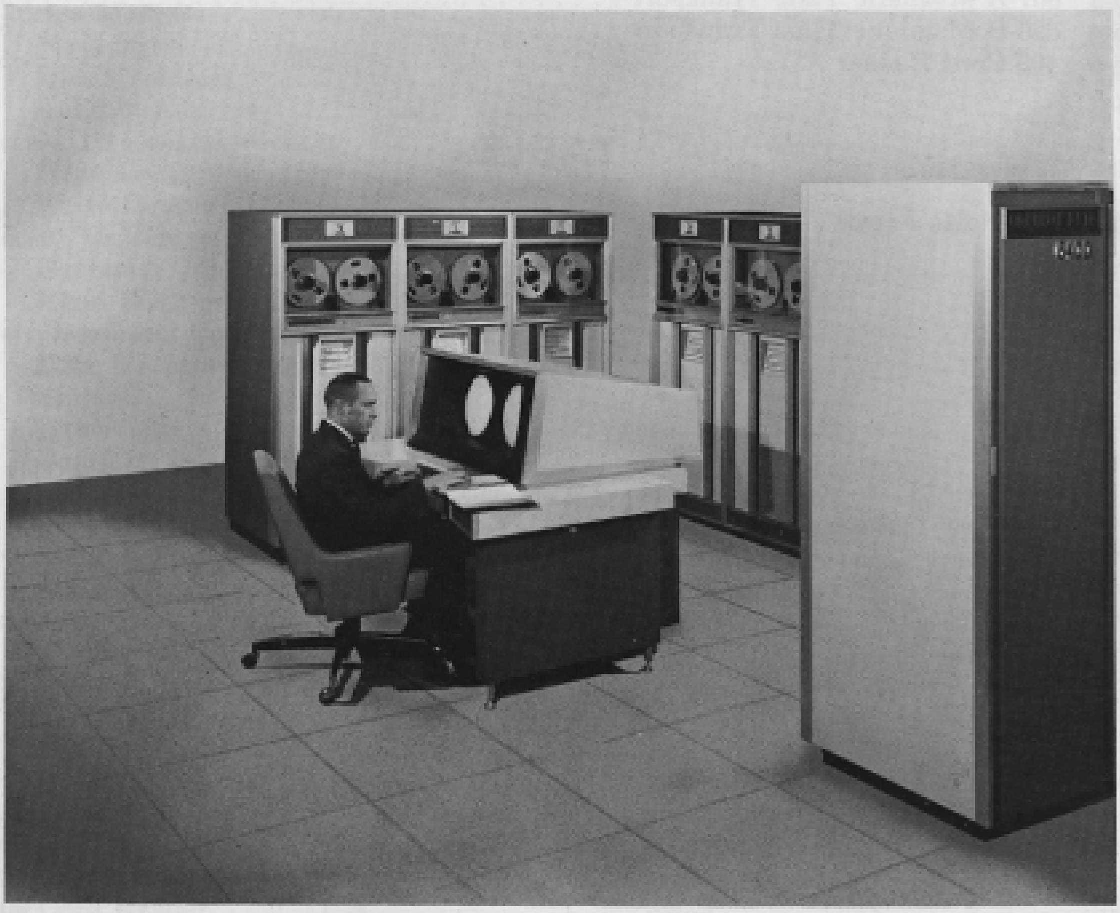
Pub. No. 60123000
December, 1964

Control Data Corporation
Technical Publications Department
4201 North Lexington
St. Paul, Minnesota 55112

CONTENTS

FIGURES

TABLE



I. INTRODUCTION

The CONTROL DATA* 6400 Computer System is a large-scale, solid-state, general purpose digital computer system designed to operate in a wide range of general and special purpose applications. It is a multiprocessing and multiprogramming computer system in the truest sense, combining multiple computers in a tight functional relationship with a fully integrated programming and operating system designed for maximal utilization of the powerful hardware configuration in simultaneous processing of multiple programs.

The 6400 Computer System is also completely compatible with the entire 6600 Computer System software.

The basic design philosophy leading to the development of the 6400 was to provide a user-oriented computation system with high computation/cost ratio that would serve a wide range of specific user's present and future needs.

In addition to handling large volumes of conventional applications, a significant design feature is its multiprogramming ability to handle multiple problems of real-time and conventional types simultaneously. The system is well adapted to handle special applications where on-line inputs and outputs with real-time calculations are involved.

SYSTEMS CONCEPTS

Two broad system objectives were defined early in the planning stages of the 6400. These objectives were:

- Machine Multiprocessing Organization
- Operational Multiprogramming Capability
- Complete Compatibility with 6000 Series hardware and software

*Registered trademark of Control Data Corporation

Multiprocessing Organization

The 6400 Computer System achieves full multiprocessing capability by the parallel integration of multiple processors, multiple memories, and multiple data channels.

Machine features include:

- 1 large Central Processor (a second Central Processor is optional)
- 1 large Central Memory
- 10 logically-independent Peripheral and Control Processors
- 12 high-speed Data Channels

Central Processor:

One of the eleven independent processors is the Central Processor. It is a very high-speed arithmetic device, called upon to solve problems requiring extremely fast arithmetic speed. The Central Processor is designed to be time-shared between several programs, all of which exist concurrently in the large Central Memory. To achieve effective multiprocessing operation, the Central Processor can be switched to alternate programs in case of input-output wait restrictions upon the program in process.

Central Memory:

Central Memory in the 6400 provides either 32K, 64K, or 128K words of memory. This large Central Memory provides the storage needed for internal multiprocessing of data. It is a bi-directional memory store, serving both the Central Processor and the Peripheral and Control Processors.

Peripheral and Control Processors:

The role of the ten Peripheral and Control Processors is to handle peripheral equipments by programmed I/O operation and to operate as the executive and monitor control in order to completely free the Central Processor for problem solving and high-speed logical and arithmetic computation. Each of these ten processors has its own separate memory and can execute programs independently of the other processors or the Central Processor.

Data Channels:

The twelve bi-directional Data Channels connect to one or more pieces of peripheral equipment. All twelve Data Channels may be simultaneously in operation, and may be communicating with any one of the Peripheral Processors, independent of the operation of the Central Processor.

Multiprogramming Capability

Multiprogramming is made possible through a functional integration of machine, operating system, and programming systems. Multiprogramming is the ability to hold more than one problem program in Central Memory at any given time. The purpose is to make more efficient use of the Central Processor by completely overlapping required input-output operations with Central Processor operation.

The 6400 Computer System achieves its multiprogramming goal by effective use of the flexible machine organization previously discussed. It is accomplished by:

- Simultaneous Processing Operating System (SIPROS)
- Programming Systems

Simultaneous Processing Operating System (SIPROS):

The Simultaneous Processing Operating System (SIPROS) is an operating system specifically tailored to make maximum use of the 6400 multiprocessor capabilities in its goal of full multiprogramming capability.

Efficient utilization of the equipment, as permitted by the hardware, to execute several jobs simultaneously in a dynamic job environment defines one of the basic objectives of SIPROS.

SIPROS is designed as a parametric system. This design methodology permits parametric control in the following critical application areas:

- Where operations of the system require buffer regions, the amounts of buffer area are controlled by parameters.
- Where the operations require peripheral devices, the units and number of units are designated by parameters.
- Where time and utilization limits are enforced, the actual values of the limits are specified by parameters.

Through selection of parameter values, the multiprocessing properties of the system are balanced to the average job mix of a user. In this manner, SIPROS provides individual consideration to the user's needs without the cost of system modification or the inconvenience of giving up compatibility with other users.

During normal operation, the operating system assumes full control of all system components and interlaces job processing on a basis of priority or data and equipment availability. By switching the execution of programs within the Central Processor during I/O delays or whenever a higher priority requirement is encountered, exceptional processing efficiency and flexibility are achieved without operator intervention and with little concern by the problem programmer for I/O optimization.

With the exception of job priority, the problem programmer may in fact be oblivious to concurrent system activities. Since the complete switching of Central Memory programs may be accomplished by execution of a single Exchange Jump instruction, the system is equally adaptable to the requirements of real-time and remote applications.

SIPROS provides complete control over activity allocation, execution, and termination of all programs. This centralized computer operating system, together with memory protection, maximizes utilization of memory and input/output devices, enabling the user to realize rapid throughput.

Programming Systems:

SIPROS provides the control under which the problem-oriented programming systems accomplish the tasks put to the 6400 by the user. These programming systems are:

- FORTRAN 66
- Assembly System, Central Processor (ASCENT)
- Assembly System, Peripheral Processors (ASPER)
- Library System of Input-Output and Utility Systems (LIBRIOUS)
- COBOL 61 Extended
- IBM 7090/94 Simulator

FORTRAN 66:

The FORTRAN 66 compiler accepts the newest and most powerful features of FORTRAN. It is a compiler system specifically designed to make maximum use of the hardware characteristics of the 6400.

Implementation of FORTRAN 66 places heavy emphasis on efficiency, both in compilation and in object code produced. Special emphasis is placed on algorithms and numerical techniques that best exploit the word size and instruction repertoire of the Central Processor, since these represent significant departures from previous machines.

Some of the features of FORTRAN 66 are:

- Language includes FORTRAN 63 and hence is compatible with both FORTRAN 63 and, essentially, with FORTRAN IV.
- Mixed Assembly language and FORTRAN statements are allowed.
- Provision is made for automatic input-output buffering.
- Register names can be used as operands.
- Object code is optimized by a variety of 6400 hardware-oriented techniques.
- Special Numerical Methods are used for certain algorithms.

ASCENT and ASPER:

Two assemblers provided with the 6400 Computer System — the Assembly System, Central Processor (ASCENT), and the Assembly System, Peripheral Processors (ASPER) — are symbolic machine-oriented languages which permit direct access to all features of the hardware, but allow the use of various functions provided by the operating system.

ASCENT allows very rapid machine language coding by using system features for control of input-output, subroutine call, and macros, yet the programmer is not restricted from using any feature of the Central Processor. Principal features of ASCENT are:

- Use of all input-output functions in the operating system.
- Macro system — use of system macros and programmer-defined macros.
- Ability to mix with FORTRAN language on a line-by-line basis.
- Subroutine call.
- Pseudo commands.
- Peripheral Processor program call.

Peripheral Processors are not normally programmed in the course of accomplishing day-to-day work; they are generally assigned the functions of the operating system and all input-output work. But in situations that require very precise machine language programming of the Peripheral Processors to obtain the desired results, ASPER provides a symbolic assembly system for the Peripheral Processors.

ASPER includes standard features such as subroutine call, pseudo commands, access to all features of the Peripheral Processors, a macro system, and access to all input-output functions provided in the operating system.

LIBRIOUS:

The Library System of Input-Output and Utility Systems (LIBRIOUS) contains a set of programs available to the user via the programming languages. The library system has the capabilities necessary for updating the library store of routines. This store is composed of routines used frequently enough to be kept on-line to the operating system, yet not frequently enough to be contained in the resident operating system. Library routines are of the following types:

- Input-Output Routines.
- Utility Routines.
- Mathematical Routines.

SYSTEM CONFIGURATIONS

The 6400 Computer System is constructed along logically expandable lines, allowing system growth to proceed from a minimum to a maximum system with little inconvenience. Three basic units can be arranged in the variety of configurations. They are:

- Central Memory expandable to provide the required amount of storage.
- Central Processor expandable to provide the required amount of computational ability.
- Input/Output Controllers required for the complement of peripheral equipment.

The over-all construction design of the 6400 allows a great variety in computer system configurations. This variety provides the user with the ability to tailor his particular system to satisfy the precise needs of his installation. System growth proceeds by adding the appropriate units as the need arises. Added reliability is obtained because identical units are employed in expanded systems, providing backup for each other if individual units malfunction. Figure 1-1 shows a typical on-line system configuration of the 6400 Computer System using one Central Processor, one Central Memory, and one Peripheral and Control Processor.

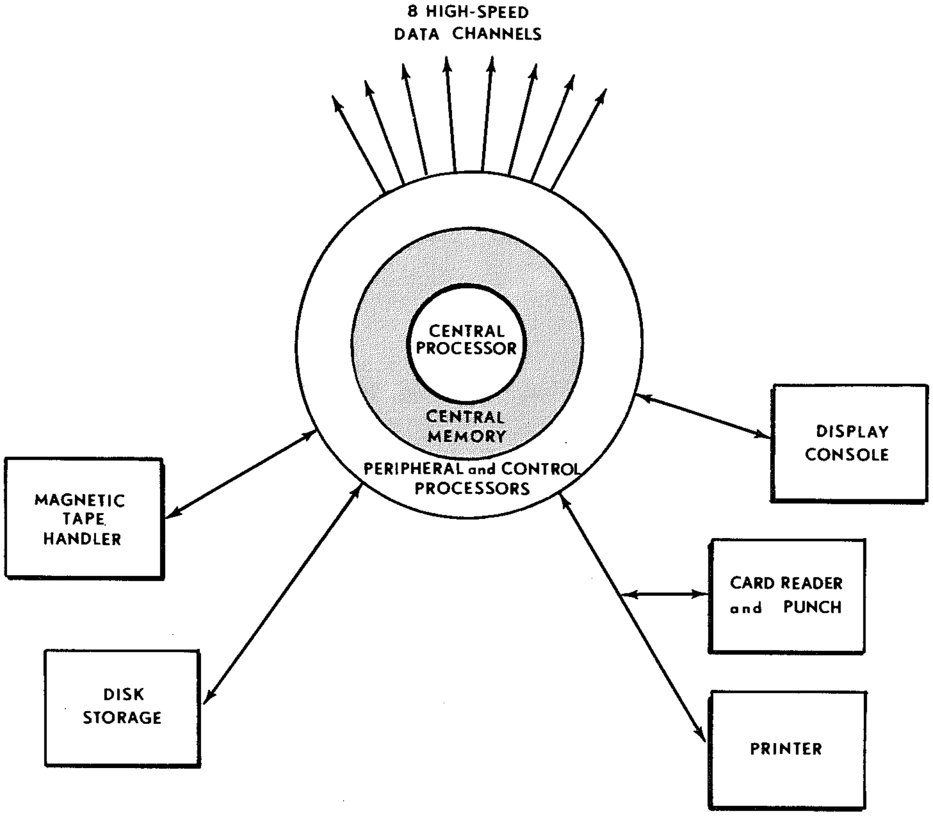


Figure 1-1. On-Line System Configuration

This basic on-line 6400 system meets all requirements necessary to classify it as a multiprocessor system. The Central Processor, in conjunction with the ten Peripheral and Control Processors, is a system with eleven independent processor units used in a cooperative computing scheme. One Peripheral and Control Processor acts in an executive and monitor capacity. Other Peripheral and Control Processors control all system operations and direct input-output operations. The Central Processor is therefore free to carry out the job it is designed for, namely, very high-speed logical and arithmetic operations. The efficiency of the basic multiprocessor system is inherently high.

An extension of this basic on-line multiprocessing system is shown in Figure 1-2. It consists of a second Central Processor integrated into the basic 6400 system. This dual Central Processor configuration extends the basic multiprocessing capability to an even greater degree. Each Control Processor is identical and allows completely independent programs to run simultaneously. Any Peripheral and Control Processor may interrupt and monitor either Central Processor. Each Central Processor has full access to the entire Central Memory, constrained only by the Boundary register contents.

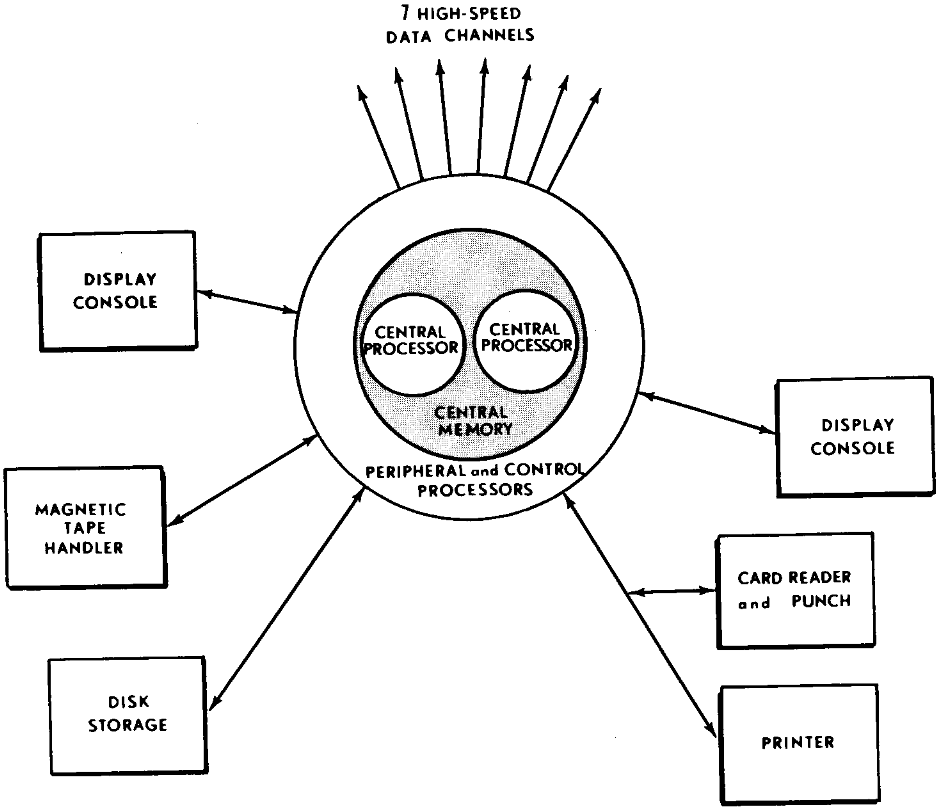


Figure 1-2. Dual Central Processor System

An expanded configuration for a 6400 Computer System is shown in Figure 1-3. This configuration is an extremely powerful multiprocessing system with greatly increased input-output capabilities. The four large-scale memories provide high-security memory specification for critical real-time requirements. Privacy of storage, as opposed to common storage, is the rule in this configuration.

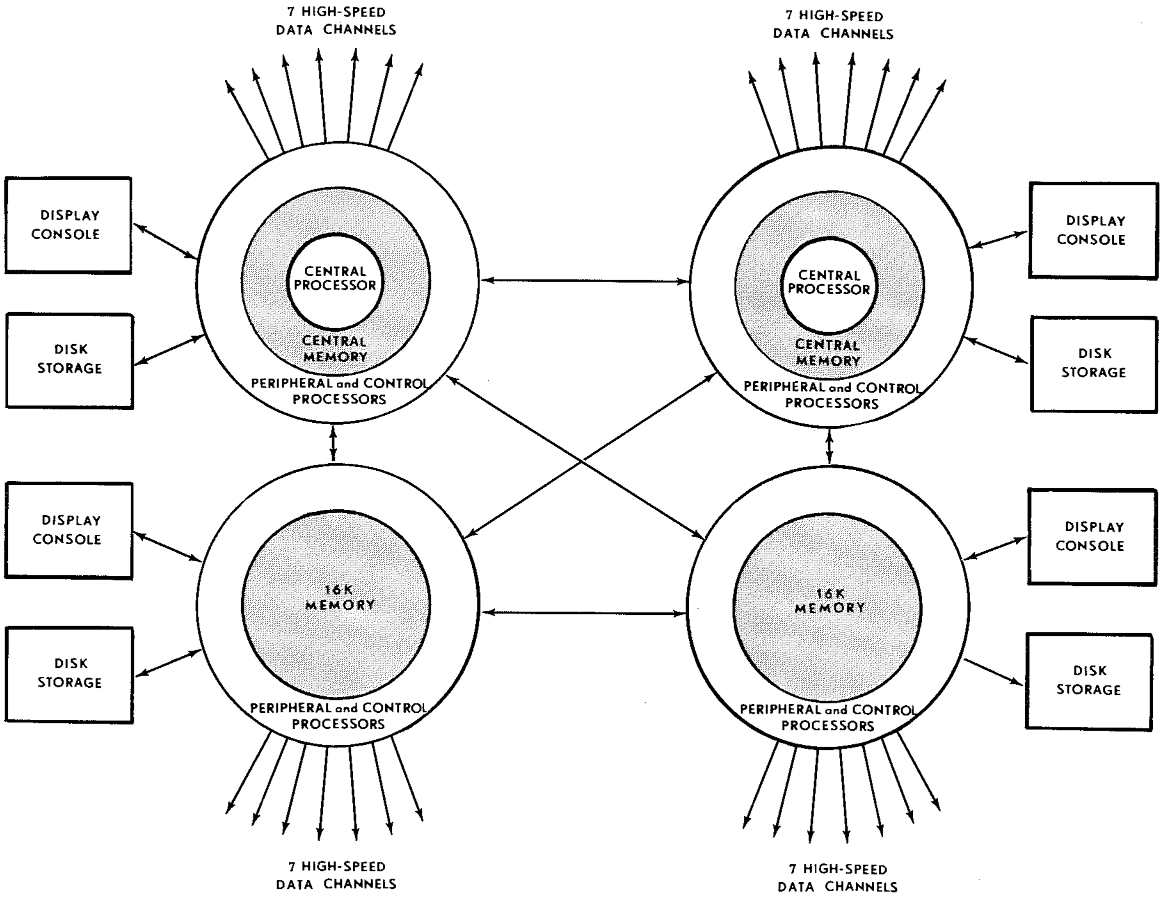


Figure 1-3. Multiprocessor Configuration with Real-Time Capability

SYSTEM FEATURES

Certain 6400 equipment features make significant contributions to the exceptional multiprogramming, high throughput, and rapid turn-around capabilities of the 6400. These features, together with the integrated programming system under the control of SIPROS, make possible the automatic supervision of the multiprogramming environment. System features include:

- Simultaneous Concurrency Levels.
- Disk Oriented System.
- Comprehensive Memory Protection.
- Dynamic Memory Relocation.
- Real-Time Interface.
- Systems Growth.

Simultaneous Concurrency Levels

The 6400 has sufficient independence between its functional segments to sustain a high number of concurrent operations, significantly increasing over-all system speed. Eleven programs may be simultaneously run on the eleven processors with each program maintaining a cooperative independence, each doing its assigned portion of the problem solution.

Overlapping of multiple memory banks provides a second level of concurrency by permitting several input-output operations to be in operation with simultaneous Central Processor requirements.

Disk-Oriented System

The use of the 6400 Disk System provides a significant part of the system's multiprogramming capabilities.

By accumulating portions of a given program's output during scattered periods of execution, the disk permits program switching without loss of previous executions whenever switching is necessary or advantageous.

After program completion and upon equipment availability, the accumulated output is transferred to specified peripheral devices under Peripheral Processor control, permitting maximum utilization of all system peripherals.

Additional uses of the disk include storage of the system library, the input job stack, and previously compiled programs for which execution has not been initiated. To the extent of job availability and installation-specified storage allocations, the input job stack is dynamically maintained at capacity to provide SIPROS with a wide choice in the scheduling of Central Processor operations.

Comprehensive Memory Protection

Two or more job programs simultaneously occupying Central Memory must be protected from each other so that one program does not inadvertently alter the area outside its boundary limits.

Relative Address and Field-Length registers provide a storage protection scheme which requires a minimum of attention from the operating system. It is a flexible system which permits a program to be located anywhere in memory and to be of any length consistent with memory size.

Dynamic Memory Relocation

The Relative Address register of the 6400 provides a method for simple and flexible relocation of programs in Central Memory. A programmer does not have to conform to specific rules to effect his relocation. The operating system merely moves the program and then resets the Relative Address register to the new location.

Real-Time Interface

The 6400 has twelve high-speed Data Channels which communicate with any of the ten Peripheral Processors, independent of the operation of the Central Processor. Peripheral Processors also possess complete instruction repetoires, further contributing to extensive flexibility for the monitoring and control of real-time or remote operations.

Systems Growth

The 6400 Computer System assures the user functional growth of his system as his needs increase. Compatibility of backup equipment and standardized peripheral equipment provide gradual broadening of a user's initial computing base.

In the event that the inherent flexibility and modularity of the 6400 system have been exploited to their economic limit, the 6400 user may upgrade to either a 6600 or 6800 system with no modification whatever of system organization, system operational procedures, system peripheral devices, or installation programs. The 6000 series is completely program compatible, both upward and downward, and is limited only by equipment configurations. This compatibility makes possible an uninterrupted increase in the 6000 series internal capabilities by a factor of from 4 to 16.

The 6400 opens up a wide range of data communication possibilities. Remote locations can be tied into the Central Computation System for remote inquiry and data processing, and multiple computer installations can communicate to form large computer complexes.

The flexibility of the 6400 system is virtually unlimited, since it may include any peripheral device or peripheral controller from either the Control Data 6000 or 3000 series. Similarly, any 6000 or 3000 series Computer System may be interfaced with the 6400 as either a master or peripheral system. Equally significant is the upward and downward program compatibility of the 6400 system with both the 6600 and 6800 systems, which makes the comprehensive 6600 programming packages, together with the advanced capabilities of the 6600 and 6800 systems, fully available to the 6400 user.

SYSTEM CHARACTERISTICS

Significant characteristics of both equipment and software available on the 6400 Computer System are outlined in the following:

Principle Hardware Features

- 60-bit Central Memory word length, with 4, 3, or 2 instructions per word.
- 24 high-speed transistor registers for storage of operands, addresses, and index constants.
- Central Memory options of 32,768, 65,536, or 131,072 words.
- Central Memory cycle time of 1 microsecond, with multiple interlacing of consecutive addresses permitting substantially higher access and transfer rates.
- Pluggable memory modules of 4096 12-bit words, accessible and replaceable within minutes.
- 12 bi-directional Data Channels accessible to all Peripheral Processors, with transfer rates of one million 12-bit words per second per channel.
- A 1 microsecond real-time clock, available to all Peripheral Processors through a 13th Data Channel.
- An extensive instruction set for both Central and Peripheral Processors, including facilities for peripheral control of Central Processor and input channel activities, block transfers between Central and Peripheral Memories, single step exchange of operating programs within the Central Processor, and multiple precision floating point arithmetic.
- Reference Address and Field Length registers, together with selectable program exit modes, for complete memory protection of Central Processor programs during multiprocessing operations.
- Compatibility with both 6000 series and 3000 series computers, channel synchronizers, and peripheral devices, including 500 million-bit disk storage units, 120 and 240kc tape transports, high speed drums, printers, card readers and card punches, display consoles, and special purpose devices.
- Program compatibility, both upward and downward, with the Control Data 6600 and 6800 Computer Systems, assuming appropriate channel assignments.

Principle Software Features

SIPROS

- Multiprocessing capabilities using a flexible priority scheme.
- Dynamic memory reassignment during multiple operations.
- Provision for automatic batch processing.
- Dynamic assignment and release of I/O equipment.
- Provision of a queuing system for operation of low speed I/O devices.
- Ready adaptability to special purpose executives for real-time or hybrid applications.
- Integration of a diagnostic system under operating system control.
- Flexible provisions for operator intervention and override.
- Maximum use of console display and keyboard for two-way communications.
- Provision of a comprehensive accounting system for time use recording of all system components.

FORTRAN 66

- Language compatibility with FORTRAN 63 and, essentially, with FORTRAN IV.
- Provision for mixed assembly language and FORTRAN statements on a line-for-line basis.
- Provision for use of register names as operands.
- Provision for automatic I/O buffering.

ASCENT and ASPER

- Provision for system macros and programmer-defined macros.
- Provision for subroutine and Peripheral Processor program calls.
- Pseudo commands.

LIBRIOUS

- Provision for addition or deletion of installation-oriented programs and subroutines.

PROGRAMMING AND OPERATING SYSTEM

The basic features of the 6400 Computer System and the manner in which these features interact implies a fundamental requirement for control program integration with the equipment. This over-all system integration is achieved through the Simultaneous Processing Operating System, SIPROS, which is provided by Control Data Corporation.

The 6400, regardless of configuration, functions entirely under the control of SIPROS. The integrated programming system is implemented in SIPROS-controlled modules, enabling the user to select the required modules for a given application and modify modules as his programming needs change.

SIPROS utilizes the previously discussed equipment to form a comprehensive system for the supervision of a multiprogramming environment. It controls all activities, including real-time interrupt response, job scheduling, equipment allocation, input/output supervision, local and remote media conversion, and routine housekeeping and accounting functions.

The entire programming system reflects the design objectives of providing user-oriented software. Programming system characteristics include:

- SIPROS (and other system software) operates from a random access disk storage base. For this reason, magnetic tapes are not required for SIPROS library files, FORTRAN compilation or other system program storage or execution. This mass random access capability provides for maximum system throughput and minimum job turn-around time.
- Symbolic file assignments provide complete flexibility of peripheral equipment allocation for dynamic scheduling and system reconfiguration.
- Parametric control programs are written in modular form, are well-documented, and can be readily modified to tailor the system to specific operating conventions and application requirements.
- Programmers writing for the 6400 system need not be concerned that their programs will be executed in a multiprogramming environment. Accepted conventions free the programmer from concern about absolute memory and peripheral requirements.

II. MEMORY ORGANIZATION

The organization of the Central Memory in the 6400 is designed to optimize the multiprogramming capability of the system. In light of this, the Central Memory is carefully integrated into the total systems design of the 6400.

BI-DIRECTIONAL MEMORY

One of the definitions of a multiprocessor system is the need for memory-to-memory communication between major elements of the system at memory access speeds. The Central Memory organization of the 6400 meets this definition by logically placing the Central Memory between the Central Processor and Peripheral and Control Processors. This organization of Central Memory isolates the Central Processor from the Peripheral and Control Processors, while allowing communication between the two at memory access speeds.

Addressing of Central Memory is accomplished independently by the Central Processor for its needs and by the Peripheral and Control Processors for their needs. A common address clearing scheme is used to coordinate and control the addressing of Central Memory locations.

ADDRESS FORMAT

The address word for Central Memory references is a 12-bit address quantity and a 5-bit bank quantity which defines one of 8, 16, or 32 banks. The 12-bit quantity defines 4096 separate locations (or addresses) in each bank.

In order to make the most efficient use of the bank phasing feature, addresses written or compiled in the conventional manner reference consecutive banks.

Not Used	Address	Bank	
3	12	3	8 Banks
2	12	4	16 Banks
1	12	5	32 Banks

CORE STORAGE UNITS

Central Memory is organized into either 32K, 64K or 128K words (60-bit) in either 8, 16 or 32 banks of 4096 words each. The banks are logically independent, and consecutive addresses go to different banks.

Several banks of Central Memory may be in operation simultaneously, thereby minimizing execution time.

III. CENTRAL PROCESSOR

The Central Processor is an extremely high-speed arithmetic processor which communicates only with Central Memory. It consists (functionally) of an arithmetic unit and a control unit. The arithmetic unit contains all logic necessary to execute the arithmetic, manipulative and logical operations. The control unit provides the register interface between the arithmetic unit and Central Memory. It also performs instruction fetching, addressing preparation, memory protection, and data fetching and storing. Both units operate with relative independence and maximum overlap to provide the highest rate of instruction execution.

PROCESSOR INDEPENDENCE

The Central Processor is isolated from the Peripheral and Control Processors by Central Memory; thus, the Central Processor is free to carry on high-speed computation unencumbered by input/output requirements.

INSTRUCTION FORMATS

Central Processor program instructions are stored in Central Memory. A 60-bit memory location may hold 60 data bits, four 15-bit instructions, two 30-bit instructions, or a combination of 15 and 30-bit instructions. Figure 3-1 shows all instruction combinations in a 60-bit word and the two instruction word formats.

Groups of bits in an instruction are identified by the letters f, m, i, j, k, and K. All letters represent octal digits except K which is an 18-bit constant.

The f and m digits identify the type of instruction and are the operation code.

In most 15-bit instructions, the i, j, and k digits each specify one of eight operating registers where operands are found and where the result of the operation is to be stored. In other 15-bit instructions, the j and k digits provide a 6-bit shift count.

In 30-bit instructions the i and j digits each specify one of eight operating registers where one operand is found and where the result is to be stored, and K is taken directly as an 18-bit second operand.

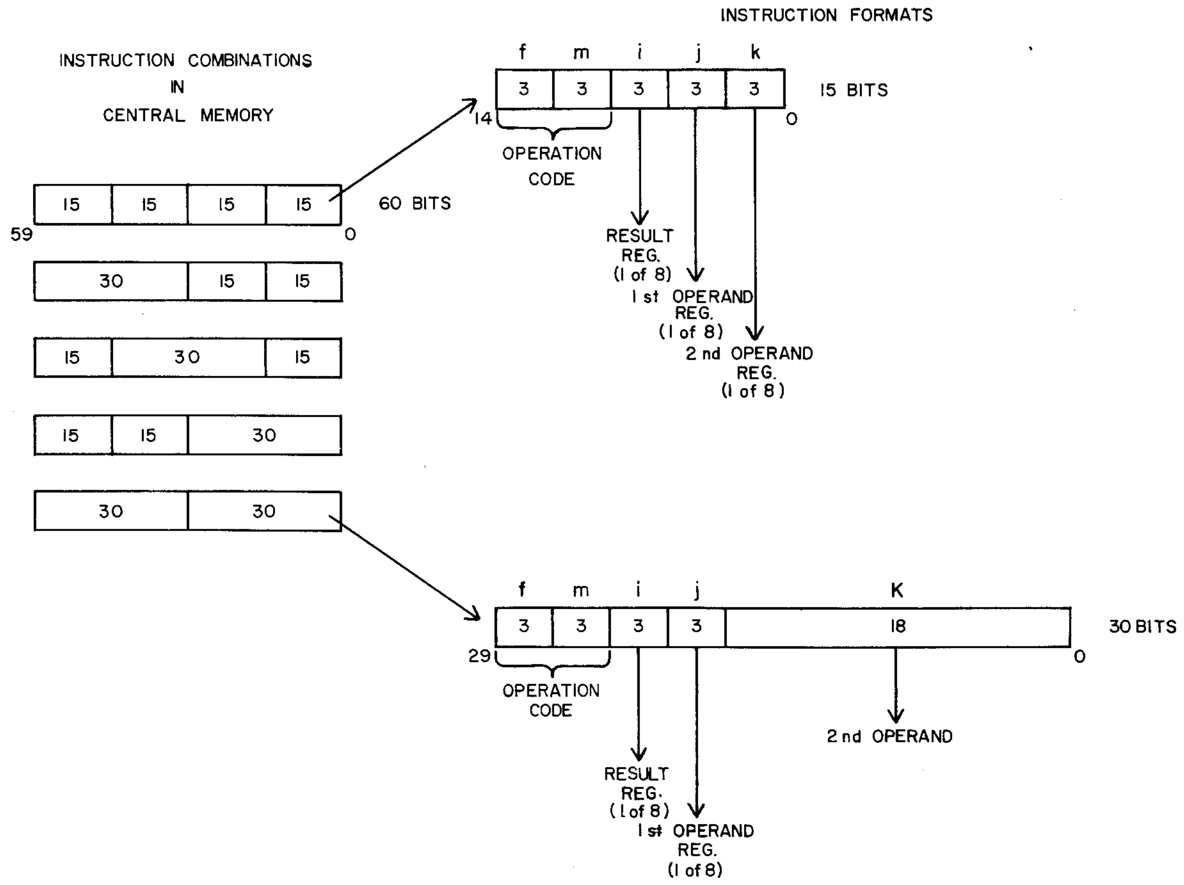


Figure 3-1. Central Processor Instruction Formats

CENTRAL PROCESSOR REGISTERS

Twenty-four operating registers are provided to lower the Central Memory requirements for operands and results. These registers are identified by letters and numbers to facilitate a compact symbolic language. The operating registers are identified as follows:

- A = address registers (A0, A1 . . . . . . A7)
- B = increment registers (B0, B1 . . . . B7)
- X = operand registers (X0, X1 . . . . . X7)

A and X Registers

Five registers (X1-X5) hold read operands from Central Memory, and two registers (X6-X7) send results to Central Memory. Operands and results transfer between memory and these registers as a result of a change in the contents of a corresponding address register (A1-A7).

A change in the contents of an address register A1-A5 produces an immediate memory reference to that address and reads the operand into the corresponding operand register X1-X5. Similarly, a change in the contents of address register A6 or A7 stores the word in the corresponding X6 or X7 operand register in the new address.

The increment instructions with the Ai result register change in A1-A7 address register in several ways.

- By adding an 18-bit signed constant K to the contents of any A, B, or X register.
- By adding the content of any B register to any A, B, or X register.
- By subtracting the content of any B register from any A register or any other B register.

The A0 and X0 registers are independent and have no connection with Central Memory. They may be used for scratch pad or intermediate results.

B Registers

The B registers have no connection with Central Memory. The B0 register is fixed to provide a constant zero (18-bit) which is useful for various tests against zero, providing an unconditional jump modifier, etc. In general, the B registers provide means for program indexing. For example, B4 may store the number of times a program loop has been traversed, thereby providing a terminal condition for a program exit.

An Exchange Jump instruction from a Peripheral and Control Processor enters initial values in the operating registers to start Central Processor operation. Subsequent address modification instructions executed provide the address changes required to fetch and store data.

Program Address Register

An 18-bit P register serves as a program address counter and holds the address of each program step. P is advanced to the next program step in the following ways:

- P is advanced by 1 when all instructions in a 60-bit word have been extracted and sent to the Instruction register(s).
- P is set to the address specified by a go to... (branch) instruction. If the instruction is a Return Jump, P + 1 is stored before the branch to allow a return to the sequence after the branch.
- P is set to the address specified in the Exchange Jump package.

All branch instructions to a new program start the program with the instruction located in the highest order position of the 60-bit word.

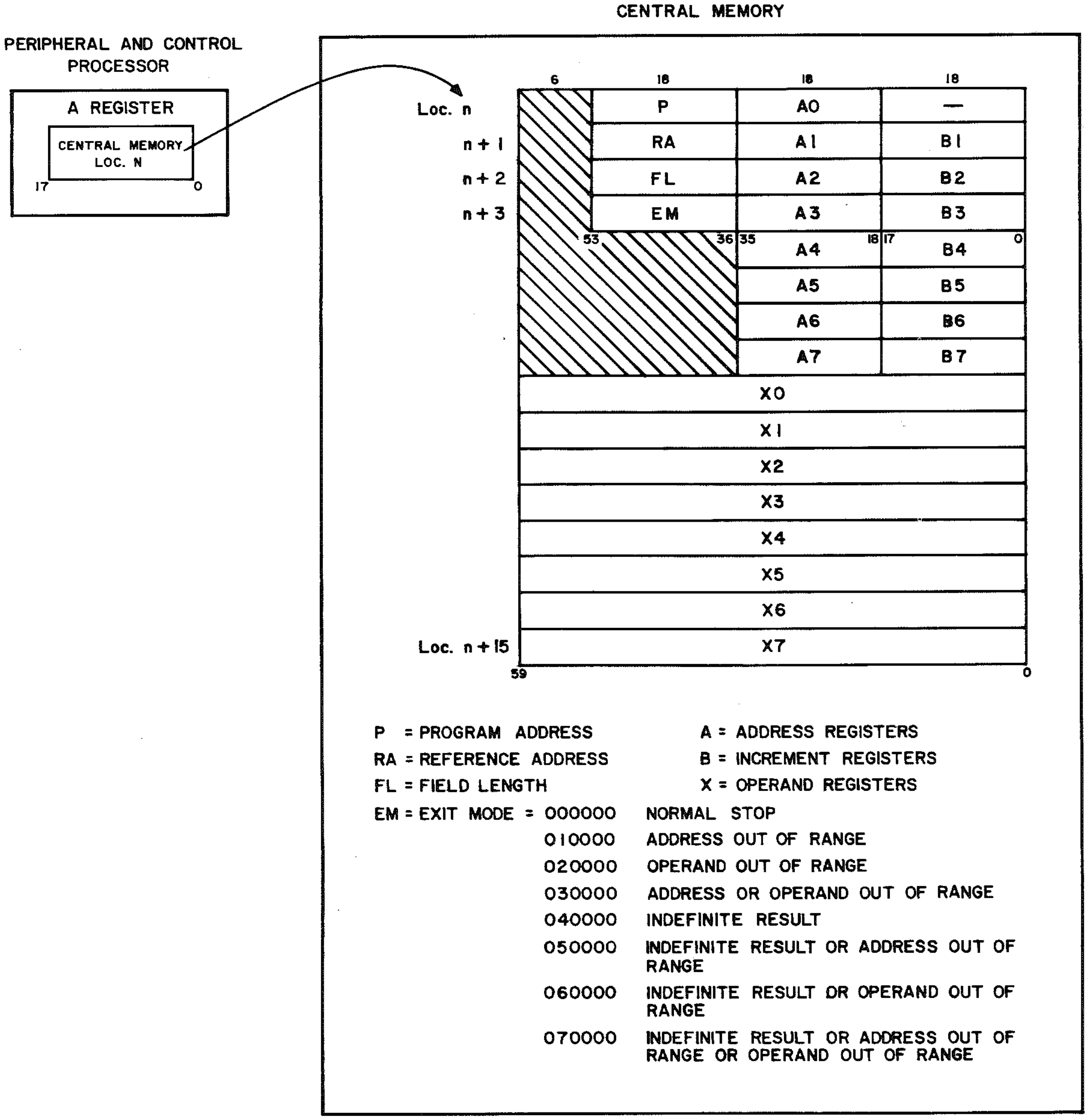
EXCHANGE JUMP

A Peripheral and Control Processor Exchange Jump instruction starts or interrupts the Central Processor and provides it with the first address (which is contained in the Peripheral and Control Processor A register) of a 16-word package in Central Memory (see Figure 3-2). The Exchange Jump package provides the following information on a program to be executed:

- Program address (P).
- Reference address (RA).
- Field length of program (FL).
- Program Exit mode (EM).
- Initial contents of the eight A registers.
- Initial contents of the eight X registers.
- Initial contents of B registers B1-B7 (B0 is fixed at 0).

The Central Processor enters the information about a new program into the appropriate registers and then stores the corresponding and current information from the interrupted program at the same 16 locations in Central Memory. Hence two programs are exchanged. A later Exchange Jump may return an interrupted program to the Central Processor for completion. The normal relation of the A and X registers (described earlier) is not active during the Exchange Jump so that the new entries in A are not reflected into changes in X.

All Central Processor reference addresses to Central Memory for new instructions, or to fetch and store data, are made relative to the reference address. This allows easy relocation of a program in Central Memory. The reference address or beginning address and field length define the Central Memory limits of the program. An optional exit condition allows the Central Processor to stop on a memory reference outside these limits.



	6	18	18	18
Loc. n		P	A0	—
n + 1		RA	A1	B1
n + 2		FL	A2	B2
n + 3		EM	A3	B3
			A4	B4
			A5	B5
			A6	B6
			A7	B7
	X0			
	X1			
	X2			
	X3			
	X4			
	X5			
	X6			
Loc. n + 15	X7			

P = PROGRAM ADDRESS

RA = REFERENCE ADDRESS

FL = FIELD LENGTH

A = ADDRESS REGISTERS

B = INCREMENT REGISTERS

X = OPERAND REGISTERS

EM = EXIT MODE =	
000000	NORMAL STOP
010000	ADDRESS OUT OF RANGE
020000	OPERAND OUT OF RANGE
030000	ADDRESS OR OPERAND OUT OF RANGE
040000	INDEFINITE RESULT
050000	INDEFINITE RESULT OR ADDRESS OUT OF RANGE
060000	INDEFINITE RESULT OR OPERAND OUT OF RANGE
070000	INDEFINITE RESULT OR ADDRESS OUT OF RANGE OR OPERAND OUT OF RANGE

Figure 3-2. Exchange Jump Package

The Program Address register P defines the location of a program step within the limits prescribed. Each reference to memory is made to the address specified by P + RA. Hence, program relocation is conveniently handled through a single change to RA.

AP = 0 condition specifies relative address zero and hence RA. This address is reserved for recording program exit condition.

The Exit mode feature allows the programmer to choose the exit or stop condition of the Central Processor. Exit selections are stored in the arithmetic unit, and the exit occurs as soon as it is sensed. The various exit conditions are shown below in octal format:

EM = 000000	Normal stop.
= 010000	Address out of range — an attempt to reference memory outside established limits.
= 020000	Operand out of range—floating point arithmetic generated or regenerated an infinite result.
= 030000	Address or operand out of range.
= 040000	Indefinite operand—floating point arithmetic generated or regenerated an indefinite result.
= 050000	Indefinite operand or address out of range.
= 060000	Indefinite operand or operand out of range.
= 070000	Indefinite operand or operand or address out of range.

The Central Processor records at P = 0 a stop instruction, exit condition, and the program address at exit time in the format shown below and jumps to P = 0 (RA), thereby stopping.

Stop	Exit	P	
0 0	0 X	XXXXXX	0 0 0 0 0 0 0 0 0 0

P = 0, Normal stop
P = (P) + 1, at time of error exit

The Peripheral and Control Processor searches for a Central Processor P = 0 condition to determine that the latter has stopped. The contents of RA may then be examined to determine the nature of the stop.

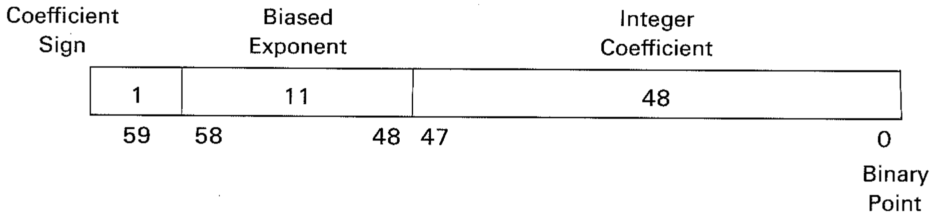
FLOATING POINT ARITHMETIC

Format

Floating point arithmetic takes advantage of the ability to express a number with the general expression kB^n, where

k = coefficient
B = base number
n = exponent, or power to which the base number is raised

The base number is constant (2) for binary-coded quantities and is not included in the general format. The 60-bit floating-word format is shown below. The binary point is considered to be to the right of the coefficient, thereby providing a 48-bit integer coefficient, the equivalent of about 15 decimal digits. The sign of the coefficient is carried in the highest order bit of the packed word. Negative numbers are represented in 1's complement notation.



The 11-bit exponent carries a bias of 2^{10} (2000_8) when packed in the floating point word (biased exponent sometimes referred to as characteristic). The bias is removed when the word is unpacked for computation and restored when a word is packed into floating format. The bias provides for a signed exponent within the following ranges:

2^{1023} $= 3777_8$
2^{0} $= 2000_8$ (zero $= 00000000000000000000_8$)
2^{-1023} $= 0000_8$

Thus, a number the true exponent of which is 342 would appear as 2342; a number the true exponent of which is −160 would appear as 1617. Exponent arithmetic is done in 1's complement notation. Floating point numbers can be compared for equality and threshold.

Normalizing and Rounding

Normalizing a floating point quantity shifts the coefficient left until the most significant bit is in bit 47. Sign bits are entered in the low-order bits of the coefficient as it is normalized. Each shift decreases the exponent by one.

A round bit is added (optionally) to the coefficient during an arithmetic process and has the effect of increasing the absolute value of the operand or result by one-half the value of the least significant bit. Normalizing and rounding are not automatic during pack or unpack operations so that operands and results may not be normalized.

Single and Double Precision

The floating point arithmetic instructions generate double precision results. Use of unrounded operands allows separate recovery of upper and lower half results with proper exponents; only upper half results can be obtained with rounded operands.

Range Definitions

A result the exponent of which is so large that it reaches or exceeds the upper limit of octal 3777 (overflow case) is treated as an infinite quantity. A coefficient of all zeros and an exponent of octal 3777 is packed for this case. An optional exit is provided for infinity since its later use may propagate an indefinite result as shown in Table 3-1.

A result the exponent of which is less than the lower limit of octal 0000 (underflow case) is treated as a zero quantity. This quantity is packed with a zero exponent and zero coefficient. No exit is provided for underflow. A result the exponent of which is octal 0000 and the coefficient of which is not zero, is a non-zero quantity and is packed with a zero exponent and the non-zero coefficient.

Use of either infinity or zero as operands may produce an indefinite result. An exponent of octal 1777 and a zero coefficient are packed in this case, and an optional exit provided. Note that zero, infinity, and indefinite results are generated or regenerated in the floating arithmetic units only; the exits are sensed in these units also. The branch unit instructions test for indefinite or infinite quantities.

Table 3-1. Indefinite Forms

∞ − ∞	= INDEFINITE	∞ ÷ N = ∞
∞ ÷ ∞	= INDEFINITE	∞ + N = ∞
∞ • 0	= INDEFINITE	∞ − N = ∞
0 ÷ 0	= INDEFINITE	N ÷ 0 = ∞
INDEFINITE +, −, ÷, • (X)	= INDEFINITE	0 ÷ ∞ = 0
∞ + ∞	= ∞	0 • 0 = 0
∞ • ∞	= ∞	0 ÷ N = 0
∞ ÷ 0	= ∞	N ÷ ∞ = 0
where: ∞ = INFINITY, N = INTEGER X = ∞, N or 0		0 • N = 0

FIXED POINT ARITHMETIC

Fixed point addition and subtraction of 60-bit numbers are handled by the long add instruction category. Negative numbers are represented in 1's complement notation, and overflows are ignored. The sign bit is in the high-order bit position (bit 59), and the binary point is at the right of the low-order bit position (bit 0).

The increment instructions provide an 18-bit fixed point add and subtract facility. Negative numbers are represented in 1's complement notation, and overflows are ignored. The sign bit is in the high-order bit position (bit 17), and the binary point is at the right of the low-order bit position (bit 0). The increment instructions allow program indexing through the full range of Central Memory addresses.

CENTRAL PROCESSOR INSTRUCTIONS

Most of the 6400 Central Processor instructions are familiar to users of large-scale computing systems. These instructions may be categorically grouped as follows:

- Branch
- Boolean
- Shift
- Add
- Long Add
- Multiply
- Divide
- Increment

For a definitive list of Central Processor instructions, refer to Appendix A.

Branch

Both conditional and unconditional jumps (or branches) from the program can be effected with this category of instructions.

Shift

Instructions in this category handle operations basic to shifting. This includes left (circular) and right (end-off sign extension) shifting, and normalize, pack and unpack floating point operations.

Add

Floating point addition and subtraction on 60-bit floating point numbers or their rounded representation are included in this category.

Long Add

Instructions in this category perform 1's complement addition and subtraction of 60-bit fixed point numbers.

Multiply

These instructions perform floating point multiplication on floating point numbers or their rounded representation.

Divide

This instruction category performs floating point division of floating point quantities or their rounded representation. Also included is a sum of the number of 1's in a 60-bit word.

Increment

One's complement addition and subtraction of 18-bit numbers is accomplished by this group of instructions.

IV. PERIPHERAL AND CONTROL PROCESSORS

The ten Peripheral Processors of the 6400 system are the basis of its outstanding efficiency and flexibility for multiprocessing. By their independent performance of system control, monitor, and I/O functions, all concurrent with Central Processor operations, the Peripheral Processors permit full utilization of the Central Processor in its intended role as a high speed problem solver, unencumbered by the delays for *overhead* functions and I/O transmissions that are unavoidable in the conventional job stack processor.

Each processor has access to any of 12 high speed Data Channels, providing system transmission rates of 2 million characters per second per channel. (A sustained rate over 800,000 characters per second per channel can be maintained in a practical operating system.) Since the Peripheral Processors are fully independent of Central Processor activities and possess complete instruction repertoires, they provide extensive flexibility for the monitoring and control of real-time or remote operations. Additionally, they may be assigned, where appropriate, to tasks completely unrelated to Central Processor activities.

PERIPHERAL AND CONTROL PROCESSOR REGISTERS

Four registers in each of the Peripheral Processors are A, P, Q, and K. Each plays an important part in the execution of processor instructions.

A Register (18 bits)

The Arithmetic or A register is an adder. Quantities are treated as 1's complement. No sign extension is provided for 6-bit or 12-bit quantities which are entered in the low order bits; however, the unused high order bits are cleared to zero. Zero is represented by all zeros. The A register holds an 18-bit Central Memory address during several instructions. A also participates in Shift, Logical, and some I/O instructions.

P Register (12 bits)

The Program Address register or P register holds the address of the current instruction. At the beginning of each instruction, the contents of P are advanced by one to provide the address of the next instruction in the program. If a jump is called for, the jump address is entered in P.

Q Register (12 bits)

The Q register holds the lower six bits of a 12-bit instruction word, or, when the six bits specify an address, Q holds the 12-bit word which is read from that address. Q is an adder which may add +1 or −1 to its content.

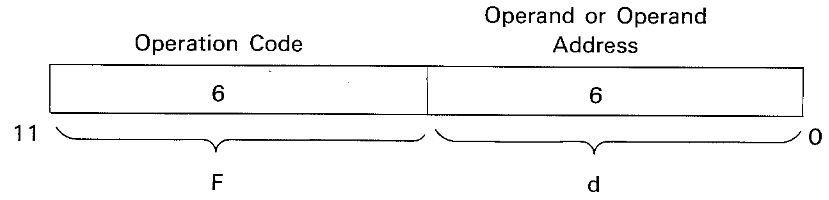
K Register (9 bits)

The K register holds the upper six bits (Operation code) of an instruction and a 3-bit trip count designator, which lends control to the sequential execution of an instruction.

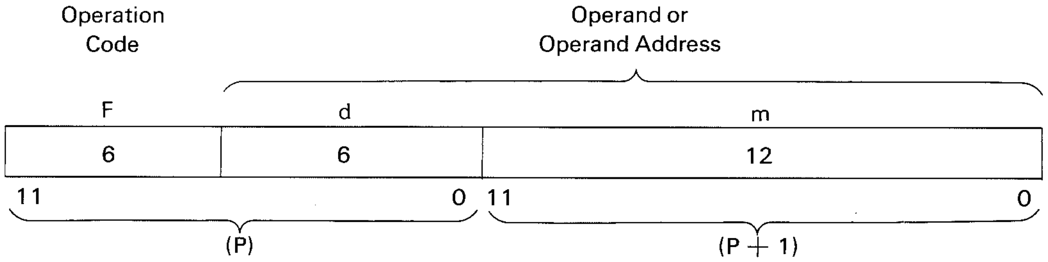
There are other registers which provide indirect or transient control during execution of instructions. These include registers associated with the I/O channels, the registers in the read and write pyramids which assemble successive 12-bit words into 60-bit words or vice versa, and registers which hold the reference address and the word at that address for each peripheral memory.

INSTRUCTION FORMAT

An instruction may have a 12-bit or a 24-bit format. The 12-bit format has a 6-bit Operation code F and a 6-bit operand or operand address d.



The 24-bit format uses the 12-bit quantity m, which is the content of the next program address (P + 1), with d to form an 18-bit operand or operand address.



ADDRESS MODES

Program indexing is accomplished and operands manipulated in several modes. The two instruction formats provide for 6-bit or 18-bit operands and 6-bit, 12-bit, or 18-bit addresses.

No Address

In this mode, d or dm is taken directly as an operand. This mode eliminates the need for storing many constants in storage. The d quantity is considered as a 12-bit number, the upper six bits of which are zero. The dm quantity has d as the upper six bits and m as the lower 12 bits.

Direct Address

In this mode, d or m + (d) is used as the address of the operand. The d quantity specifies one of the first 64 addresses in memory ($0000\text{-}0077_8$). The m + (d) quantity generates a 12-bit address for referencing all possible peripheral memory locations ($0000\text{-}7777_8$). If $d \neq 0$, the content of address d is added to m to produce an operand address (indexed addressing). If $d = 0$, m is taken as the operand address.

Indirect Address

In this mode, d specifies an address, the content of which is the address of the desired operand. Thus, d specifies the operand address indirectly. Indirect addressing and indexed addressing require an additional memory reference over direct addressing.

The list of instructions (Appendix A) uses the expression (d) to define the contents of memory location d. An expression with double parentheses ((d)) refers to indirect addressing. The expression (m + (d)) refers to direct addressing when $d = 0$ and to indexed direct addressing when $d \neq 0$.

ACCESS TO CENTRAL MEMORY

The Peripheral and Control Processors have access to all Central Memory storage locations. Four of the instructions transfer one word or a block of words from a peripheral memory to Central Memory or vice versa. Data from an external equipment is read into a peripheral memory and, with separate instructions, transferred from there to Central Memory where it may be used by the Central Processor. Conversely, data is transferred from Central Memory to a peripheral memory and then transferred by separate instructions to external equipment.

Read Central Memory

The 60 and 61 instructions read one word or a block of 60-bit Central Memory words. The Central Memory words are delivered to a five-stage read pyramid where they are disassembled into five 12-bit words, beginning with the high-order word. Successive stages of the pyramid contain 60, 48, 36, 24, and 12 bits. The upper 12 bits of the word are removed and sent to a peripheral memory as the word is transferred through each stage. Thus, a 60-bit word is disassembled into five 12-bit words.

Words move through the pyramid when the stage ahead is clear. Once the next stage is clear, 12 bits of the word are sent to a peripheral memory, and the word is moved ahead to the cleared stage. The pyramid may be time-shared by up to four processors. Thus four Central Memory words may be in the pyramid at one time in varying stages of disassembly.

With a full pyramid, read instructions from other processors are partially executed (housekeeping) and circulated unchanged until the number of pyramid users drops below four. Waiting processors are serviced in order. Other instruction control provides address incrementing and keeps the word count.

The Central Memory starting address must be entered in A before a Read instruction is executed. A Load dm (20) instruction may be used for this. For a one word transfer, the d portion of the Read (60) instruction specifies the following:

d = peripheral address (0000-0077$_8$) of first 12-bit word; remaining words go to d + 1, d + 2, etc.

For block transfer, d and m of the Read (61) instruction specify the following:

(d) = number of Central Memory words to be transferred; reduced by one for each word transferred.

m = peripheral starting address: increased by one to provide locations for successive words. (A) is increased by one to locate consecutive Central Memory words.

Write Central Memory

The 62 and 63 instructions assemble 12-bit peripheral words into 60-bit words and write them in Central Memory. Peripheral words are assembled in a write pyramid and delivered from there to Central Memory. As in Read Central Memory, the pyramid is time-shared by up to four processors. Write pyramid action is similar to read pyramid action except for the assembly.

The starting address in Central Memory is entered in A before the Write instruction is executed. For a one word transfer, the d portion of the Write (62) instruction specifies the following:

d = peripheral address (0000-0077$_8$) of first 12-bit word; remaining words are taken from d +1, d + 2, etc.

For block transfer, d and m of the Write (63) instruction specify the following:

(d) = number of Central Memory words to be transferred; reduced by one for each word transferred.

m = peripheral starting address; increased by one to locate each successive peripheral word. (A) is increased by one to provide consecutive Central Memory locations.

ACCESS TO CENTRAL PROCESSOR

The Peripheral and Control Processors use two instructions to communicate with the Central Processor. One instruction starts a program running in the Central Processor, and the other instruction monitors the progress of the program.

Exchange Jump

The 26 instruction initiates program execution in the Central Processor or interrupts a current program and starts a new program running. In either case, the Central Processor is directed to a Central Memory file of 16 words which stores information about the new program to be executed. The 18-bit starting address of this file must be entered in A before the Exchange Jump instruction is executed. The Central Processor replaces the file with similar but current information from the interrupted program. A later Exchange Jump instruction referencing this file returns the interrupted program to the Central Processor for completion. This exchange feature permits the Peripheral Processors to time-share the Central Processor.

Read Program Address

The 27 instruction transfers the content of the Central Processor P register into a peripheral A register. The peripheral program tests the A register content to determine the condition of the Central Processor. If A $\neq$ 0, the Central Processor is running a program. If A = 0, the Central Processor has stopped in a Normal or Exit Mode; the reference address for the Central Processor program is then examined to determine which condition exists. A Stop instruction (00_8) in the upper six bits of the reference address signals a stop; the next lower six bits define the nature of the exit.

PERIPHERAL AND CONTROL PROCESSOR INSTRUCTIONS

The instruction repertoire for the Peripheral and Control Processors may be logically grouped as follows:

- Data Transmission
- Shift
- Arithmetic
- Jump
- Logical
- Replace
- Central Processor and Central Memory
- Input/Output

For a definitive list of Peripheral and Control Processor instructions, refer to Appendix A.

Data Transmission

This category of instructions permits loading the A register with peripheral memory operands and storing the contents of A into peripheral memory. Various addressing modes lend flexibility to this category.

Shift

The Shift instruction shifts a quantity right (end off with no sign extension) or left (circular), whichever is selected.

Arithmetic

Instructions in this group provide for fixed-point addition and subtraction of 12 and 18-bit operands, depending on the addressing mode used.

Jump

Unconditional, conditional, and return jumps (or branches) may be effected in the program with this group of instructions.

Logical

This group of instructions performs several basic logical operations such as Logical Difference, Logical Product, and Selective Clear, with variations resulting from using various addressing modes.

Replace

Using various addressing modes, replace operations such as Replace Add, Replace Subtract, Replace Add One, and Replace Subtract One are performed with this group of instructions.

Central Processor And Central Memory

Two instructions in this group relate to the Central Processor: a) an Exchange Jump instruction to begin the Exchange Jump, and b) a Read Program Address instruction which reads the program address of the Central Processor.

Other instructions in this group permit the Peripheral Processors to read and write from/to Central Memory in single words or in blocks.

Input/Output

The Input/Output instructions direct all activity with external equipment. These instructions determine the status of and select an equipment on any channel, and transfer data to or from the selected device.

INPUT/OUTPUT

All Peripheral Processors communicate with external equipment and each other on 12 independent Data Channels. All channels are 12-bit (plus control) and each may be connected to one or more external devices. The channels are bi-directional, but data flows in only one direction at a time.

Data flows between a processor memory and the external device in blocks of words (a block may be as small as one word). A single word may be transferred between an external device and the A register of a processor.

Any processor may determine the condition of any equipment on any channel; thus, simultaneous I/O operations may be carried out in an orderly manner. One processor may communicate with another over a channel which is selected for output by one and input by the other. A common channel can be reserved for inter-processor communication and order preserved by determining equipment and channel status.

Channel Flags

Two channel conditions are made available to all processors as an aid to orderly use of channels.

- Each channel has an active/inactive flag to indicate that it has been selected for use and is busy with an external device.
- Each channel has a full/empty flag to indicate that a word (function or data) is available in the register associated with the channel.

Either state of both flags can be sensed by the Input/Output instructions.

REAL-TIME CLOCK

A real-time clock reading is available on a channel which is separate from the twelve Data Channels. The clock period is 4.096 milliseconds. The clock starts with power on, runs continuously, and cannot be preset or altered. The clock may be used to determine program running time or other functions as required.

V. PERIPHERAL EQUIPMENT

All peripheral equipment communicates with the 6400 system through the Data Channels to the Peripheral and Control Processors. The Data Channels present a standard interface that can accept a variety of peripheral device controllers, such as magnetic disc, drum, and tape controllers; card reader and punch controllers; printer, perforated tape reader/punch, and data communication controllers.

A special Data Channel Converter permits connection of any 3000 series peripheral equipment to the 6400 system.

6402 DISPLAY CONSOLE

The 6402 unit comprises the display section of the 6400 system. The display console consists of two 10-inch cathode ray tube display units and a manual keyboard containing 50 alphanumeric and special characters.

The console may be selected to display in either Character or Dot mode. In the Character mode, two alphanumeric characters may be displayed for each 12-bit word sent from a Peripheral Processor. The three character sizes are:

- Small — 64 characters/line
- Medium — 32 characters/line
- Large — 16 characters/line

In the Dot mode, a pattern of dots (graphs, figures, etc.) may be displayed. Dots are generated for 12-bit words containing vertical or horizontal coordinates. Upon receipt of a y coordinate, a dot is painted at that y coordinate and the last previous x coordinate.

Typical operation of a display console in the 6400 system allocates one screen for presentation of operator directives. The remaining screen provides the operator with status information on the current problem or information on other problems being run. Although none of the registers or memory locations are displayed automatically, a Peripheral Processor control program can extract register information and Central Memory contents and send it to a screen for viewing. In addition, a control program can change Central Memory or register contents and can interrupt, step, or terminate a Central Processor program from manual input on the console keyboard.

The 6402 Display Console operates under the control of a Peripheral and Control Processor. This processor may operate several displays simultaneously. Multiple units minimize idle time in the system and allow simultaneous debugging and monitoring of many unrelated problems.

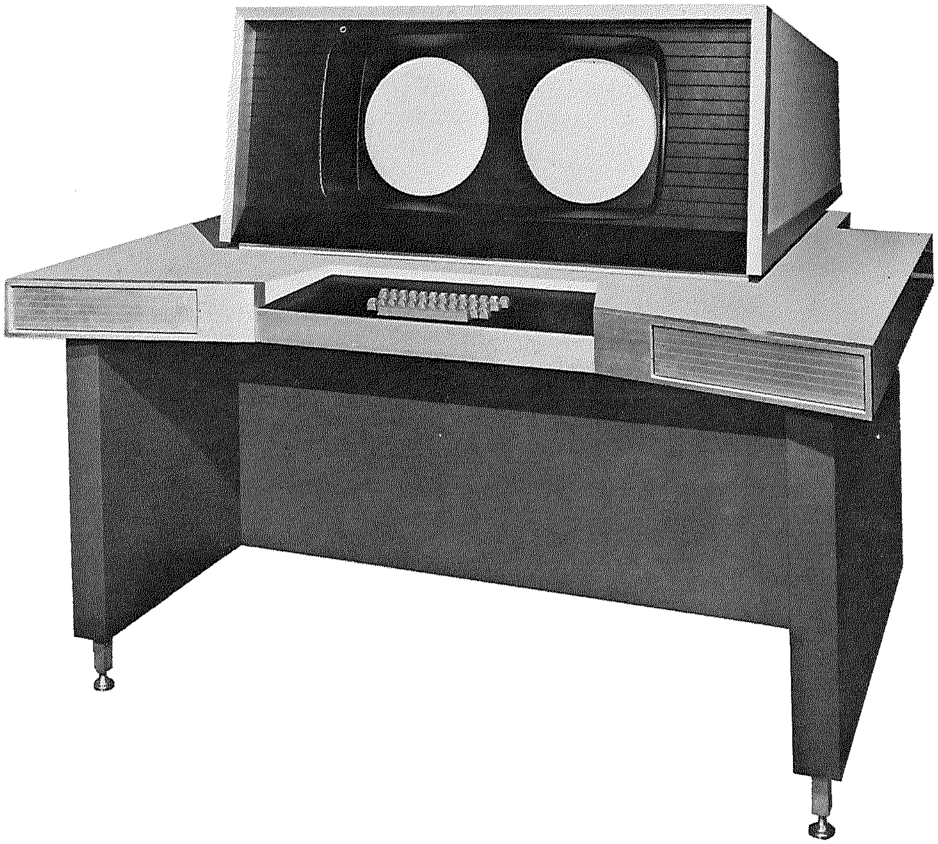


Figure 5-1. 6402 Display Console

6403 DISK FILE

The 6403 Disk File is the mass storage unit of the 6400 system. It has the following features:

- Capacity of 448,266,240 bits, or 74,711,040 characters.
- Transfer rate of 1.09385 x 10^6 characters/second (12 bits every 1.82 microseconds) for the inner zones when controlled by two Peripheral and Control Processors.

  1.4 x 10^6 characters/second maximum sustained rate for outer zones.

  1.2 x 10^6 characters/second average sustained rate.
- Rotational speed — 66.6 milliseconds/revolution.
- 266.4 millisecond average total random access time.
- 96 magnetic heads, 4 heads per disk surface. All heads are moved in unison by a single positioning arm.
- Error checking by program control and read parity checks.

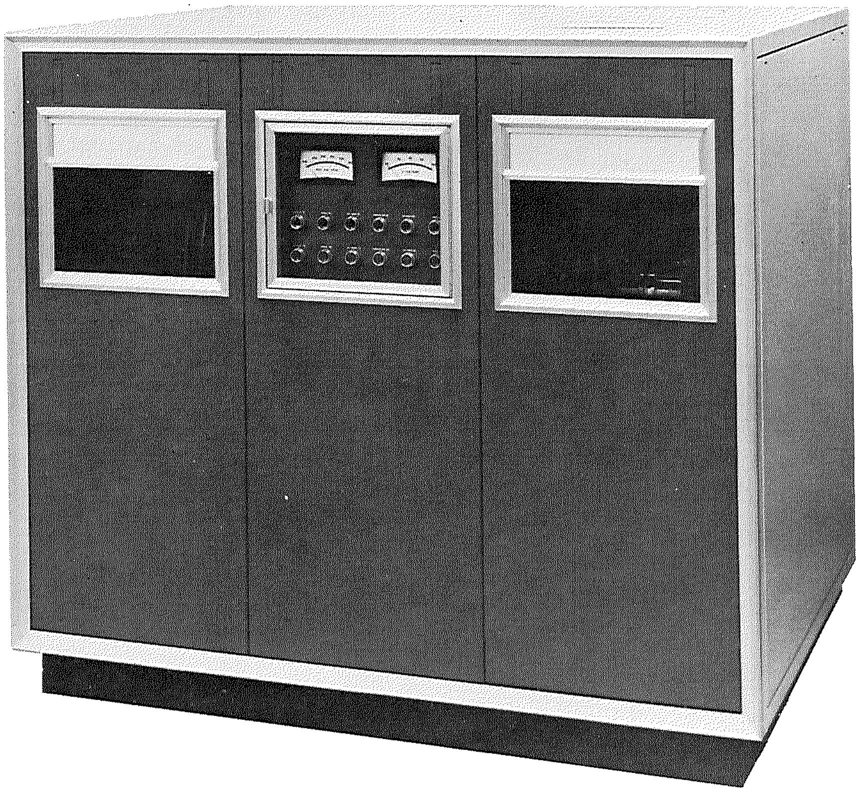


Figure 5-2. 6403 Disk File System

607-B MAGNETIC TAPE TRANSPORT

Control Data Corporation has developed a completely new series of magnetic tape transports which utilize advanced concepts in mechanical and electronic engineering. The Control Data 607-B Magnetic Tape Transport is one of this series and is specifically designed for use with large computer systems where high speed and reliability are of the utmost importance.

In terms of recording format and physical characteristics, the tapes prepared on all Control Data tape units, as well as those prepared on IBM 729 Series, are compatible with 607-B tapes. The tape is one-half inch wide and 2400 feet long. Six of the seven recording channels contain information; the other contains the parity check bit. A non-return-to-zero, change-on-ones recording method is used with 800, 556 or 200 characters per inch density. High density recording capability, coupled with a tape speed of 150 inches per second allows a character transfer rate of up to 120,000 characters per second.

The 607-B features complete pneumatic control of tape. The use of pneumatic capstans eliminates the need for pinch rollers or mechanical clutches and results in smooth, uniform acceleration. The tape is braked smoothly and precisely by a pneumatic brake pad. Vacuum tape loop boxes are used to provide uniform and precisely controlled tape tension. The vacuum column loops are photo-electrically sensed. The recording surface of the tape does not touch metal in the vacuum column or on the capstans.

The use of flat, combined, read-write heads permits immediate verification of written data and significantly reduces tape wear. Broad band erase head and tape cleaners insure clean tape for all activities. Reflective spots are used for loadpoint and end-of-tape sensing. Operator control is enhanced by quick change hubs, leaderless tape, and straight line threading.

The circuitry on the 607-B is completely solid-state. Solid-state circuitry combined with the use of pluggable modules and subassemblies makes maintenance easy and keeps down-time to a minimum.

626-B MAGNETIC TAPE TRANSPORT

The Control Data 626-B Magnetic Tape Transport is very much like the Control Data 607-B Magnetic Tape Transport. The 626-B is designed specifically as an intermediate storage device for the Control Data 6400 system. The 626-B tape unit uses one inch tape with 16 tracks per frame (across the tape). The frame format is as follows:

Track	*Information*
0	Sync bit
1-6	Bits $2^0 - 2^5$
7	Parity bit
8-13	Bits $2^6 - 2^{11}$
14	Parity bit
15	Sync bit

Information is recorded in the binary mode only. Each frame represents one 12-bit Peripheral Processor word, split into two characters, with an odd parity bit for each half.

The packing density is fixed at 800 bits per inch. Tape speed forward is 150 inches per second. The nominal transfer rate is 240,000 (six bit) characters per second.

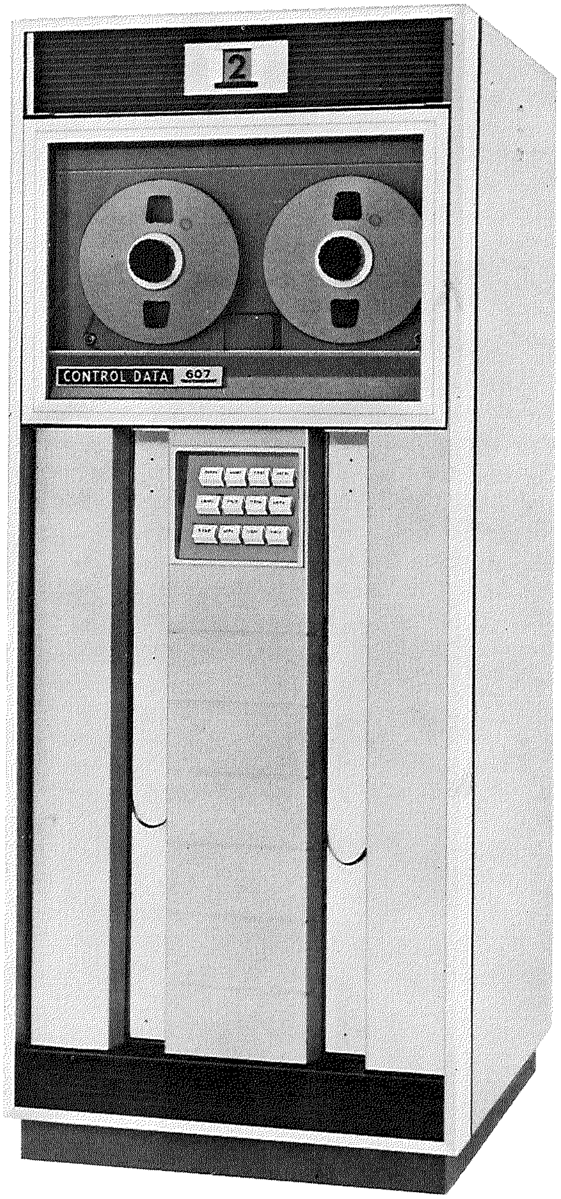


Figure 5-3. 607-B Magnetic Tape Transport

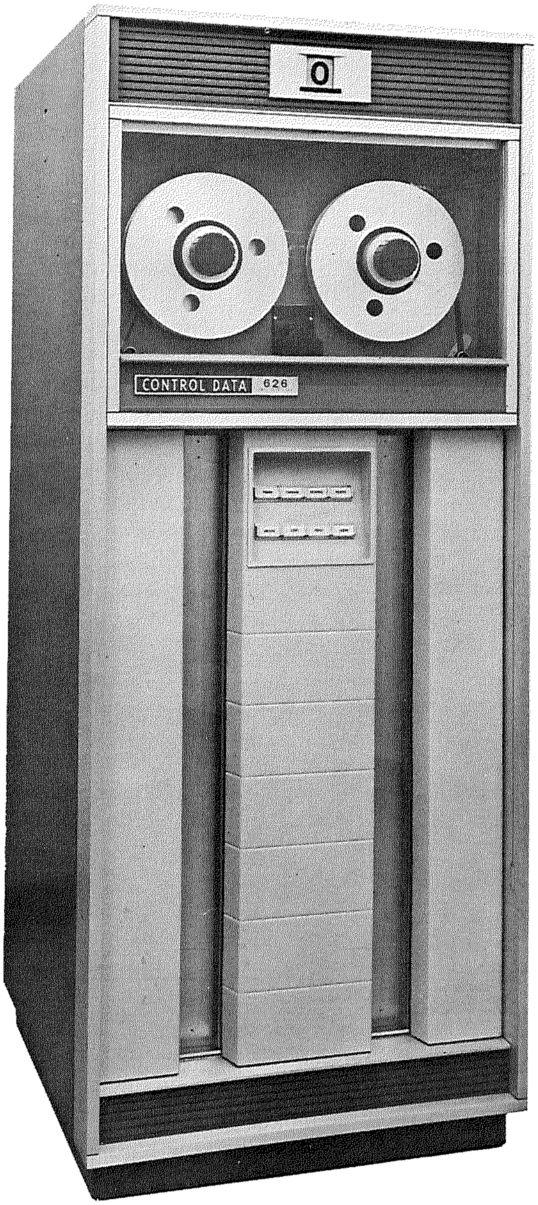


Figure 5-4. 626-B Magnetic Tape Transport

405 CARD READER

Significant advances in engineering have been incorporated in the Control Data 405 Card Reader. The combination of advanced card handling and reading techniques assures operational ease, maximum efficiency, high reliability, and long card life. The Control Data 405 Card Reader reads standard 80-column punched cards at 1200 cards per minute or 51-column punched cards at 1600 cards per minute. The punched cards are read by column rather than by row. Reading column by column eliminates the need for large storage and disassembly registers and frees the computer of the necessity to rearrange the card word.

The movement of punched cards in the 405 Card Reader is accomplished pneumatically. The input tray holds approximately 4,000 cards. Cards may be added to the input tray while the card reader is operating. Cards proceed down the hopper where they are picked, one at a time, with a pneumatic picker. Air is also injected at the bottom of the card stack to provide positive separation. The rotating vacuum capstan passes the card through a throat, past a brake-mechanism, and between a pinch roller where the card is transported past the read station.

After a card has been read, it may go either to the primary or secondary stacker. The primary stacker has a capacity of approximately 4,000 cards. The original sequence and orientation of the card deck is preserved. Cards will normally go to the primary stacker. The secondary stacker holds up to 240 cards. Cards may be directed to the secondary stacker by way of external function instructions from the computer. The secondary receiving tray may be used to store rejected cards or to do a limited amount of sorting. Cards may be added to or removed from either tray while the reader is in operation.

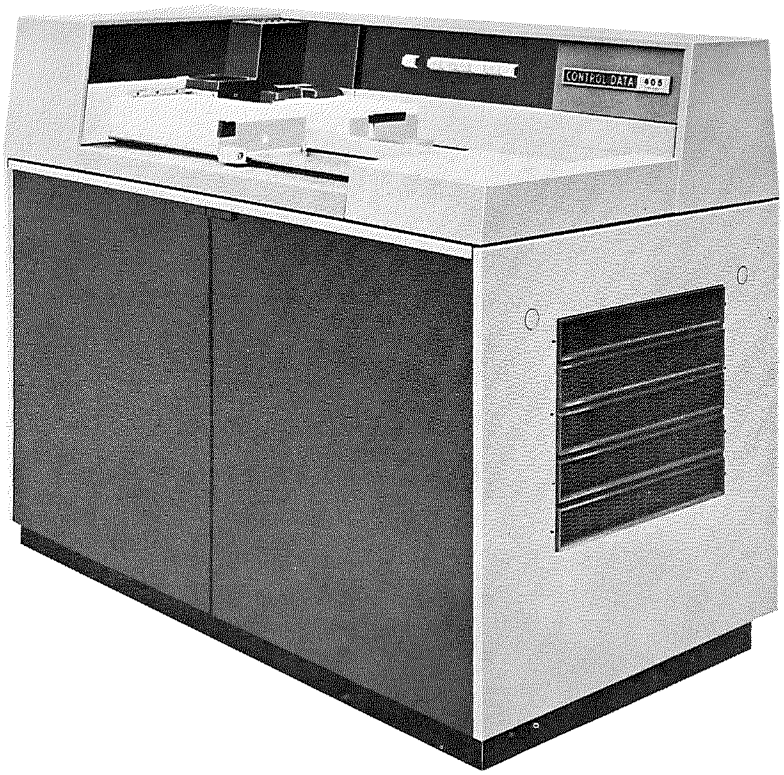


Figure 5-5. 405 Card Reader

Both the supply and receiving trays vibrate to give a constant force at the picker and stacker, regardless of the length of the deck in either tray. The handling of the cards in the Control Data 405 Card Reader has certain distinct advantages for the user. Aside from the facility of adding cards to the supply tray and removing them from the receiving tray, the 405 Card Reader also handles punched cards which are not perfectly flat.

The read station of the 405 Card Reader is a dual-read with photo-electric sensing; it monitors each vertical column of punched holes twice, comparing the two information blocks to automatically assure accuracy, and accepts or rejects cards on the basis of data error and comparison checks. Any errors noted during the read operation are made known to the computer through a status response.

501 LINE PRINTER

The Control Data 501 High-Speed Line Printer is a high-speed, on-line, data output device in direct communication with the Control Data 6400 system. The 501 is adaptable to any application requiring a large volume of printed output.

A single cabinet contains the entire print mechanism. Another cabinet contains all electronic control circuits. Solid-state components achieve maximum reliability of the control circuitry. Controls on the printer allow selection of print penetration and form adjustment.

The printer output format is controlled by a Peripheral Processor program. Programming a print operation requires the selection of a BCD code for each character to be printed on a line and storing these codes in a buffer area of the peripheral processor memory. After the program selects the printer and receives the Ready signal, the computer transmits the information to the printer. The printer stores, decodes, and sets the print hammers at the proper time.

CHARACTERISTICS

Printing Rate	1000 lines/minute
Paper Speed	1500 inches/minute, maximum
Line Length	136 columns
Line Spacing	6 lines/inch
Number of Characters	64 including blank
Character Spacing	10 per inch

415 CARD PUNCH

The 415 Card Punch is the newest member of the rapidly expanding line of high performance peripheral equipment manufactured by Control Data. The 415 permits punching of 80-column cards at 250 cards-per-minute. Read-after-write circuitry provides error checking capability.

DD-10 DISPLAY SYSTEM

The DD-10 system is an electronic typewriter display system. Each station consists of a keyboard and bright cathode ray tube display. The entire unit occupies only slightly more space than a typewriter.

Up to 64 stations may be operated from a single central control unit, which is capable of handling more than 20,000 inquiries or entries per hour.

Composing messages is much like using an electric typewriter except that the characters are stored in a buffer memory while the system is off-line from the computer. The stored message is displayed on the cathode ray tube screen. An index marker indicates where the next character to be typed will appear; the marker can be positioned anywhere on the page by the operator, and it moves automatically across the page as each character is typed. The operator can electronically erase or type over mistakes to insure the accuracy of the data before it is committed for computer entry.

When an inquiry is directed to the computer, an entire 500 character message is received and displayed in a fraction of a second.

861 DRUM STORAGE

The Control Data 861 Drum Storage is a peripheral memory device which features rapid access and addressable bytes. The byte size is 12 bits with each byte having an additional parity bit.

The characteristics of the 861 Drum are as follows:

Capacity

Data Bits — 25,165,824
Characters — 4,194,304
Data Tracks — 768
Bits per Track — 32,768

Organization

Bands per Drum — 64
Data Tracks per Band
(plus 1 parity bit per band) — 12
Bytes per Band (12 data bits
and 1 parity bit per byte) — 32,768
Addressable Bytes — 2,097,152
Characters per Byte — 2

Speed

Average Latency — 17 ms
Maximum Latency — 34 ms
Rotational Speed — 1800 rpm

Transfer

Minimum Quantity — 1 byte
Maximum Quantity — 2,097,152 bytes
Data Transfer Rate — 2 M characters
per second

The 861 Drum Storage stores and handles all data in a 12-bit parallel arrangement. With each 12-bit byte containing two characters, a data transfer rate of 2 mc per second is achieved. A significant feature of this drum is the ability to address as many bytes as desired or handle each byte separately.

The 861 Drum Storage uses a 3436 Controller. Drum parity is generated and checked in the controller. Up to eight 861's may be controlled by one 3436.

DATA CHANNEL CONVERTER

The Data Channel Converter permits connection of any 3000 series peripheral equipment to the 6400 system. Eight peripheral controllers may be connected to any Data Channel Converter.

APPENDIX A

INSTRUCTION EXECUTION TIMES

The execution times for Central Processor and Peripheral and Control Processor instructions are given in the following paragraphs. Factors which influence instruction execution time and hence program running time are also given.

Central Processor

The execution time of Central Processor instructions is given in minor cycles,* and instructions are grouped functionally. Time is counted from the time the unit has both input operands to the time when the instruction result is available in the specified result register. Central Memory access time is not considered in those increment instructions which result in memory references to read operands or store results.

The paragraphs following give some general statements about Central Processor instruction execution and summarize the statements into a list which may be used as a guide to efficient use of the Central Processor.

Central Processor programs are written in the conventional manner and are stored in Central Memory under direction of a Peripheral and Control Processor. After an exchange jump start by a Peripheral and Control Processor program, Central Processor instructions are sent automatically, and in the original sequence, to the arithmetic unit.

Peripheral And Control Processor

The execution time of Peripheral and Control Processor instructions is influenced by the following factors:

- Number of memory references — indirect addressing and indexed addressing require an extra memory reference. Instructions in 24-bit format require an extra reference to read m.
- Number of words to be transferred — in I/O instructions and in references to Central Memory the execution times vary with the number of words to be transferred. The maximum theoretical rate of flow is one word/major cycle. I/O word rates depend upon the speed of external equipments which are normally much slower than the computer.
- Central Memory conflicts — Central Read and Write instructions may be delayed if the desired Central Memory bank is busy.

Aside from the factors listed above, basic execution times vary from one major cycle for simple instructions to as much as 10 major cycles minimum for Central Read and Central Write instructions. (These instructions involve a minimum of five peripheral words.)

*A major cycle is 1000 nanoseconds and a minor cycle is 100 nanoseconds.

Central Processor Instruction Execution Times

(Times listed in Minor Cycles)

BRANCH

00	STOP	12
01	RETURN JUMP to K	12
02	GO TO K + Bi	12
030	GO TO K if Xj = zero	12
031	GO TO K if Xj ≠ zero	12
032	GO TO K if Xj = positive	12
033	GO TO K if Xj = negative	12
034	GO TO K if Xj is in range	12
035	GO TO K if Xj is out of range	12
036	GO TO K if Xj is definite	12
037	GO TO K if Xj is indefinite	12
04	GO TO K if Bi = Bj	12
05	GO TO K if Bi ≠ Bj	12
06	GO TO K if Bi ≧ Bj	12
07	GO TO K if Bi < Bj	12

BOOLEAN

10	TRANSMIT Xj to Xi	4
11	LOGICAL PRODUCT of Xj and Xk to Xi	4
12	LOGICAL SUM of Xj and Xk to Xi	4
13	LOGICAL DIFFERENCE of Xj and Xk to Xi	4
14	TRANSMIT Xk COMP. to Xi	4
15	LOGICAL PRODUCT of Xj and Xk COMP. to Xi	4
16	LOGICAL SUM of Xj and Xk COMP. to Xi	4
17	LOGICAL DIFFERENCE of Xj and Xk COMP. to Xi	4

SHIFT

20	SHIFT Xi LEFT jk places	5
21	SHIFT Xi RIGHT jk places	5
22	SHIFT Xk NOMINALLY LEFT Bj places to Xi	5
23	SHIFT Xk NOMINALLY RIGHT Bj places to Xi	5
24	NORMALIZE Xk in Xi and Bj	6
25	ROUND AND NORMALIZE Xk in Xi and Bj	6
26	UNPACK Xk to Xi and Bj	6
27	PACK Xi from Xk and Bj	6
43	FORM jk MASK in Xi	5

ADD

30	FLOATING SUM of Xj and Xk to Xi	11
31	FLOATING DIFFERENCE of Xj and Xk to Xi	11
32	FLOATING DP SUM of Xj and Xk to Xi	11
33	FLOATING DP DIFFERENCE of Xj and Xk to Xi	11
34	ROUND FLOATING SUM of Xj and Xk to Xi	11
35	ROUND FLOATING DIFFERENCE of Xj and Xk to Xi	11

LONG ADD

36	INTEGER SUM of Xj and Xk to Xi	6
37	INTEGER DIFFERENCE of Xj and Xk to Xi	6

MULTIPLY

40	FLOATING PRODUCT of Xj and Xk to Xi	56
41	ROUND FLOATING PRODUCT of Xj and Xk to Xi	56
42	FLOATING DP PRODUCT of Xj and Xk to Xi	56

DIVIDE

44	FLOATING DIVIDE Xj by Xk to Xi	56
45	ROUND FLOATING DIVIDE Xj by Xk to Xi	56
47	SUM of 1's in Xk to Xi	68

INCREMENT

50	SUM of Aj and K to Ai	5
51	SUM of Bj and K to Ai	5
52	SUM of Xj and K to Ai	5
53	SUM of Xj and Bk to Ai	5
54	SUM of Aj and Bk to Ai	5
55	DIFFERENCE of Aj and Bk to Ai	5
56	SUM of Bj and Bk to Ai	5
57	DIFFERENCE of Bj and Bk to Ai	5
60	SUM of Aj and K to Bi	5
61	SUM of Bj and K to Bi	5
62	SUM of Xj and K to Bi	5
63	SUM of Xj and Bk to Bi	5
64	SUM of Aj and Bk to Bi	5
65	DIFFERENCE of Aj and Bk to Bi	5
66	SUM of Bj and Bk to Bi	5
67	DIFFERENCE of Bj and Bk to Bi	5
70	SUM of Aj and K to Xi	5
71	SUM of Bj and K to Xi	5
72	SUM of Xj and K to Xi	5
73	SUM of Xj and Bk to Xi	5
74	SUM of Aj and Bk to Xi	5
75	DIFFERENCE of Aj and Bk to Xi	5
76	SUM of Bj and Bk to Xi	5
77	DIFFERENCE of Bj and Bk to Xi	5
46	Pass	

Octal Code at left of instruction

Comp.—Complement

DP—Double Precision

Peripheral and Control Processor
Instruction Execution Times

Octal Code	Name	Time (Major Cycles)	Octal Code	Name	Time (Major Cycles)
00	Pass	1		42 Subtract ((d))	3
				43 Logical difference ((d))	3
01	Long jump to m + (d)	2-3			
02	Return jump to m + (d)	3-4	44	Store ((d))	3
03	Unconditional jump d	1	45	Replace add ((d))	4
04	Zero jump d	1	46	Replace add one ((d))	4
05	Nonzero jump d	1	47	Replace subtract one ((d))	4
06	Plus jump d	1			
07	Minus jump d	1	50	Load (m + (d))	3-4
10	Shift d	1	51	Add (m + (d))	3-4
11	Logical difference d	1	52	Subtract (m + (d))	3-4
12	Logical product d	1	53	Logical difference (m + (d))	3-4
13	Selective clear d	1	54	Store (m + (d))	3-4
14	Load d	1	55	Replace add (m + (d))	4-5
15	Load complement d	1	56	Replace add one (m + (d))	4-5
16	Add d	1	57	Replace subtract one (m + (d))	4-5
17	Subtract d	1			
			60	Central read from (A) to d	min. 6
20	Load dm	2	61	Central read (d) words	5 plus
21	Add dm	2		from (A) to m	5/word
22	Logical product dm	2	62	Central write to (A) from d	min. 6
23	Logical difference dm	2	63	Central write (d) words	5 plus
24	Pass	1		to (A) from m	5/word
25	Pass	1	64	Jump to m if channel d active	2
26	Exchange jump	min. 20	65	Jump to m if channel d inactive	2
27	Read program address	1	66	Jump to m if channel d full	2
			67	Jump to m if channel d empty	2
30	Load (d)	2			
31	Add (d)	2	70	Input to A from channel d	2
32	Subtract (d)	2	71	Input (A) words to m	4 plus
33	Logical difference (d)	2		from channel d	1/word
34	Store (d)	2	72	Output from A on channel d	2
35	Replace add (d)	3	73	Output (A) words from m	4 plus
36	Replace add one (d)	3		on channel d	1/word
37	Replace subtract one (d)	3	74	Activate channel d	2
			75	Disconnect channel d	2
40	Load ((d))	3	76	Function (A) on channel d	2
41	Add ((d))	3	77	Function m on channel d	2

CONTROL DATA SALES OFFICES

ALAMOGORDO • ALBUQUERQUE • ATLANTA • BILLINGS • BOSTON • CAPE CANAVERAL • CHICAGO • CINCINNATI • CLEVELAND • COLORADO SPRINGS DALLAS • DAYTON • DENVER • DETROIT • DOWNEY, CALIFORNIA • HONOLULU HOUSTON • HUNTSVILLE • ITHACA • KANSAS CITY, KANSAS • LOS ANGELES MADISON, WISCONSIN • MINNEAPOLIS • NEWARK • NEW ORLEANS • NEW YORK CITY • OAKLAND • OMAHA • PALO ALTO • PHILADELPHIA • PHOENIX PITTSBURGH • SACRAMENTO • SALT LAKE CITY • SAN BERNARDINO • SAN DIEGO • SEATTLE • ST. LOUIS • WASHINGTON, D.C.

ATHENS • CANBERRA • DUSSELDORF • FRANKFURT • THE HAGUE • HAMBURG JOHANNESBURG • LONDON • MELBOURNE • MEXICO CITY (REGAL ELECTRONICA DE MEXICO, S.A.) • MILAN • MONTREAL • MUNICH • OSLO • OTTAWA PARIS • SAVYON • STOCKHOLM • STUTTGART • SYDNEY • TOKYO (C. ITOH ELECTRONIC COMPUTING SERVICE CO., LTD.) • TORONTO • ZURICH

CONTROL DATA
CORPORATION

8100 34th AVE. SO., MINNEAPOLIS, MINN. 55440

PUB. NO. 60123000 LITHO U.S.A. 12/64